

動画像合成による景観シミュレーション

曾山豊^{*1}

岡本敬^{*2}

鳥脇純一郎^{*3}

^{*1}中部電力(株) ^{*2}ダイキン工業(株) ^{*3}名古屋大学

本研究は大規模建築物が自然景観に与える影響の評価を行うために、設計段階や環境アセスメント等で利用する資料作成を目的としている。そのために、VTRから取り込んだ実際の情景の連続画像とComputer Graphicsで生成した建築構造物とを合成する。本研究で使用するVTRは路上を走る車上から撮影したものであり、耐震台などの特殊な装置は一切使用していない。建築物モデルからCGの画像を作成するために、実写の撮影時のカメラの位置や方向などの逐次変化するパラメータの解析が必要となる。そこで、本研究では背景となるVTR画像のみからカメラの位置・方向などのパラメータ解析を行い、合成VTRを作成した。

Landscape Simulation using Dynamic Image Montage

Yutaka SOYAMA^{*1} Takasi OKAMOTO^{*2} Jun-ichiro TORIWAKI^{*3}

^{*1}Chubu Electric Power Co. ^{*2}Daikin Industries, Ltd. ^{*3}Nagoya University

^{*1}22, Kitasekiyama, Ohdaka-cho, Midori-ku, Nagoya 459 Japan

^{*2}1304, Kanaoka-cho, Sakai-shi, Osaka 591 Japan

^{*3} Furo-cho, Chikusa-ku, Nagoya 464-01 Japan

In this paper, we report a dynamic image montage for previous landscape simulation. The dynamic image montage is created from a construction's image generated by a computer and usual video. The video used in this paper was taken from a vehicle running on a road where power transmission lines will be crossed. Therefore, we developed the dynamic image analysis section extracting positions and directions of the video camera from sequential background video images to allow the dynamic image montage to be created efficiently.

1. はじめに

最近、建築構造物を設計する際にプレゼンテーションや景観評価のためにコンピュータグラフィックス(Computer Graphics 以下ではCG)を利用することが普及し始めている。電力会社では数多くの大規模構造物を有し、その建設の際には事前にその景観の評価を行うことが大変重要なこととなっている。そこで、我々は電力会社における大規模構造物(送電線鉄塔等)が自然景観に与える影響の評価を行うための資料作成を目的とした景観シミュレーションシステムの開発を進めている^[1]。このシステムでは、VTRから取り込んだ実際の情景の連続画像とCGで生成した建築構造物の画像を合成し、該当建築物を実際の景観の中に埋め込んだ合成VTRを作成する。

従来の実写を利用した景観シミュレーションに関する研究では、主に2つの手法が行われている。一つは静止画におけるモンタージュ技法である。これは背景写真に建築構造物を画像合成する手法であり、リアルな合成画像を作成できる^[2]。他方は地形データへ航空写真をマッピングした地形のモデルと建築構造物のモデルを計算機上に作成し、視点等を自由に変えて景観評価のための画像を作成する手法である^[2, 3]。

前述のモンタージュ技法は、背景画像を撮影する際のカメラの位置などと計算機によって建築物の画像を作成する際のそれらを一致させ整合性を取る必要がある。動画像では1コマごとにこの整合性を取る必要があり多大な労力を要するために敬遠されてきた。そこで、本システムではVTR画像からカメラの位置などが自動抽出でき、効率的な合成VTRの作成を可能にした。また、本システムによって作成された合成VTRは静止画では得られなかった情報を有しており、構造建築物の景観に与える効果を事前に評価するために有効であることが解かった。

2. 景観シミュレーションシステムの概要

本システムにおける景観合成VTRの作成手順のあらましは以下の通りである(図1)。

- (1) 実際の景観のVTR画像を用意する。これは路上を走る車上から普通のVTRで撮影し、デジタル化する。
- (2) 建設予定の構造建築物の図面より建築物の3次元モデルを計算機内に生成する。(建築物モデル作成部)
- (3) VTR画像の解析により、各フレームにおけるカメラの位置、方向(カメラパラメータ)を求める。(動画像解析部)
- (4) 上記のカメラパラメータと地図上の建築物位置からVTR画像における建築物の画像を生成する。VTR画像と建築物の画像を合成する。(画像合成部)

以下、第3節で動画像解析部、第4節で建築物モデル作成部、第5節で画像合成部について報告する。

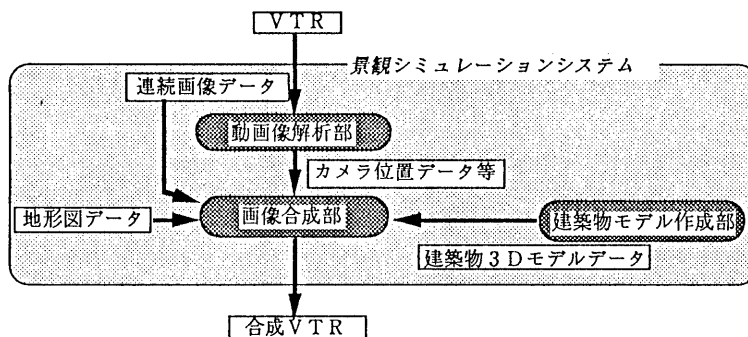


図1 システムの概要

3. 動画像解析部

実写とCGの画像合成を行う場合、実写の撮影条件であるカメラの位置や撮影方向などのパラメータから建築物モデルがいかに見えるかを計算してCGの画像を作成する必要がある。動画像の場合、カメラの位置・方向などのパラメータが逐次変化するのでそれらの解析が必要となる。そこで、背景となるVTRのみからカメラの位置・方向などのパラメータ解析を行う。本研究で使用するVTRは路上を走る車上から撮影したものであり、耐震台などの特殊な装置は一切使用していない。そのためVTR画像には、車の進行による動きのほかに路面の凹凸による揺れなども含まれることを考慮に入れなくてはならない。

動画像解析部は2つの部分から成る。パターンマッチング法によるオプティカルフロー（以下フロー）の算出部と、そのフローからカメラの位置・方向を求める部分である。

3.1 オプティカルフロー算出

フローの算出にはテンプレートマッチング法を用いた。すなわち、第 n フレームの点 (i,j) を中心とする定められた大きさの長方形領域（テンプレート）をとり、第 $n+1$ フレーム上でこれと最も類似した部分画像を選ぶ。具体的には、式(1)の2乗誤差評価式を最小にする k,l を求め、 $(i+k,j+l)$ を中心とする同じ大きさの部分画像をテンプレートの移動後の位置とする。

$$En(k,l) = \sum_i \sum_j (Ln+1(i+k,j+l) - Ln(i,j))^2 \quad (1)$$

$n=0,1,2,\dots$

但し $Ln(i,j)$ は第 n フレームの位置 (i,j) の画素の輝度値

i,j の積算範囲はテンプレートの内部

k,l は第 n フレームと第 $n+1$ フレームの位置の差

なお、第 n フレーム¹で求めたマッチング位置を新しいテンプレートの中心として次の第 $n+1$ フレームでのマッチング位置を探索する。式(1)をそのまま適用することにより第 n フレームと第 $n+1$ フレームの位置の差を求めると、その精度は1ピクセル単位となる。これを各フレーム間で計算していくと、誤差を徐々に累積していくことになり、最終的にはかなりの誤差を生じてしまう。そこで、実際には $En(k,l)$ を k,l に関して補間し、補間した曲面 $En(k,l)$ を最小にする k,l の値を求めることによって、フレーム間のフロー $((k,l)$ の値)はピクセル単位より細かく計算している。これにより誤差の累積の一部を防ぎ、フローの精度を上げることができる。

3.2 カメラの3次元移動ベクトルの算出

VTRから取り込んだ画像は中心投影像とし、図2のように原点をレンズの中心、Y軸方向を視線方向、XZ平面に平行で原点から距離 F （カメラレンズの焦点距離）のところに画像面があるとする。ここでは、平行移動成分、回転成分に分けて解析法を説明し、後にそれぞれの成分の分離法について説明する。

O(原点): レンズ中心

(V_x, V_y, V_z) : カメラの平行移動量

$(\theta_{pitch}, \theta_{roll}, \theta_{yaw})$: カメラの回転移動量

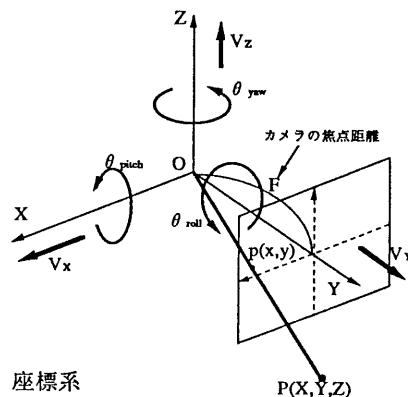


図2 座標系

3.2.1 平行移動成分の解析

点P(X,Y,Z)は式(2)によってスクリーン座標系上の投影点p(x,y)に変換される。

$$x = \frac{X}{Y}F, \quad y = \frac{Z}{Y}F \quad (2)$$

カメラの平行移動による単位時間（本システム内ではフレーム間隔=1/30秒）当たりのX,Y,Z軸方向に関する移動量をそれぞれ V_x, V_y, V_z とする。t時間のカメラ移動後の投影点p'(x',y')は式(3)の様に表される。

$$x' = \frac{X'}{Y'}F = \frac{X - V_x \cdot t}{Y - V_y \cdot t}F, \quad y' = \frac{Z'}{Y'}F = \frac{Z - V_z \cdot t}{Y - V_y \cdot t}F \quad (3)$$

ここで、シーン中のある点P(X,Y,Z)がカメラから遠ざかる様な平行移動速度成分(V_x, V_y, V_z)のみからなる運動でカメラが移動していくと、点Pの投影像は空間内での点Pの位置とは無関係に平行移動速度成分によってのみ決まるスクリーン上の1点（式(4)）に収束する。この点を消失点という^[4]。

$$(x,y) = \left(\frac{V_x}{V_y}F, \frac{V_z}{V_y}F \right) \quad (4)$$

上式(4)は式(3)のtを $t \rightarrow \infty$ （または $t \rightarrow -\infty$ ）とすることで求めることができる。

従って、消失点から式(4)によってカメラの速度比が求まり、さらに画面内に既知点が1点以上存在すれば、式(4)を式(3)に代入することで、平行移動成分を一意に決定できる。

3.2.2 回転成分の解析

カメラの回転による単位時間当たりのX,Y,Z軸に関する回転角をそれぞれ θ_{pitch} , θ_{roll} , θ_{yaw} とする（すべて右ねじの方向を正方向とする）。カメラの回転による変換式は下式（5）の様に表される。

（式(5)の導出は付録参照）

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \quad (5)$$

ここで、 $(X,Y,Z)^T$ はカメラ移動前の座標系に於ける点Pの位置
 $(X',Y',Z')^T$ はカメラ移動後の座標系に於ける点Pの位置

カメラ移動後の投影点p'(x',y')は式（2）と式（5）から下式（6）の様に表される。

$$x' = \frac{X'}{Y'}F = \frac{X + \theta_{yaw}Y - \theta_{roll}Z}{-\theta_{yaw}X + Y + \theta_{pitch}Z}F = \frac{x + \theta_{yaw}F - \theta_{roll}y}{-\theta_{yaw}x + F + \theta_{pitch}y}F$$

$$y' = \frac{Z'}{Y'}F = \frac{\theta_{roll}X - \theta_{pitch}Y + Z}{-\theta_{yaw}X + Y + \theta_{pitch}Z}F = \frac{\theta_{roll}x - \theta_{pitch}F + y}{-\theta_{yaw}x + F + \theta_{pitch}y}F$$

以上より、回転成分だけの場合、画面上での速度成分($\delta x, \delta y$)が画面上での位置によって一意に、式(17)のように決まる^[5]。

$$\delta x = x' - x = \theta_{yaw}F - \theta_{roll}y - \frac{1}{F}(-\theta_{yaw}x + \theta_{pitch}y)x \quad (7)$$

$$\delta y = y' - y = -\theta_{pitch}F + \theta_{roll}y - \frac{1}{F}(-\theta_{yaw}x + \theta_{pitch}y)y$$

ここでは $F \gg \theta_{yaw}x, \theta_{pitch}y$ としている。

3.2.3 平行移動と回転成分の分離

3.1で求めるフローは上述の平行移動と回転両成分の和として形成されている。次に、それらの分離方法について述べる。

まず、回転成分 θ_{roll} について述べる。説明を簡単にするために $\theta_{pitch} = \theta_{yaw} = 0$ と仮定する。平行移動成分のみの時はオブティカルフローは消失点に収束するはずである。そこで、ある θ_{roll} を仮定して式(5)によりフローを回転させ、以下のようにフローの収束度を計算する。

〈収束度の計算方法〉

- (1) すべてのフローを延長し、各フローどうしの交点を求める。
- (2) (1) より求めたすべての交点について、式 (8) のように重み付け平均をして消失点の候補点を求める。

$$(Pv_x, Pv_y) = \frac{\sum_k (Px_k, Py_k) \times \text{weight}_k}{\sum_k \text{weight}_k} \quad (8)$$

$$\text{weight}_k = \left| \delta x_{1k} \cdot \delta y_{2k} - \delta x_{2k} \cdot \delta y_{1k} \right|$$

(Pv_x, Pv_y) : 消失点の候補点

(Px_k, Py_k) : フロー間の交点

kはすべてのフローの交点の中でk番目の交点を表す

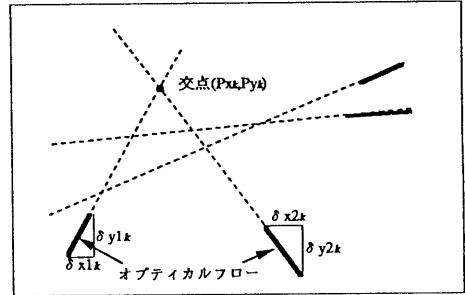


図3 式(8)説明図

i3) 消失点の候補点について式 (9) の評価式により消失点の収束度を計算する。

$$E = \frac{\sum_n \left(\left| \delta x_{1n} \cdot \delta Pv_{yn} - \delta y_{1n} \cdot \delta Pv_{xn} \right| \times (\text{第} n \text{ 番目のフローの長さ}) \right)}{\sum_n (\text{第} n \text{ 番目のフローの長さ})} \quad (9)$$

nはすべてのフローの中でn番目のフローを表す

これを幾つかの θ_{roll} について行い、上記の収束度が最大（評価式(9)では最小）の時の回転角および速度比を最適値として選ぶ。

次に、平行移動成分と回転成分 θ_{pitch} , θ_{yaw} の分離について述べる。説明を簡単にするために $\theta_{roll} = 0$ と仮定する。フローを求めた点Pが遠くにあるとき、すなわち $X, Y, Z \gg V_x, V_y, V_z$ のときは式(3)からわかるように、カメラが平行移動しても投影点pはほとんど移動しない。しかし、式(7)からわかるようにカメラが回転移動すると投影

点pは移動する。従って、遠距離点のフローから式(7)を使って回転成分 θ_{pitch} , θ_{yaw} を求めることができる。また他の近距離点のフローを式(7)によって回転成分を取り除いたフローに変換し、消失点から式(4)を用いて速度比を求める。

すべてを考慮したアルゴリズムを図5に示す。

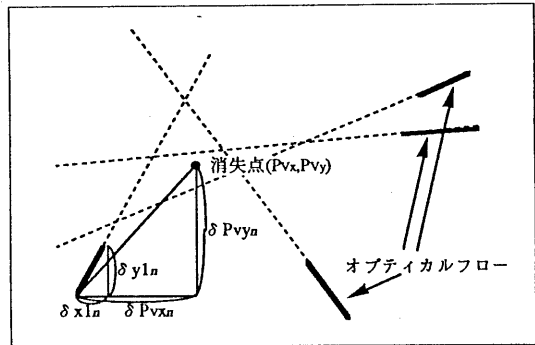


図4 式(9)説明図

4. 建築物モデル作成部

電力設備の中には事前の景観シミュレーションを必要とする大規模建築物が数多く存在する。本研究では、その1つである送電線鉄塔を例に建築物モデル作成部を開発した^[6]。送電線鉄塔のモデルは、建設予定の送電線鉄塔の図面から送電線鉄塔本体や舟形（鉄塔のアーム部）の梁の位置情報を2次元的に入力し、送電線鉄塔の形の対称性などを利用してシステムが自動的に3次元モデルを作成するようになっている。送電線鉄塔のモデルは他に、建設予定地の位置や送電線鉄塔の本体部・舟形の梁の大きさ、色及び形状（丸材や角材など）、また碍子の種類や接続位置、そして送電線の張具合^[7]などを求めるための送電線重量や張力などの各種パラメータを入力して作成される。図6に送電線鉄塔モデルの作成画面とそれより作成された送電線鉄塔のモデルのCGの画像を示す。

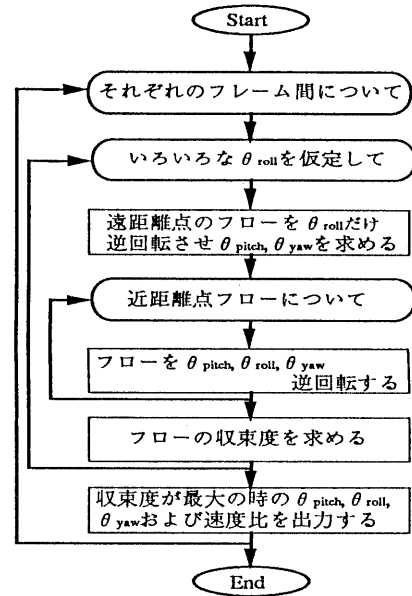


図5 成分分離の手法

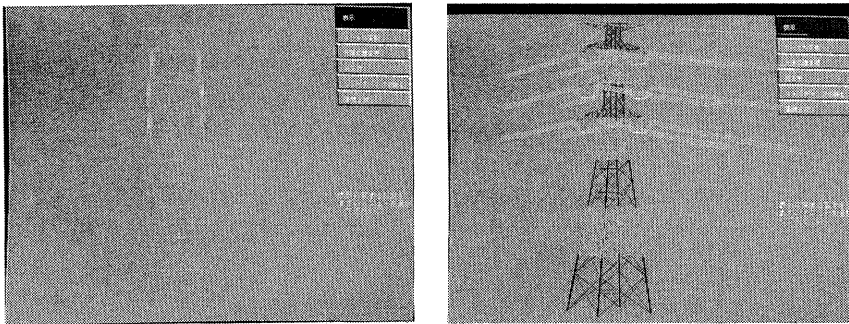


図6 送電線鉄塔モデル作成画面と完成した送電線鉄塔のCG

5. 画像合成部

第3節で述べた手法によるVTRの解析結果と第4節で述べた送電線鉄塔のモデルのCG画像の合成を本節で行う。VTRの解析結果より撮影時の逐次のカメラの位置や撮影方向などのパラメータが1フレーム毎に得られ、それをもとに送電線鉄塔モデルが投影面（VTR 1フレーム）上にいかに見えるかを計算して1フレーム毎の送電線鉄塔モデルのCGの画像を作成する。以上までにより、投影面上での2次元の整合性が取れたことになる。次に、奥行き方向の整合性を取る必要がある。これは送電線鉄塔の足元が山なみにより隠れたりする時などに必要となる。送電線鉄塔モデルはその建設予定位置の情報も有しているため撮影位置からの距離を容易に計算することにより視点からの奥行き値が解かる。しかし、背景画像となるVTRの各1フレームは奥行き方向の情報は有しておらず、合成の際に前後関係を加味することはできない。そこで以下の方でVTRの1フレームの画像に奥行き値を付加してやることにより前後関係を判断し合成画像を作成する。まず、VTRの最初の1フレームの画像を明度値によるしきい値処理（しきい値を手入力）により分割する。そこで分割された各領域に対して疑似的な奥行き値を手入力する。VTRの2フレーム以降は先のしきい値で画像を分割し、各領域の疑似奥行き値も先の値をそのまま使用する。画像に与える疑似奥行き値は、分割された画像の部分と送電線鉄塔のモデルの前後関係が判断できれば十分であるので高精度である必要はない。極端な場合、画像上の空の部分は ∞ を、そして

山の部分には0の疑似奥行き値を与えれば、送電線鉄塔のモデルは常に送電線鉄塔の足元は山に隠れ、スカイラインを割った送電線鉄塔の上部は描かれる。

6. 適用例

今回作成した合成ビデオは全部で711フレーム（24秒弱）である。1フレームの大きさは640画素×480画素で、各画素はRGB各々8bitである。計算時間のほとんどは、フローの算出とカメラの運動の解析に費やされる。これら二つの計算に要する1フレーム当たりの平均CPUタイムは前者が62.8秒、後者が47.4秒かかる（使用計算機はIRIS 4D 320 VGX）。図7に合成前後のVTRの1フレームを示す。

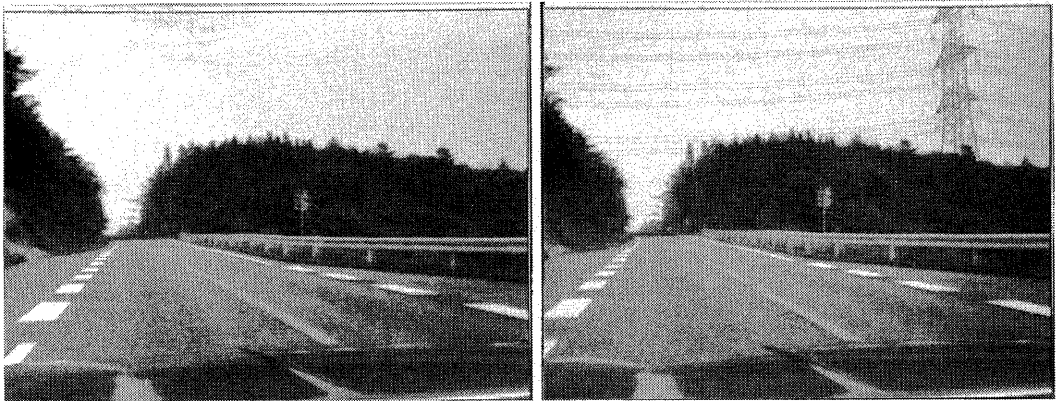


図7 合成前のVTRと合成後のVTR例（1フレーム）

7. まとめ

動画像合成による景観シミュレーションの手法とその適用例について報告した。本方法を実際の送電線鉄塔に適用した結果、動画像による景観シミュレーションによって静止画では得られなかった情報、すなわち構造物がどのあたりからいかにして見え始め、かつ、どのように変化していくかなどの情報が得られ、構造建築物の景観に与える影響を事前に評価するために有効であることが解かった。

謝辞

本研究の共同研究者であるダイキン工業（株）CAEセンターの石淵耕一氏、高島純氏、平野徹氏に感謝致します。日頃、御討論していただく横井茂樹助教授、安田孝美氏をはじめとする名古屋大学情報工学科鳥脇研究室の皆様には感謝致します。また本研究の機会を与え、ご助言いただいた中部電力（株）電力技術研究所の皆様には感謝致します。

参考文献

- 【1】石淵、高島、平野、曾山、志澤、井上：動画像処理技術を用いた景観シミュレーション、第6回 NICOGRAPH論文コンテスト論文集、pp.329-334,1990
- 【2】中前：景観シミュレーション、情報研報、Vol.89,No.64,pp.1-7,1989
- 【3】曾山、嶋本、石淵：航空写真を利用した地形図データの半自動作成および景観シミュレーションへの利用、平成3年電気学会全国大会講演論文集、p13-70,1991
- 【4】H. Nakatani, S. Kimura, O. Saito, T. Kitahasi : "Extraction of vanishing point and its application to scene analysis based on image sequence", Proc. 5th ICPR ,pp.370-372, 1980
- 【5】Ken-ichi Kanatani "Transformation of Optical Flow by Camera Rotation" IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, Vol. 10, No.2, March 1988
- 【6】松田、田中、高月、高島、石淵、平野：電力分野における画像処理技術の応用（その1）---鉄塔モデラーの開発---、情報処理学会第38回全国大会、pp.705-706,1989
- 【7】竹下：架空送電線の弛度--弛度・張力の理論とその計算--、電力社(1966)

付録 式 (5) の導出について

カメラの回転による単位時間当たりのX,Y,Z軸に関する回転角をそれぞれ θ_{pitch} , θ_{roll} , θ_{yaw} とする (すべて右ねじの方向を正方向とする)。それぞれの変換行列を式 (a.1) ~ 式 (a.2) に示す。

$$R_{\theta_{pitch}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{pitch} & \sin \theta_{pitch} \\ 0 & -\sin \theta_{pitch} & \cos \theta_{pitch} \end{pmatrix} \quad (a.1)$$

$$R_{\theta_{roll}} = \begin{pmatrix} \cos \theta_{roll} & 0 & -\sin \theta_{roll} \\ 0 & 1 & 0 \\ \sin \theta_{roll} & 0 & \cos \theta_{roll} \end{pmatrix} \quad (a.2)$$

$$R_{\theta_{yaw}} = \begin{pmatrix} \cos \theta_{yaw} & \sin \theta_{yaw} & 0 \\ -\sin \theta_{yaw} & \cos \theta_{yaw} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (a.3)$$

ここで、式 (a.4) の様に近似する。

$$\begin{aligned} \sin \theta_{pitch} &\approx \theta_{pitch}, \sin \theta_{roll} \approx \theta_{roll}, \sin \theta_{yaw} \approx \theta_{yaw} \\ \cos \theta_{pitch} &\approx 1, \cos \theta_{roll} \approx 1, \cos \theta_{yaw} \approx 1 \end{aligned} \quad (a.4)$$

それぞれの軸に対する回転行列の合成は式 (a.4) を式 (a.1) ~ 式 (a.3) に代入しX,Y,Z軸に関する回転の順番をすべての場合について行い、さらに θ_{pitch} , θ_{roll} , θ_{yaw} の2次以上の項を無視すると以下のようになる。

$$R_{\theta_{pitch}} R_{\theta_{roll}} R_{\theta_{yaw}} = \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} + \theta_{pitch} \theta_{roll} & 1 + \theta_{pitch} \theta_{roll} \theta_{yaw} & \theta_{pitch} \\ \theta_{roll} + \theta_{pitch} \theta_{yaw} & -\theta_{pitch} + \theta_{roll} \theta_{yaw} & 1 \end{pmatrix} \approx \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \quad (a.5)$$

$$R_{\theta_{pitch}} R_{\theta_{yaw}} R_{\theta_{roll}} = \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} + \theta_{pitch} \theta_{roll} & 1 & \theta_{pitch} + \theta_{roll} \theta_{yaw} \\ \theta_{roll} + \theta_{pitch} \theta_{yaw} & -\theta_{pitch} & 1 - \theta_{pitch} \theta_{roll} \theta_{yaw} \end{pmatrix} \approx \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \quad (a.6)$$

$$R_{\theta_{roll}} R_{\theta_{pitch}} R_{\theta_{yaw}} = \begin{pmatrix} 1 - \theta_{pitch} \theta_{roll} \theta_{yaw} & \theta_{yaw} + \theta_{pitch} \theta_{roll} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} + \theta_{pitch} \theta_{yaw} & -\theta_{pitch} + \theta_{roll} \theta_{yaw} & 1 \end{pmatrix} \approx \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \quad (a.7)$$

$$R_{\theta_{roll}} R_{\theta_{yaw}} R_{\theta_{pitch}} = \begin{pmatrix} 1 & \theta_{yaw} + \theta_{pitch} \theta_{roll} & -\theta_{roll} + \theta_{pitch} \theta_{yaw} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} + \theta_{roll} \theta_{yaw} & 1 + \theta_{pitch} \theta_{roll} \theta_{yaw} \end{pmatrix} \approx \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \quad (a.8)$$

$$R_{\theta_{yaw}} R_{\theta_{pitch}} R_{\theta_{roll}} = \begin{pmatrix} 1 + \theta_{pitch} \theta_{roll} \theta_{yaw} & \theta_{yaw} & -\theta_{roll} + \theta_{pitch} \theta_{yaw} \\ -\theta_{yaw} + \theta_{pitch} \theta_{roll} & 1 & \theta_{pitch} + \theta_{roll} \theta_{yaw} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \approx \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \quad (a.9)$$

$$R_{\theta_{yaw}} R_{\theta_{roll}} R_{\theta_{pitch}} = \begin{pmatrix} 1 & \theta_{yaw} + \theta_{pitch} \theta_{roll} & -\theta_{roll} + \theta_{pitch} \theta_{yaw} \\ -\theta_{yaw} & 1 - \theta_{pitch} \theta_{roll} \theta_{yaw} & \theta_{pitch} + \theta_{roll} \theta_{yaw} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \approx \begin{pmatrix} 1 & \theta_{yaw} & -\theta_{roll} \\ -\theta_{yaw} & 1 & \theta_{pitch} \\ \theta_{roll} & -\theta_{pitch} & 1 \end{pmatrix} \quad (a.10)$$

以上、式 (a.5) ~ (a.10) から解かるようにカメラの回転による変換式は式 (5) の様に表される。