

# 人の眼球運動に基づくロボットカメラの制御

## Control of Robot Camera based on Human Eye Motion

○植野博之\*, 村松鋭一\*

○Hiroyuki Ueno\*, Eiichi Muramatsu\*

\*山形大学

\*Yamagata University

**キーワード** : フィードバック誤差学習(Feedback error learning), 協調動作(Cooperative Motion)  
前庭動眼反射(Vestibulo Ocular Reflex)

**連絡先** : 〒992-8510 米沢市城南 4-3-16 山形大学工学部応用生命システム工学科 村松研究室  
村松 鋭一, tell 0238-26-3327, E-mail : muramatu@yz.yamagata-u.ac.jp

### 1 はじめに

ロボットの視覚情報をもとにロボットカメラの視界中央を物体の動きに追従させる制御をビジュアルサーボ制御という。従来のロボットカメラ制御の研究においては、カメラ単独での制御が多く、人間のように個々の眼球が適応的かつ全体が協調的に動作するようなシステムの研究はあまりなされていない。

そこで本研究では人の眼球運動のように2つの個々のカメラを制御し、人の眼球運動のような両眼による協調的動作かつ素早く滑らかな動作で追従対象の注視を行うための制御システムやそれに伴う計算のアルゴリズムを構築していく。また、両眼とそれを搭載した頭部とで協調性を持たせることにより、さらに物体の追従性能・効率が上がるのではないのかという仮定を立て、頭部とカメラの協調動作のためのシステムの構築と検証を行っていく。

本研究を進める上で注目したのが人の前庭小脳で行われている「前庭動眼反射」である。この動作は頭部の動作を半規管で検出し、その動きとは逆方向に眼球を動かし網膜上のぶれを抑える前向き制御である。この動作に注目した理由は2つある。

まず1つは、この動作の実現により、頭部と両眼に協調性を持たせることが可能だと考えたからである。そもそも、前庭動眼反射は頭部の傾きにに応じて同時に目が頭部傾きの逆方向に動作する制御である。よって、前庭動眼反射の実現は頭

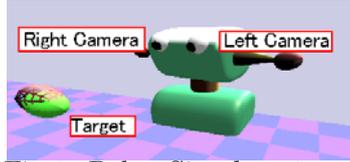
部とカメラの協調動作の実現につながる。

2つ目は、ロボットカメラに人同様な滑らかで素早い動作の獲得させるためである。頭部の動作に合わせてカメラを動かす際に、フィードバック制御だけではループタイムが長い。また、不安定に陥らないためにゲインも小さく設定されなくてはならないために、素早く滑らかな動作の実現は困難である。そこで、前庭動眼反射でも行われている前向き制御、すなわちフィードフォワード制御が必要であると考えた。このフィードフォワード制御を行うために生体では、小脳で運動系の逆モデルを生成していることが分かっている。逆モデルは実際に行動を行った際に生成される。運動誤差信号が用いられた学習により適応的に形成される。このような逆モデル獲得の学習方法をフィードバック誤差学習といい Kawato によって提案されている。[1] 本研究ではカメラの制御にフィードバック誤差学習を適用することで、ロボットカメラをすばやく滑らかに制御しようと考えた。以上のことから、まずロボットに前庭動眼反射の実現のための制御システムの構築を行った。

次に、頭部とカメラの協調動作の方法として、幾何学的に追従対象をロボットカメラ画像中心で注視するために、頭部とカメラの回転角度の和の最小を求める計算手法を考案した。またその手法を組み込んだ人型ロボットの物体追従のシミュレーションを、3次元コンピュータグラフィックを用いたシミュレータで検証し計算手法の妥当性を検証していく。

## 2 3次元CGシミュレータ

本研究では、制御システムの検証のために3次元コンピュータグラフィックス(以下3DCG)のシミュレータを開発した。この、3DCGシミュレータはグラフィックライブラリ OpenGL を用いてC言語で作成したものである。

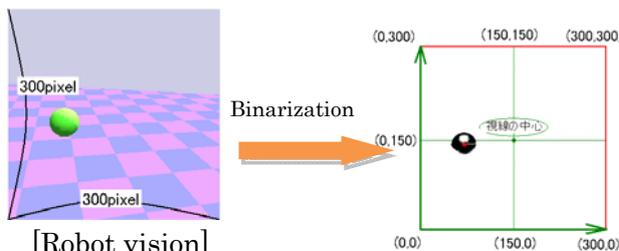


< Fig.1 : Robot Simulator >

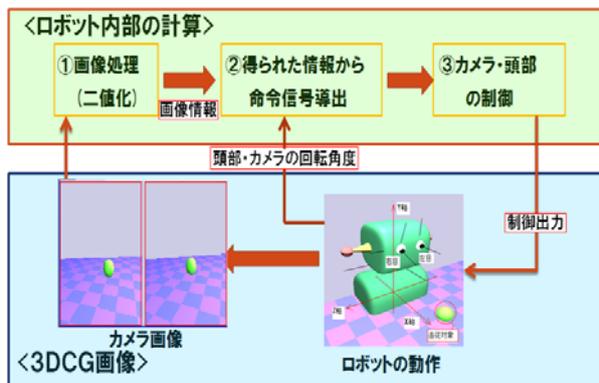
研究に用いたロボットシミュレータのCGはFig.1で示す通りである。このロボットは頭部に2つのカメラを搭載した人型のロボットを想定している。また、このロボットは頭部とカメラの回転角度を角度センサで測定できるものとしている。そこで得られた頭部とカメラの回転角度は、頭部とカメラの目標信号を求める情報の1つとなる。

また、本研究でロボットが追従対象とするのはFig.1中のTargetで示したような、球状の物体である。この追従対象は半径0.1[m]の緑色の球としている。ロボットはカメラに映った追従対象の色情報を基に二値化の画像処理を行う。そして、追従対象がカメラ画像のどの位置で捉えられたかを座標として割り出している。(Fig.2) この画像処理の結果も、頭部とカメラの動作の目標信号を求める情報の1つである。

上記のようにして得られた目標信号を基にロボットを動作させ、追従対象をロボットカメラで注視できているかどうかを検証していく。(Fig.3)



< Fig.2:Image processing >

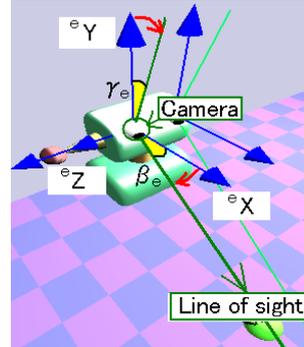


< Fig.3:Flow chart of Robot Simulator >

## 3 制御対象

### 3.1 ロボットカメラ

まず、制御対象の1つであるロボットカメラの動作について説明する。カメラのローカル座標系を $\Sigma_e$ として下のFig.4のように右手系での座標軸をとる。



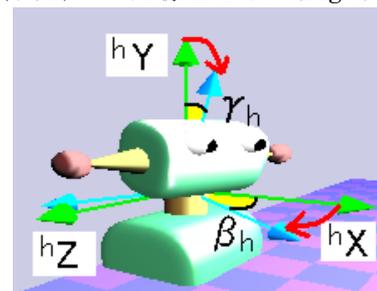
< Fig.4:Local coordinate system of Camera >

Fig.4の場合におけるカメラの動作は回転と並進を考慮すると6自由度挙げられるが、本研究では $e_y$ 軸と $e_z$ 軸の回転の2自由度を制御することでカメラを動作させる。この際、カメラの座標軸 $e_y$ 回りの回転角度を $\beta_e$ 、 $e_z$ 軸回りの回転角度を $\gamma_e$ と定義している。

カメラの動作は、 $\beta_e$ がプラスになるような目標入力を加えると左方向に動く。同じように、 $\beta_e$ がマイナスになるような入力を加えると右に動く。また $\gamma_e$ をプラスにすると上に、マイナスにすると下に動作する。

### 3.2 ロボットの頭部

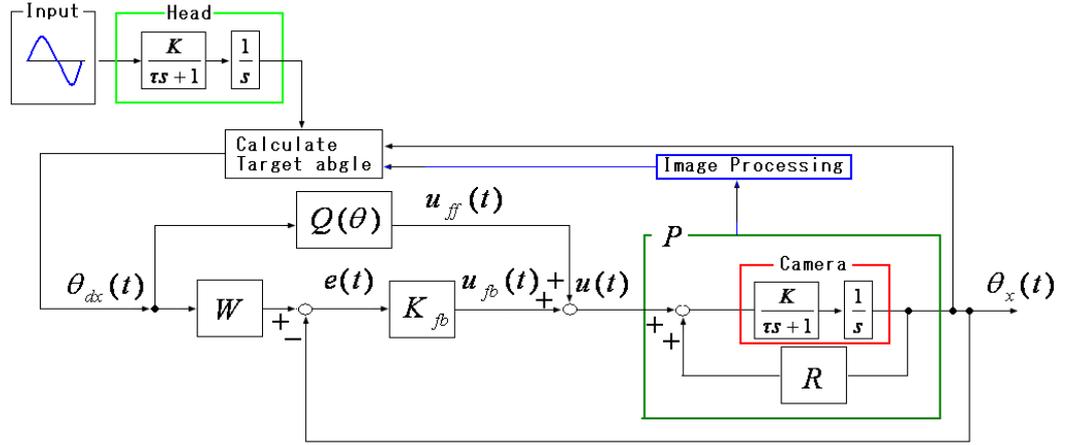
次に、制御対象の2つめがロボットの頭部である。ロボット頭部のローカル座標系を $\Sigma_h$ としてカメラ同様右手系での座標をとる。(Fig.5)



< Fig.5: Local coordinate system of Head >

カメラ同様に頭部も6自由度であるが、 $h_y$ 軸回りと $h_z$ 軸回りの回転の2自由度を制御している。この時、頭部座標軸 $h_y$ 回りの回転角度を $\beta_h$ とし、頭部回転軸 $h_z$ 回りの回転角度を $\gamma_h$ とする。

頭部の動作も基本的にはカメラの動作と同じである。 $\beta_h$ がプラスになるような目標入力がかわれば左に頭部が動き、マイナスであれば右に動く。また、 $\gamma_h$ がプラスになれば上、マイナスであれば下に動く。



< Fig.6:Block Diagram of VOR(Vestibulo Ocular Reflex) >

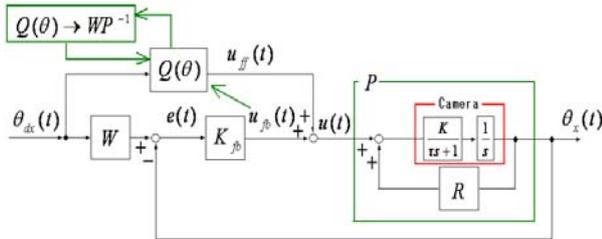
## 4 前庭動眼反射のシミュレーション

### 4.1 シミュレーションブロック線図

前庭動眼反射を行わせるためのブロック線図は Fig.6 のようになる。カメラは1自由度の回路になっているが、これを各軸に適用することで両眼2軸の動作が可能になる。今回は頭部の1軸( $\theta_{dx}$ )に正弦波を加えて頭部を左右に動かしている。そして、カメラが頭部の動きに応じて画像偏差を抑えるように学習をしながら制御を行っていく。

### 4.2 フィードバック誤差学習

ここでは、本研究で適用したフィードバック誤差学習の制御器の説明をしていく。Fig.7はFig.6のカメラの制御部分を抜粋したものであり、カメラの目標入力 $\theta_{dx}$ から出力 $\theta_x$ を出力するブロック線図である。



< Fig.7:Feedback error learning >

$K_{fb}$ はフィードバックゲインである。制御対象はモータで回転するカメラであり、モータの伝達関数には積分器を含む。これを解消するためにモータに正の実数である定数  $R$  でローカルフィードバックをかけており、モータにローカルフィードバックをかけた回路を制御対象  $P$  とする。また、 $Q(\theta)$ は逆モデルを形成している部分でフィードフォワード信号( $u_{ff}$ )を出力する。

このとき、 $Q(\theta)$ により制御対象  $P$ の逆モデルすなわち $P^{-1}$ を形成しようとするとき、逆モデルの相対次数は分母より分子の方が大きくなり制御に不適になってしまう。それを補償するのが  $W$ で

ある。 $W$ の設定条件としては、

- (1) 逆モデル $P^{-1}$ の分子の次数と同じ次数をもつ。
- (2)  $W \equiv I$ になること。

である。ちなみに Fig.7 の伝達関数を求めると、

$$\theta_x = \frac{W(1+PK_{fb})}{1+PK_{fb}} \theta_{dx} \quad (1)$$

となるため、ここで $\theta_x = \theta_{dx}$ とするために $W \equiv I$ となる必要がある。この次数調整要素  $W$ を用いた場合のフィードフォワード信号 $u_{ff}$ の導出の計算は文献[3]より、

$$\frac{d\xi_1}{dt} = F\xi_1 + g\theta_{dx}(t) \quad (2)$$

$$\frac{d\xi_2}{dt} = F\xi_2 + gu(t) \quad (3)$$

$$\begin{aligned} u_{ff}(t) &= c^T(t)\xi_1(t) + d^T(t)\xi_2(t) + k(t)\theta_{dx}(t) \\ &= \theta(t)^T \xi(t) \end{aligned} \quad (4)$$

ただし、

$$\xi(t) = [\xi_1(t)^T \ \xi_2(t)^T \ r(t)^T]^T \quad (5)$$

また、 $F$ は安定な行列、 $g$ は $(F, g)$ が可制御となるベクトルである。さらに $\theta(t)$ は $Q(\theta) \rightarrow WP^{-1}$ を達成するための可調整パラメータベクトルであり、

$$\frac{d\theta}{dt} = \alpha \tilde{\xi}_w(t) \{c(t)^T \xi_e(t) + k(t)e(t) + W(s)u_{fb}(t)\} \quad (6)$$

ただし、

$$\tilde{\xi}_w(t) = \begin{bmatrix} \xi_{wr}(t) - \xi_e(t) \\ \xi_{wu}(t) \\ W(s)r(t) - e(t) \end{bmatrix} \quad (7)$$

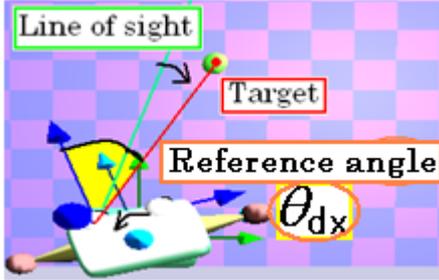
$$\frac{d\xi_{wr}(t)}{dt} = F\xi_{wr}(t) + gW(s)\theta_{dx}(t) \quad (8)$$

$$\frac{d\xi_{wu}(t)}{dt} = F\xi_{wu}(t) + gW(s)u(t) \quad (9)$$

である。また式(6)中の  $\alpha$ は調整則 $\theta(t)$ の速度を調整するパラメータであり、 $\alpha > 0$ である。

### 4. 3 目標入力

この制御対象の目標値は Fig.8 中の  $\theta_{dx}$  であり、現在の視線の位置から追従対象を注視するためのカメラの回転角度となる。この回転角度を目標角と定義する。目標角は画像情報やカメラの回転角度から幾何学的に求めている。



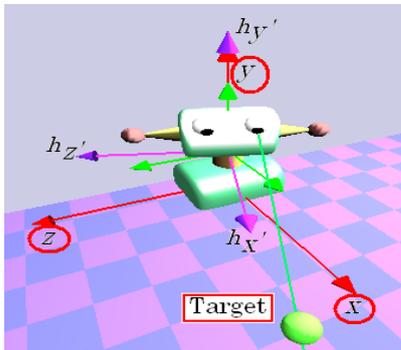
< Fig.8:Definiton of Reference angle >

目標角を導出する過程は大きく 2 つあり、

- (1) 追従対象の座標を求める。
- (2) 追従対象の座標を基に目標角を導出の手順で求める。

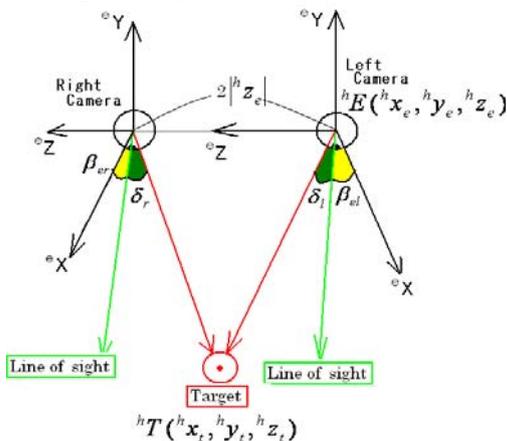
#### 4. 3. 1 追従対象の座標算出

ロボットの中心と地面との接点を絶対座標での原点とし、そこから追従対象の空間座標を求め、(Fig.9 の  $(x, y, z)$  の座標軸が絶対座標軸)



< Fig.9:Each coordinate axes >

まず頭部回転後の座標軸  $(h_x', h_y', h_z')$  における追従対象の座標を  ${}^hT = ({}^hx_t, {}^hy_t, {}^hz_t)$  として求める。(Fig.9) (Fig.10)



< Fig.10:Coordinates of Target >

このとき、座標軸  $({}^hx', {}^hy', {}^hz')$  における左カメラの初期位置を  ${}^hE = ({}^hx_e, {}^hy_e, {}^hz_e)$  とする。また、左右カメラのカメラ座標軸  ${}^ey$  における回転角度を  $\beta_{el}$ ,  $\beta_{er}$  とそれぞれおく。さらに左右カメラ視線から追従対象までの横方向の偏差の角度を  $\delta_l$ ,  $\delta_r$  とし、縦方向の偏差角度は  $\varepsilon$  とする。このとき追従対象  ${}^hT = ({}^hx_t, {}^hy_t, {}^hz_t)$  は

$${}^hT = \begin{bmatrix} {}^hx_t \\ {}^hy_t \\ {}^hz_t \end{bmatrix} = \frac{2 \cdot |{}^hz_t| \cos(\beta_{el} + \delta_l)}{\sin(\beta_{er} - \beta_{el})} \begin{bmatrix} \cos(\beta_{er} + \delta_r) \\ \tan(\gamma + \varepsilon) \\ -\sin(\beta_{er} + \delta_r) \end{bmatrix} + \begin{bmatrix} {}^hx_e \\ {}^hy_e \\ {}^hz_e \end{bmatrix} \quad (10)$$

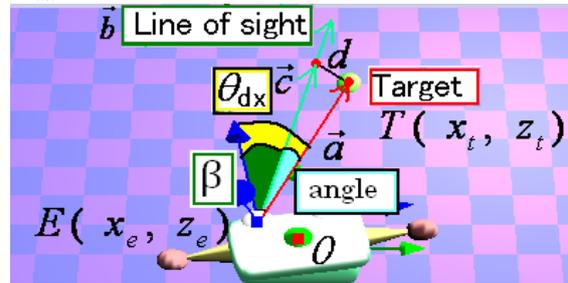
となる。さらに座標軸  $({}^hx', {}^hy', {}^hz')$  から絶対座標系での座標  $T(x_t, y_t, z_t)$  へ座標変換すると、

$$T = \begin{bmatrix} 0 \\ y_h \\ 0 \end{bmatrix} + \begin{bmatrix} \cos \beta_h & 0 & \sin \beta_h \\ 0 & 1 & 0 \\ -\sin \beta_h & 0 & \cos \beta_h \end{bmatrix} \begin{bmatrix} \cos \gamma_h & -\sin \gamma_h & 0 \\ \sin \gamma_h & \cos \gamma_h & 0 \\ 0 & 0 & 1 \end{bmatrix} {}^hT \quad (11)$$

となる。式中の  $y_h$  は絶対座標軸の原点から頭部回転軸の原点までの距離を示す。

#### 4. 3. 2 目標角の導出

前庭動眼反射の目標信号は頭部の角度変化に応じてカメラを動作させる必要がある。今頭部が微小に移動した場合の目標信号にあたる目標角  $(\theta_{dx})$  を考える。



< Fig.11:Calculation of Reference angle (theta\_dx) >

Fig.11 において、カメラの位置を点  $E(x_e, z_e)$ 、追従対象座標を  $T(x_t, z_t)$ 、視線の先端座標を  $(x_l, z_l)$ 、絶対座標の中心点を点  $o$ 、カメラ(点  $E$ ) から追従対象までのベクトルを  $\vec{a}$ 、視線のベクトルを  $\vec{b}$ 、 $\vec{a}$  と垂直で視線  $\vec{b}$  と接する線を  $d$ 、視線上で辺  $d$  と重なるベクトルを  $\vec{c}$  とおく、 $\vec{a}$  と  $\vec{c}$  は以下のように表すことができる。

$$\vec{a} = (x_t - x_e, z_t - z_e) \quad (12)$$

$$\vec{b} = (x_l - x_e, z_l - z_e) \quad (13)$$

また、辺  $d$  の大きさを求めるために、

$$|\vec{a}| = \sqrt{(x_t - x_e)^2 + (z_t - z_e)^2} \quad (14)$$

$$|\vec{c}| = \frac{\vec{a} \cdot \vec{b}}{|\vec{b}|} = \frac{(x_t - x_e)(x_l - x_e) + (z_t - z_e)(z_l - z_e)}{\sqrt{(x_l - x_e)^2 + (z_l - z_e)^2}} \quad (15)$$

を求めると、式(14), (15)から

$$d = \sqrt{a^2 + c^2} \quad (16)$$

となり、視線と追従対象の間の角  $angle$  は

$$angle = \tan^{-1}\left(\frac{d}{c}\right) \quad (17)$$

となり目標角  $\theta_{dx}$  はカメラの回転角度  $\beta$  と  $angle$  の和なので

$$\theta_{dx} = \beta + \tan^{-1}\left(\frac{d}{c}\right) \quad (18)$$

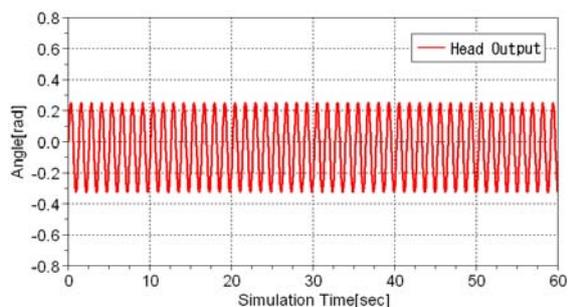
と導出できる。

## 4.4 シミュレーション

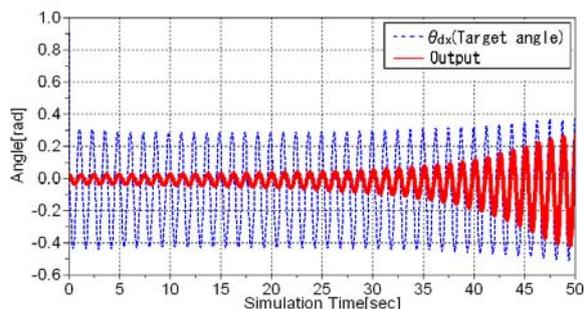
第2節で述べた3DCGシミュレータにFig.6の回路を組み込んでシミュレーションを行った。シミュレーションの行い方は以下の通りである。

- (1) 追従対象を絶対座標 (2.0, 0.5, 0.0) に固定。
- (2) 頭部回転軸  $h_y$  回りのモータに正弦波を加え左右に動作させる。

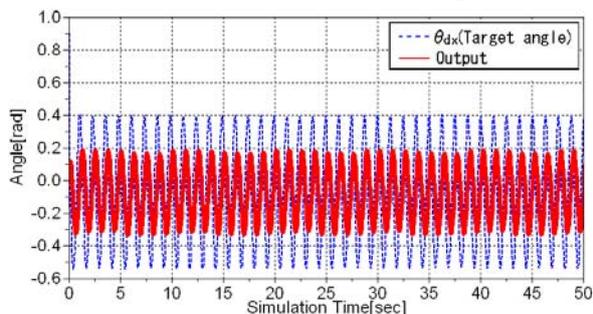
## 4.5 シミュレーション結果



< Fig.12:Head motion >



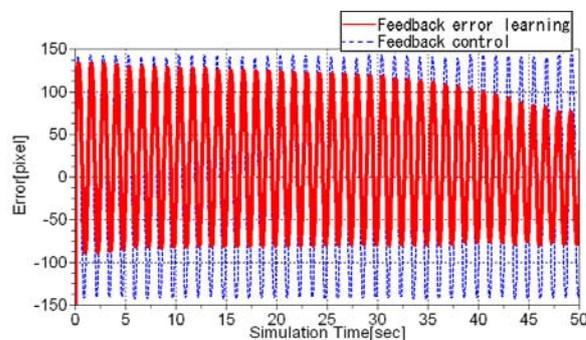
< a > Feedback error learning



< b > Feedback control only

< Fig.13:Camera motion >

Fig.12はシミュレーション時の頭部の動作である。1.25秒で左右に往復している。また、Fig.13は左カメラの動作である。<a>はフィードバック誤差学習を用いた結果で<b>はフィードバック制御のみの結果である。<b>の方は目標入力どおりに出力が出ていないのがわかる。一方<a>の方はシミュレーション時間が進むほど目標角の目標入力に出力が合致してきているのがわかる。



< Fig.14:Camera Image declination >

さらに、追従能力の評価のためにカメラ画像中心からカメラで投影された追従対象までの画像上での距離を出した。(Fig.14)この距離が0に近いほどカメラの中心で追従対象をとらえていることになり追従性能が高いことを示す。学習が進むほどにフィードバック誤差学習の追従性能が向上してきていることがわかる。

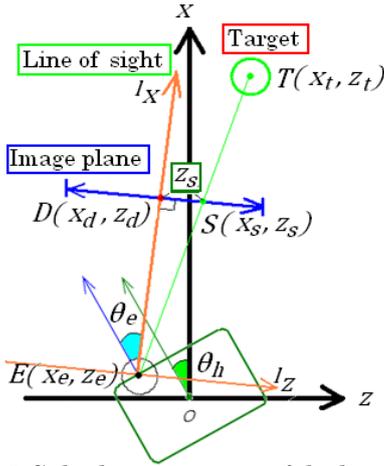
以上からフィードバック誤差学習を用いた前庭動眼反射のシミュレーションは成功したといえる。頭部とカメラの協調動作とフィードバック誤差学習を用いたカメラの動作を活かして、以下の項では頭部とカメラの協調動作の手法を検討していく。

## 5 頭部とカメラの協調動作

### 5.1 基本原理

頭部、カメラが静止した物体を注視している状態から、追従対象が動くと、追従対象と視線の間で画像偏差が生じる。すなわち、画像偏差は追従対象と頭部・カメラの回転角度の変化に影響を受ける。そこでまず、画像偏差と追従対象と頭部・カメラの回転角の関係式を出す。この関係式により頭部・カメラの回転角度の変化に伴い画像偏差がどのように変わるかが算出できる。あとは画像偏差によって頭部とカメラの動きが最小になるようにそれぞれの角度を求めていけば、頭部とカメラの効率のよい動作の答えが出てくると考えた。以下の項では具体的な計算方法を説明する。

## 5. 2 画像偏差の数式化



< Fig.15: Calculate equation of declination >

画像偏差を  $Z_s = f(\theta_h, \theta_e, x_t, z_t)$  と表す計算をしていく。まず視線を  $l_x$  とし、カメラの初期位置を原点とする座標系  $\Sigma_l$  をとる。そして、座標系  $\Sigma_l$  における追従対象の座標  $lT = (l_x_t, l_z_t)$  を求めていく。絶対座標系での追従対象の座標を  $T = (x, z)$  とすると、

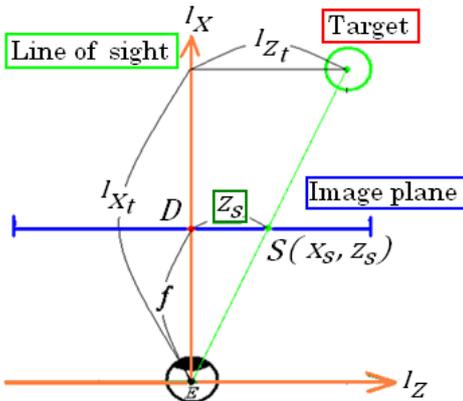
$$\begin{bmatrix} l_z_t \\ l_x_t \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(-\theta_e) & -\sin(-\theta_e) & 0 \\ \sin(-\theta_e) & \cos(-\theta_e) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(-\theta_h) & -\sin(-\theta_h) & -z_e \\ \sin(-\theta_h) & \cos(-\theta_h) & -x_e \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_t \\ x_t \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} z_t \cos \alpha + x_t \sin \alpha - \sqrt{x_e^2 + z_e^2} \sin \beta \\ -z_t \sin \alpha + x_t \cos \alpha + \sqrt{x_e^2 + z_e^2} \sin \gamma \\ 1 \end{bmatrix} \quad (19)$$

ただし

$$\alpha = \theta_h + \theta_e, \beta = \theta_h + \tan^{-1}\left(\frac{z_e}{x_e}\right), \gamma = \theta_h - \tan^{-1}\left(\frac{x_e}{z_e}\right)$$

ここで、座標系  $\Sigma_l$  を中心に考えていく。Fig.15 を座標系  $\Sigma_l$  中心に記述したものが下の Fig.16 である。



< Fig.16: Coordinate axes of line of sight >

Fig.16 から  $l_z_t : Z_s = l_x_t : f$  の相似の関係が分かる。この時、 $f$  はカメラの焦点距離となり  $f = 1$  とする。以上のことから画像偏差を数式化すると、

$$Z_s = \frac{l_z_t}{l_x_t} \quad (20)$$

となり、式(19)、(20)から

$$Z_s = \frac{z_t \cos \alpha + x_t \sin \alpha - \sqrt{x_e^2 + z_e^2} \sin \beta}{-z_t \sin \alpha + x_t \cos \alpha + \sqrt{x_e^2 + z_e^2} \sin \gamma} \quad (21)$$

が導出できる。

## 5. 3 協調動作目標角導出

前項で示した画像偏差の数式

$Z_s = f(\theta_h, \theta_e, x_t, z_t)$  を用いて、頭部とカメラが微小に動いた場合を考慮する。この時の追従対象の座標変化が無いものとする、

$$\delta Z_s = \frac{\partial f}{\partial \theta_h} \delta \theta_h + \frac{\partial f}{\partial \theta_e} \delta \theta_e \quad (22)$$

となる。さらに式(22)を微小時間  $\delta t$  で微分すると、

$$\dot{Z}_s = \frac{\partial f}{\partial \theta_h} \dot{\theta}_h + \frac{\partial f}{\partial \theta_e} \dot{\theta}_e \quad (23)$$

となる。

ここで、画像上の追従対象を画像中心に速度  $v_z$  で移動させるとする、Fig.16 で説明すると点  $S$  から  $D$  への速度を  $v_z$  とし追従対象を画像中心で捕捉しようとしていることになる。そのためには、

$$v_z = \begin{bmatrix} \frac{\partial f}{\partial \theta_h} & \frac{\partial f}{\partial \theta_e} \end{bmatrix} \begin{bmatrix} \dot{\theta}_h \\ \dot{\theta}_e \end{bmatrix} \quad (24)$$

として、 $\dot{Z}_s \rightarrow v_z$  となるような  $(\dot{\theta}_h, \dot{\theta}_e)$  をもてることになる。さらに、 $(\dot{\theta}_h, \dot{\theta}_e)$  の 2 乗ノルムを最小にするためには擬似逆行列を用いて、

$$\begin{bmatrix} \dot{\theta}_h \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial \theta_h} & \frac{\partial f}{\partial \theta_e} \end{bmatrix}^+ v_z \quad (25)$$

の計算をすればよい。式(25)をまとめたものが以下の式(26)となる。

$$\begin{bmatrix} \dot{\theta}_h \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} v_z \frac{\partial f}{\partial \theta_h} & v_z \frac{\partial f}{\partial \theta_e} \\ \left(\frac{\partial f}{\partial \theta_h}\right)^2 + \left(\frac{\partial f}{\partial \theta_e}\right)^2 & \left(\frac{\partial f}{\partial \theta_h}\right)^2 + \left(\frac{\partial f}{\partial \theta_e}\right)^2 \end{bmatrix}^T \quad (26)$$

シミュレーションに質量の要素を加えるために、式(26)に重み要素  $M_h, M_e$  をくわえ、

$$\begin{bmatrix} \dot{\theta}_h \\ \dot{\theta}_e \end{bmatrix} = \begin{bmatrix} M_h v_z \frac{\partial f}{\partial \theta_h} & M_e v_z \frac{\partial f}{\partial \theta_e} \\ \left(\frac{\partial f}{\partial \theta_h}\right)^2 + \left(\frac{\partial f}{\partial \theta_e}\right)^2 & \left(\frac{\partial f}{\partial \theta_h}\right)^2 + \left(\frac{\partial f}{\partial \theta_e}\right)^2 \end{bmatrix}^T \quad (27)$$

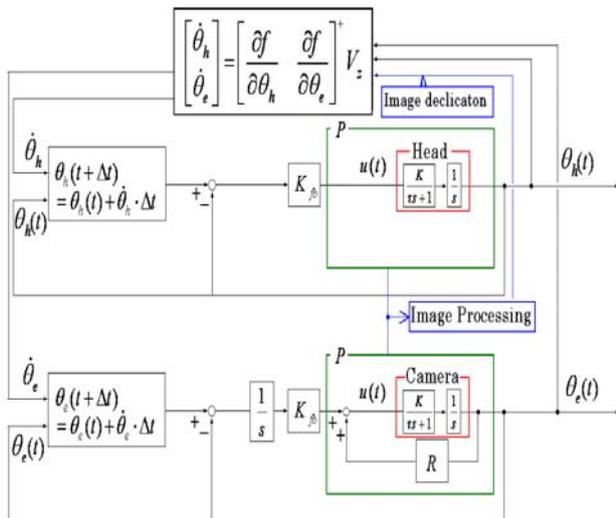
とする。

以上の結果から  $(\dot{\theta}_h, \dot{\theta}_e)$  が求められた。これを用いて頭部とカメラの目標角  $(\theta_h, \theta_e)$  を求め、制御シミュレーションを行う。

## 5.4 シミュレーション

### 5.4.1 追従対象が静止している場合

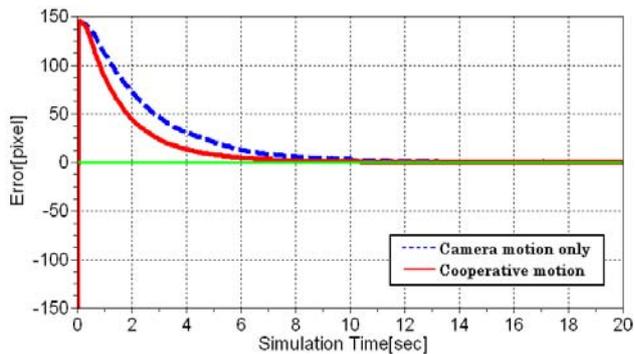
前項で求めた頭部とカメラの目標角を目標入力としてシミュレーションを行った。シミュレーション時の回路を以下の Fig.17 に示す。



< Fig.17:Block Diagram of cooperative motion >

シミュレーションは以下の条件で行った。

- (1) 追従対象は絶対座標系(2.0,0.5,0.6)に固定。
  - (2) 画像偏差を0にする画像平面上の追従対象の理想速度 $v_z$ は画像偏差を1secで補正する値。
- シミュレーション結果は以下の通りである。



<c>Camera image declination  
< Fig.18: Simulation results >

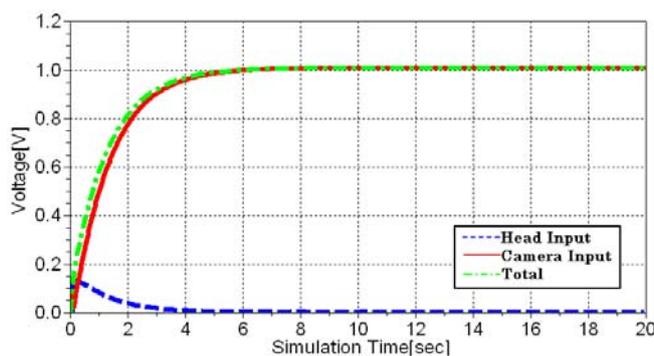
まず, Fig.18 の結果を見ると協調動作<a>の方がカメラ動作のみ<b>よりも制御入力が小さくなった。それに対し, 画像偏差<c>を見ると協調動作の方が素早く追従対象をカメラ画像中心で捕捉できることが確認できる。よって, 静止している追従対象に対しては頭部とカメラの協調動作は有効に働くことが実証できた。

### 5.4.2 追従対象が動作している場合

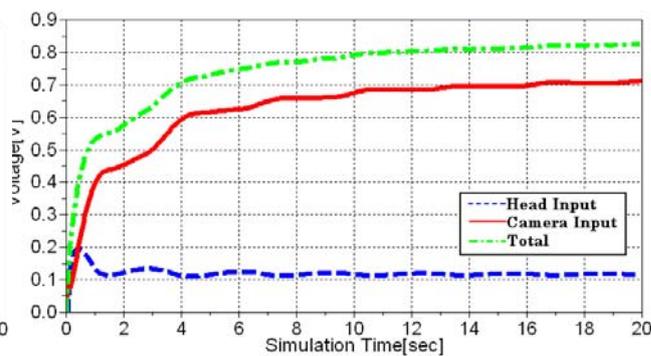
追従対象を動作させた場合も頭部とカメラの協調動作による制御が有効か検証してみた。シミュレーション条件を以下に示す。

- (1) 用いる回路は Fig.17 と同じ回路
- (2) 追従対象は絶対座標軸(2.0,0.5, 0.6cos(t))  
ちなみに t はシミュレーション時間。
- (3) 理想速度 $v_z$ は画像偏差を 0.5sec で補償する値。

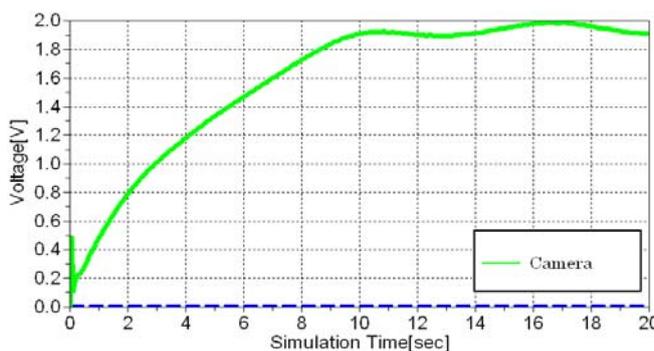
シミュレーション結果は以下の通りである。



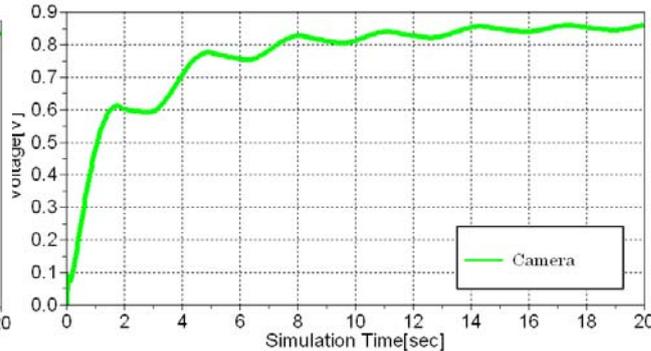
< a > Input of Cooperative Motion



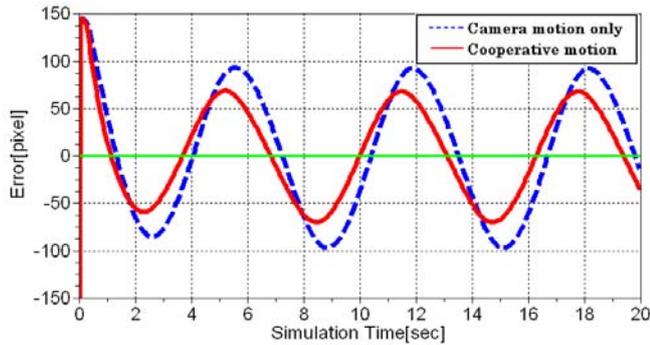
< a > Cooperative Motion



< b > Input of Camera Motion Only



< b > Camera Motion Only



< c>Camera image declination  
<Fig.19: Simulation results>

Fig.19<a><b>は各シミュレーション時間での制御入力の平均値である。この結果を見ると、協調動作<a>の方がカメラのみの制御<b>より、各時間における電圧の消費がわずかに小さいことが分かる。また画像偏差<c>の比較においては協調動作の方が全体的に画像偏差が小さく、追従性能が高いことを示した。動作している追従対象に対しても、頭部とカメラの協調動作はカメラのみの制御よりも制御入力が小さく追従性能が高いことが実証でき、効率よい動作で追従対象を捕捉していることが分かった。

## 6 おわりに

本研究では、頭部に2つのカメラを搭載したヒューマノイドロボットのカメラ制御について2つのシミュレーションを行った。

まず1つが前庭動眼反射のシミュレーションである。このシミュレーションにおいては、人の小脳内でも行われているフィードバック誤差学習の制御をカメラの制御に利用した。この制御手法によりロボット頭部の動作に応じて2つのカメラで追従対象を素早く捕捉することができた。

次に、カメラと頭部に追従対象を捕捉するための目標信号を、頭部とカメラの速度の2乗ノルムの最小値を求めて生成し、ロボット頭部とカメラを制御する協調動作の制御をおこなった。この手法自体は、カメラだけの制御よりも消費電圧が少なく、追従性能が高いことをシミュレーションにより実証した。

今後の課題としては、協調動作の手法とフィードバック誤差学習の手法の融合が挙げられる。カメラの制御はフィードバック誤差学習により、目標信号と出力がほぼ一致している。しかし、頭部の制御に至っては目標信号と出力の誤差が大きくなってしまふことがある。そこで、まず頭部の制御方法を検討し直し、協調動作に有効な制御システムの構築を行っていく必要があると考えられる。

## 参考文献

- [1] 川人光男：脳の計算理論，産業図書，1996.
- [2] A.Miyamura and H.Kimura, Stability of Feedback Error Learning Scheme, System & Control Letters, Vol.45, pp.303-316, 2002.
- [3] E.Muramatu and K.Watanabe, Feedback Error Learning Control without Recourse to Positive Realness, IEEE Transactions on Automatic Control, Vol.49, No.10, pp.1762-1767, 2004
- [4] 柴田：脳科学と眼球運動制御とヒューマノイド，日本ロボット学会誌，Vol.19, No.5, pp.569-573, 2001
- [5] 松田隆夫：視知覚，培風館，1995
- [6] 橋本，小林：OpenGLによる3次元CGアニメーション，オーム社，2005
- [7] 島田明：モーションコントロール，オーム社，2004