２台のカメラを用いた手話動作の３次元取得とその表示に関する研究

大嶋崇之[*1], 戸崎哲也[*2]

Three Dimensional Data Acquisition of Sign Language using Two Digital Video Cameras and Visualization

Takayuki OHSHIMA[*1], Tetsuya TOZAKI[*2]

ABSTRACT

In this study, we propose the three dimensional data acquisition method for the sign language using two digital video cameras. And we visualize its data based on the CG animation. Our 3D data acquisition method consists of four steps. In the first step, we get the movie data sets of the sign language using two video cameras settled on different two directions. In the second, we chase the control points pasted on the human joints by estimating the optical flow. In the third, we calculate the three dimensional coordinate of the control points based on the DLT method, and the last is the visualization of these data on the computer display in order to enable to learn the sign language. We reconstructed 12 words of sign language, and we certified that it is possible to make the sentence of sign language and animate its by assembling these reconstructed words.

Keywords: sign language, tree-dimensional data acquisition, DLT, animation

1. はじめに

最近のコンピュータ技術の進展に伴い，人物や物体の動きをディジタルデータとして記録する技術が開発されてきた．この技術はモーションキャプチャと呼ばれ，スポーツ分野，医療・リハビリテーション分野，映画・ゲームなどのエンターテイメント分野等で広く利用されている．このような技術を手話学習に応用することで，キャプチャされた手話動作をコンピュータグラフィクスを用いて可視化をすることができ，直感的に分かりやすい動作解析が可能となる．そのため，感覚的な学習が可能となるため，学習効率の向上に期待が持てる．

現在よく利用されている手話の教材や辞書においては，手話動作の絵や写真を掲載し，その解説文を併記する表現方法が一般的である．複雑な動作の場合は，複数の角度から見た絵を同時に載せてある [(1)]．また，WEB 上で手話の動作を動画で紹介しているサイトも存在するが，定型的な文章の手話動作を紹介しているのがほとんどで，必ずしも学習者の要望に答えられているとは言いがたい面もある．また，単語と単語の区切りが分かりにくい場合がある．

そこで本研究では，3 次元モーションキャプチャ技術とコンピュータグラフィクスを用いて手話の辞書作成および手話学習システムの開発を目的とする．これは，手話文章を構成する単語の動作を計算機で 3 次元化して辞書データとして保存する．学習時には学習者が必要な単語を辞書データから選択し，それらを連結して表示することで学習効率の向上を図るものである．辞書データを作成するための手話動作の 3 次元復元は次の手法に基づいて行う．まず，手話の動作を 2 台のカメラで撮影し，次にその動画からオプティカルフロー推定に基づいて自動的に特徴点の追跡を行う．さらに，得られた特徴点の 2 次元座標から DLT(Direct Linear Translation) 法を用いて 3 次元座標を算出する．最後に，得られた結果を 3D をディスプレイ上でアニメーション表示して学習者に提示する．アニメーションには任意の角度からの観察や拡大・縮小機能を設け，学習者の意向に沿った表示を実現することを目指す．

本報告の構成は次のとおりである．2 章では我々が提案する手話動作の 3 次元化手法を，3 章では手話動作の 3 次元取得とその表示例を示し，4 章で本報告のまとめを行う．

2. 手法

2.1 手話動作の撮影　手話動作は，上半身の大まかな動きと手指の動きの 2 つに分けて行う．上半身の動きは，頭，首，胸部，腹部，両肩，両肘，両手首，両手

[*1]電気電子工学専攻 2 年
[*2]電子工学科　准教授

(a) 左カメラ　　(b) 右カメラ

図 1: 撮影環境

の親指の付け根と小指の付け根の合計 14 点に蛍光テープを貼付けて特徴点とし，2 台のデジタルビデオカメラを用いて撮影する．使用したデジタルビデオカメラは，SANYO 社製 Xacti DMX-CG110 である．手話動作者は椅子に座り，上半身の動作が写るようにカメラを設置する．撮影時の様子を図 1 に示す．(a) が左のカメラから撮影した画像であり，(b) が右のカメラから撮影した画像である．

手指の動きに関しては，定型の動作をあらかじめ撮影してデータ化しておき，手話単語に応じて選択することとする．この手指の定型データのためには，手の各関節と手首の合計 20 点を特徴点として 3 次元化する．

2.2 特徴点の追跡　撮影で得られた左右それぞれの動画から各フレーム上の特徴点の追跡を行い，それぞれのカメラ画像上での 2 次元座標を取得する．特徴点の追跡はオプティカルフローを推定することによって行う．オプティカルフローとは時間的に連続する画像中での物体の動きをベクトルで表したものである．ここではこのオプティカルフローを推定する手法として局所勾配法 [2] を用いる．局所勾配法とは，物体上の点の明るさは移動後も変化しないという仮定から時空間微分とオプティカルフローとの関係式を導出し，それを利用して対象の動きを推定するものである．

画像上の時刻 t におけるある点 (x, y) の輝度を $I(x, y, t)$ とし，微小時間 δt 後の移動点を $(x+\delta x, y+\delta y)$ とする．物体上の輝度は不変であると仮定すると以下の式が成り立つ．

$$I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) \tag{1}$$

式 (1) の右辺をテイラー展開すると，

$$I(x, y, t) = I(x, y, t) + \delta x \frac{\partial I}{\partial x} + \delta y \frac{\partial I}{\partial y} + \delta t \frac{\partial I}{\partial t} + e \tag{2}$$

ここで e は $\delta x, \delta y, \delta t$ に関する 2 次以上の高次項であり微小であるとして無視する．両辺を δt で割ると，

$$\frac{\delta x}{\delta t}\frac{\partial I}{\partial x} + \frac{\delta y}{\delta t}\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} = 0 \tag{3}$$

δt の極限として $\delta t \to 0$ とすると，

$$\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0 \tag{4}$$

ここでオプティカルフローの速度成分 (u, v)，空間的な輝度勾配 I_x, I_y，時間的な輝度勾配 I_t を

$$u = \frac{dx}{dt}, v = \frac{dy}{dt}, I_x = \frac{\partial I}{\partial x}, I_y = \frac{\partial I}{\partial y}, I_t = \frac{\partial I}{\partial t}$$

とおくと次のように表せる

$$I_x u + I_y v + I_t = 0 \tag{5}$$

この式はオプティカルフローの拘束方程式と呼ばれる．

しかし，式 (5) だけではオプティカルフローを一意に決定することができない．そこで，局所勾配法では同一物体の濃度パターン上の局所領域においては，オプティカルフローは一定であるとする．すなわち，

$$\frac{\partial(u, v)}{\partial x} = \frac{\partial(u, v)}{\partial y} = 0 \tag{6}$$

と仮定する．式 (5) と式 (6) の 2 つの拘束式から誤差 E は次式で表せる．

$$E = \sum_x \sum_y (I_x(x, y, t)u + I_y(x, y, t)v + I_t(x, y, t))^2 \tag{7}$$

上式の評価関数が最大となるようにオプティカルフローの速度成分 u, v を決定する．このオプティカルフローを特徴点の移動ベクトルとする．

2.3 3 次元座標の推定　特徴点の追跡によって得られた 2 方向からのカメラ画像上の座標を用いて，各特徴点の実空間上の 3 次元座標をそれぞれ推定する．ここでは，DLT 法 (Direct Linear Transformation Method) を用いる．これは，複数のカメラから得られた画像の制御点と呼ばれる既知の 3 次元座標からカメラキャリブレーションを行い，3 次元座標を算出する手法である

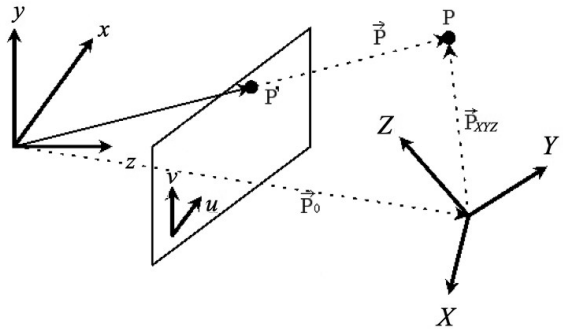


図 2: DLT 法における座標系の関係

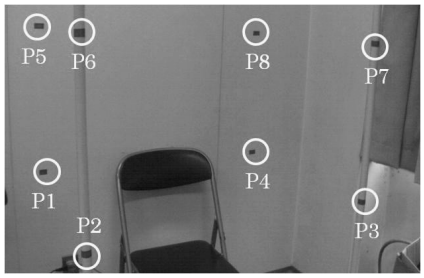


図 3: 設定した制御点

表 1: 設定した制御点の座標 [cm]

点	x	y	z
P_1	0	70	0
P_2	0	70	70
P_3	100	70	70
P_4	100	70	0
P_5	0	120	0
P_6	0	120	70
P_7	100	120	70
P_8	100	120	0

(3),(4)．この手法はカメラの設置に関する制約が少なく正確な計測が可能であるという利点がある．DLT 法における座標系の関係を図 2 に示す．実空間座標系 XYZ と，カメラのレンズ中心に原点を持つカメラ座標系 xyz がある．また，カメラのフィルム面に固定された座標系 uv はカメラ座標系 xyz を平行移動した座標系で，xy 座標軸と uv 座標軸は平行である．特徴点の座標はこの実空間座標系 XYZ で表される．また，実空間座標の点 P(X, Y, Z) がカメラのフィルム面の点 P'(u, v) に投影されている．DLT 法の基本式は

$$u = \frac{A_1X + A_2Y + A_3Z + A_4}{C_1X + C_2Y + C_3Z + 1} \tag{8}$$

$$v = \frac{B_1X + B_2Y + B_3Z + B_4}{C_1X + C_2Y + C_3Z + 1} \tag{9}$$

である．ここで A_1 から C_3 はカメラ定数と呼ばれる定数である．そして式 (8)，式 (9) を展開すると，

$$u = A_1X + A_2Y + A_3Z + A_4 - uC_1X - uC_2Y - uC_3Z \tag{10}$$

$$v = B_1X + B_2Y + B_3Z + B_4 - vC_1X - vC_2Y - vC_3Z \tag{11}$$

となる．この式 (10) と式 (11) に制御点の 3 次元座標と画像座標の組を代入することで，A_1〜C_3 のカメラ定数を求めることができる．本研究では，手話動作者を囲む空間上に 8 点の制御点（P_1〜P_8）を設定した．その制御点の様子を図 3 に示す．また，それらの実空間上の座標を表 1 に示す．

カメラ定数が全て求まると，式 (10) と式 (11) は変数が X, Y, Z, u, v の方程式となり，ある点の 2 方向からの 2 次元座標を与えることで，その点の 3 次元座標が求まる．今，左右のカメラ定数をそれぞれ A_{L1}〜C_{L3}，A_{R1}〜C_{R3} とすると，フィルム面上の座標と実空間上の座標は，式 (12) のように行列表現できる．

$$\begin{bmatrix} A_{L1} - C_{L1}u_L & A_{L2} - C_{L2}u_L & A_{L3} - C_{L3}u_L \\ B_{L1} - C_{L1}v_L & B_{L2} - C_{L2}v_L & B_{L3} - C_{L3}v_L \\ A_{R1} - C_{R1}u_R & A_{R2} - C_{R2}u_R & A_{R3} - C_{R3}u_R \\ B_{R1} - C_{R1}v_R & B_{R2} - C_{R2}v_R & B_{R3} - C_{R3}v_R \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} u_L - A_{L4} \\ v_L - B_{L4} \\ u_R - A_{R4} \\ v_R - B_{R4} \end{bmatrix} \tag{12}$$

また，方程式を $Ax = b$ であるとすると，

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix},$$

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, b = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{bmatrix}$$

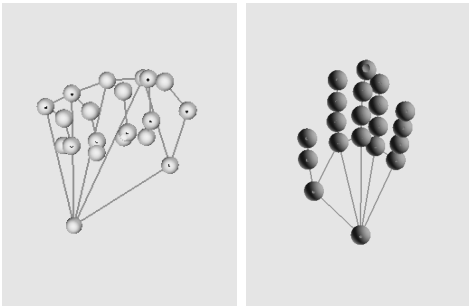
と表現できる．これを擬似逆行列を用いることで，

$$A^TAx = A^Tb$$

$$x = (A^TA)^{-1}A^Tb \tag{13}$$

となり，x を求めることができる．

2.5 手指の表現　手指は，非常に繊細な動きをするため，体の大まかな動きを捉える前述の手法は適応できない．そこで，手話に用いられる手の型は限定的であると仮定し，その定型パターンをあらかじめ作成しておき，手話の単語に応じてそのパターンを選択して上体の動作と合成させることで表現することとする．定型パターンの作成においては，対象とする手の型を 2 方向から静止画撮影し，指や手首の間接 20 点を対話的に選択し



(a) 右手のモデル　(b) 左手のモデル

図 4: 手指モデル

て DLT 法に基づいて 3 次元化を行う．上体動作との合成は，次のステップから成る．

Step1) 上半身のデータからその特徴点の 1 つである手首，親指の付け根，小指の付け根の 3 点を選択し，手首を起点として親指の付け根，小指の付け根へ向かう 2 つのベクトルを導く．そして，これらのベクトルの外積より，手首を始点として両ベクトルに垂直なベクトルを導き，$\vec{n}_1$ とする．

Step2) Step1) と同様の手法で手指のデータの同じ特徴点から得られる 2 つのベクトルより手首を始点としそれらに垂直なベクトル $\vec{n}_2$ を求める．

Step3) $\vec{n}_2$ の始点が $\vec{n}_1$ と一致するように平行移動し，その後，この始点を中心に $\vec{n}_1$ と向きが一致するように回転させる．

Step4) $\vec{n}_2$ を軸とする回転を加え，それぞれの手首と親指を結ぶベクトルの向きを一致させる．

尚，特徴点の座標はどちらも DLT 法を用いて実空間上のものに置き換えられているので，ベクトルの大きさは考慮していない．

また，ある単語から別の単語へ移行中の手指の動きは，移行前と移行後の各点の 3 次元位置を移動フレーム数に応じて内分させることで表現する．図 4 に，手指のモデルの 1 例を示す．これは，単語「成功」に使用する定型パターンであり，(a) が右手の型である軽く握った様子を，(b) が左手の型である手を広げている状態を表現している．

2.6 表示　算出された各間接の 3 次元座標に基づいて，隣り合う間接同士を連結して表示する．表示には，3 次元 CG を用い，連続する単語を動的に表現することで，学習者が直感的に理解しやすいことを念頭に表示システムを整える．ディスプレイ上での手話動作はヒューマノイド型ロボットをモデル化したものが行い，さらに任意に角度を変更出来る機能や，拡大・縮小機能をインタフェースとして盛り込んでいる．これらインタフェースの作成には，標準的な 3D グラフィクスの API である OpenGL と glui を使用した．つまり，オープンプラットフォームで使用可能である．また，複数の単語を選択した順にスムースにアニメーションすることで，文章を表現する機能も有している．これは，単語数に制約されるものの，学習者が任意の文章を作成して理解することが可能であり，学習の幅が広がるものと期待される．

3. 結果と評価

現時点で，12 個の単語 [5] の動作をデータ化した．その内訳は，名詞として「目」,「成功」,「テーマ」,「本当」,「あなた」,「みんな」，疑問詞として「〜ですか？」,「どっち？」，動詞として「わかる」,「見る」，形容詞として「美味しい」，感謝の意として「ありがとう」である．また，手指のパターンは”握る”，”開く”，”人差し指を立てる”，”人差し指と中指を立てる”の 4 パターンを定型化した．

表 2: 制御点のカメラ座標

制御点	左カメラ	右カメラ
P_1	(50,401)	(293,362)
P_2	(139,574)	(70,478)
P_3	(777,461)	(699,568)
P_4	(519,359)	(751,380)
P_5	(42, 58)	(270, 90)
P_6	(125, 35)	(26,108)
P_7	(802, 84)	(686, 15)
P_8	(519,359)	(735, 42)

表 3: カメラ定数

カメラ定数	左カメラ	右カメラ
A_1	5.666606	2.851161
A_2	-0.255159	-0.583853
A_3	0.543546	-3.355665
A_4	65.016773	321.963899
B_1	0.468383	-0.570596
B_2	-6.505611	-5.173400
B_3	-0.459555	-0.278561
B_4	830.568753	706.388155
C_1	0.002420	-0.002038
C_2	-0.000849	-0.000549
C_3	-0.004877	-0.004069

今回撮影した環境下でカメラ定数を求めるために使用した左右両画像からの既知の点の座標を表 2 に示す．表 2 中の制御点は図 3 に示すそれである．これらの値を用いて式 (10)，(11) より導かれたカメラ定数を表 3 に示す．本章で示す結果は，すべてこのカメラ定数より得られた 3 次元座標を用いた．

3.1 特徴点の追跡　実際の動作の動きおよび特徴点追跡の様子を図 5 に示す．緑の点が特徴点を追跡している様子である．被写体に付けた特徴点となるマーカを追従していることから，局所勾配法による特徴点追跡が行えていることが確認できる．しかし，特徴点同士が重なったり，特徴点がカメラの死角になる等の理由から特徴点を誤って追跡してしまうケースが生じる．このような場合，そのまま．3 次元化すると不自然な動きになってしまう．それを防ぐために，まず 1 フレーム目の 3 次元座標から隣り合う特徴点間の距離をあらかじめ算出する．次に，2 フレーム目以降のこれらの距離と 1 フレーム目の距離を比較し，誤差が 20 %以上になる場合を誤追跡とする．そして，誤追跡となったフレームは前後のフレームの 3 次元座標値から線形補間を行って補正することで，誤追跡の影響を軽減させた．

図 5: 特徴点追跡の様子

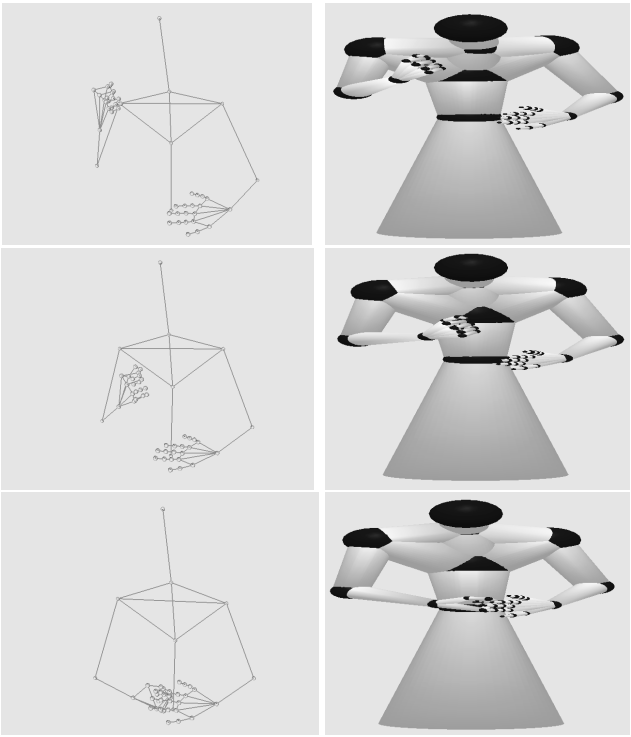


図 6: アニメーション

3.2 手話動作の 3 次元復元　2 方向からの特徴点の 2 次元座標を DLT 法によって 3 次元座標に復元した．その一例を図 6 に示す．表現している単語は「成功」であり，実際の動きは図 5 のものである．また，手指のモデルとして図 4 に示すものを合成している．図左が隣り合う特徴点間を線で表現したワイヤーフレームモデル，右がその立体表現である．上から順に時系列的に表現している．これらの図より，手話動作者の特徴点を追跡して 3 次元化出来ていることが分かる．また，上半身を立体的に表現するすることで直感的に動作が理解しやすくなることが分かる．

図 7 には，今回作成した手話単語から数語を選択して一連の文章にしてアニメーション化した結果を示す．選択した単語は,「私」,「あなた」,「目」,「見る」,「本当」,「どうか」,「分かる」であり，これらを連結することでディスプレイ上では,「私は，あなたの目を見れば本当かどうか分かる.」という文章が表現できる．ここで，各単語間の連結は単語の最終フレームと次の単語の 1 フレーム目のそれぞれの特徴点を線形補間をすることで実現し，動きをスムースに表現した．また，3D アニメーションで表示するユーザインタフェースは簡単な操作で拡大・縮小，任意角度からの表示を可能にしており，背後からの観察等も可能である．

3.3 他手法との比較　Google で”手話学習”をキーワードにして検索し，ヒット件数上位の 2 手法を学習効率の高いものであると仮定して本手法と比較を行った．比較対象とした手話学習システムは，文献 (6),(7) であり，比較項目として ” 単語数 ”, ” 表示方法 ”, ” 見やすさ ”, ” 文章作成の可否 ”, ” 任意角度表示の可否 ” を設定して，主観的な評価を行った．その結果を表 4 に示す．文献 (6) と文献 (7) の両方のシステムに共通しているのは，実際に人が手話をしている様子をそのまま紹介していることである．文献 (6) は動画をそのままで表示していたが，動画ではデータが大きいためスムーズな再生ができていない場合があった．また，文章を作成する機能はない．文献 (7) は，1 つの動作を数枚の静止画に分割し，それを連続で表示することで手話動作を表示していた．こちらのシステムでは複数の単語を選択することで，連続して手話動作を表示することが可能であるが，1 つの動作に対する静止画の数が少ないため，動作の流れが分かりづらい．それに対して，本システムは 1 つの動作のデータ量が 100〜300kB と動画に比べて非常に少なく，複数の単語を選択して文章作成を可能にし，さらにスムーズな再生が可能である．しかし，現在までのところ単語数が少ないため文章の作成に制約がある．

表示は CG アニメーションを用いているため，同一の人物が撮影する必要がなく，撮影環境が異なってもアニメーションで表示する際には影響しないという利点もある．実際に人が動作をしている様子を動画で紹介している文献 (6) に比べると，本システムの 3D アニメー

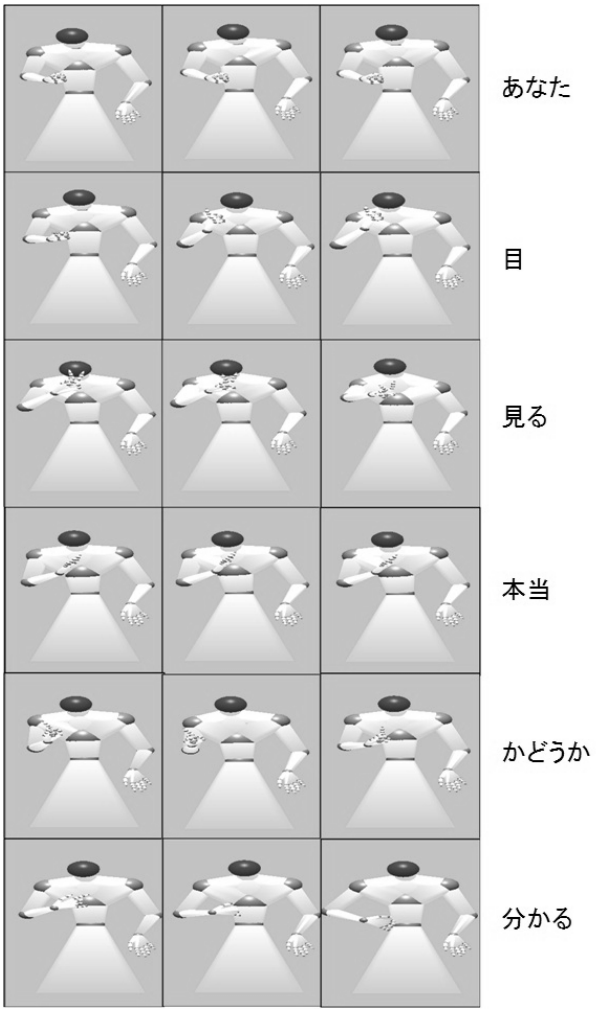


図 7: 文章の表現例

ションではややぎこちない部分が目立ち，直感的な分かりやすさと言う点では劣る点があった．しかしながら，本システムは文献 (6),(7) には無い任意の角度での表示や拡大・縮小などの任意性を有しているため，手話の学習に有効であると考えられる。今後は，単語数を増やすことにより，任意の文章作成を可能にする必要がある．また，操作性についても単語の選び方や，動作とともに文章も表示する機能を追加する等の検討が必要である．

4. まとめ

本研究では手話の動作を 3 次元に復元して，手話の学習システムを開発することを目的とした．12 種類の手話単語に対して 3 次元復元を行った．特徴点追跡は局所勾配法を用いたオプティカルフローを推定することで可能であることが確認できた．また，DLT 法を用いて 2 方向からの 2 次元座標から 3 次元座標を推定できることが確認できた．さらに，動作の 1 フレーム目の各特徴点間の距離を利用して補正をすることでその誤追跡の影響を軽減することができた．得られた 3 次元データを 3D アニメーション化することで，直感的な手話学習が可能になり，任意角度での表示等を機能として追加することで，学習の幅が広がるがことが確認出来た．これらより，本システムは新しい手話学習支援システムとして利用することに期待出来るものと考えられる．

表 4: 手話学習システムの比較

	本システム	文献 (6)	文献 (7)
単語数	少ない	少ない	多い
表示方法	アニメーション	動画	静止画
見やすさ	○	○	×
文章作成	△	×	○
任意角度表示	○	×	×

今後の課題としては，特徴点追跡の際，カメラの死角となり特徴点を追跡できなくなるケースを軽減させる必要がある．これには，3 台目のカメラを導入して死角となる特徴点を軽減させることを検討している．さらには，より多くの手話単語を 3 次元データ化し信頼性の高い手話学習支援システムへと発展させることを目指す予定である．

参考文献

(1) 例えば,「Weblio 手話辞書」， http://shuwa. weblio.jp.

(2) 三池秀敏，古賀和利：「パソコンによる動画像処理」，森北出版株式会社，pp.133-143，1993.

(3) 前田一真，戸崎哲也：「市販のビデオカメラを用いたスポーツ動作の 3 次元解析」, 平成 20 年電気学会全国大会講演論文集，p.75，2008.

(4) Liang Chen, Charles W. Armstrong and Demetrios D. Raftopoulos, ：”AN INVESTIGATOIN ON THE ACCURACY OF THREE-DIMENSIONAL SPACE RECONSTRUCTION USING THE DIRECT LINEAR TRANSFORMATION TECHNIQUE”, Journal of Biomechanics, Vol. 27, Issue 4, pp.493-500, 1994.

(5) 丸山浩路：「百万人の手話」，ダイナミックセラーズ出版，1980.

(6)「ブラウザ手話学習システム」， http://portal.kumamoto-net.ne.jp/portal-ud/syuwa/m1/.

(7)「手話学習システム」， http://siva.cc.hirosaki-u.ac.jp/usr/koyama/semi/syuwa/syuwa.htm.