長岡技術科学大学大学院
工学研究科修士論文

題　目

OpenPoseを用いた
手話単語認識に関する研究

指導教員　　准教授　杉田　泰則

著者　電気電子情報工学専攻
15312585　川又　亮太

提出期日　　平成31年2月10日

目 次

第 1 章　はじめに

1.1　研究背景

近年，障害者の活躍できる社会づくりが盛んになっており，その一環として聴覚障害者 (ろうあ者) の社会進出も進められている。ろうあ者の主なコミュニケーション手段の一つとして，手話が挙げられる。手話は手指動作と非手指動作を用いて行う視覚言語であり，ろうあ者を中心に用いられている。ろうあ者の社会進出に伴い，手話を普及させる動きも見られるようになった。その一つに，手話を広く使える社会を目指す「手話言語条例」があり，2013 年に国内初の成立を皮切りに，現在では 225 の自治体で成立している。しかし，聴者で手話を理解できる者はまだまだ少なく，ろうあ者が社会生活の中で手話を使える機会は僅かである。

ろうあ者と聴者のコミュニケーションの補助を行う仕事として，手話通訳者が存在する。手話通訳者を介することで，ろうあ者は手話で聴者に意思を伝えることができ，スムーズなコミュニケーションが期待できる。しかし，手話通訳者は職業としての環境が整っておらず，育成の環境も整っていない。このため数が少なく，手話通訳者は慢性的に不足している。

このような背景を受け，手話通訳者に代わって手話通訳を行うことのできる手話認識システムが期待されている。近年，ニューラルネットワークが一般物体認識をはじめ，多くの分野で成功を収めており，手話認識に関しても多くの研究で，優れた成果が報告されている [1][2][3]。例として [1] では，178 種類のドイツ語手話に対して，50 名の手話者による 100 時間以上の動画で学習を行い，82.7% の認識精度を達成している。

ニューラルネットワークでは，学習データが重要とされており，どのようなデータで学習を行うかによって結果が大きく左右される。一般的に，大規模で質の良いデータセットを用いることで，よりよい結果を得られることが知られている。ニューラルネットワークを用いた手話認識は現在までに多く研究され，使用される学習データも様々であり，代表的なものは RGB カメラ，Kinect[4][5][6]，Leap motion[7][8][9] それぞれから得られるデータである。Kinect を用いたものの例として [4] では，12 種類の英語手話に対して，10 名の手話者による 336 個のデータで，95.5% の認識精度を達成しており，Leap motion を用いたもの [7] では，30 種類の英語手話に対して，20 名の手話者による 1200 個のデータで，96.4%

の認識精度を達成している。Kinect，Leap motion は 3 次元空間情報として手指の位置を取得できるため，手話認識に優位であると考えられるが，一般への普及率で RGB カメラに大きく劣る。従って，本論文では RGB カメラによって撮影した動画を使用してデータセットを作成する。

動画は情報量が多く，手話認識に必要な情報を十分有していると考えられる。しかし，情報量の多さ故に，良い結果を得るにはそれ相応のデータ量が必要となり，それに伴うモデル規模の増大，学習の長時間化という欠点がある。また，多くの研究で用いられる動画は，スタジオなどの好条件の下で撮影されているものがほとんどである。このため，実際の環境下では背景などの撮影条件の影響を受けることが危惧される。

1.2 研究目的

学習時のデータサイズの削減と，撮影条件の影響を受けにくい手話単語認識の実現を目的とする。本論文では，OpenPose を用いて抽出した,「鼻」·「両手首」·「手指」の座標情報を学習データとして用いる。座標情報を用いることで，動画を用いる場合に比べ，大幅なデータ量の削減が期待できる。また，背景などの情報を完全に排除できるため，それらの影響を受けることは無い。動画を用いた場合と，提案法とでシミュレーションを行い，比較検討を行うことで提案法の有効性を評価する。

1.3 本論文の構成

本論文の構成は以下の通りである。第 1 章では，本論文の研究背景と目的を述べた。第 2 章では，提案法に必要な要素技術について述べる。第 3 章では，提案法で用いる OpenPose と，それによって得られる座標情報の使用方法について述べる。第 4 章では，実験に必要な条件を示し，動画を用いた場合と，提案法での，手話単語認識の精度などを比較する。また，それらの結果を示し，提案法の有効性を確認する。第 5 章では，本論文のまとめを述べる。第 6 章では，今後の課題を述べる。

第 2 章　ニューラルネットワーク

2.1　畳み込みニューラルネットワーク

畳み込みニューラルネットワーク (Convolutional Neural Network, CNN) とは結合構造を制限することで，結合重みの自由度を減らし，画像に対しての学習を容易にしたネットワークである。2012 年に大規模一般物体認識のコンペティション ImageNet Large Scale Visual Recognition Challenge において，大規模な CNN によるモデルが優勝したことで，CNN，ひいては深層学習へ注目が高まり，現在では幅広い分野での研究に用いられている。

CNN は，畳み込み層とプーリング層と呼ばれる特殊な層を交互に接続した構造を持っており，これを除けば一般的なフィードフォワード型のニューラルネットワークである。

2.1.1　畳み込み層

畳み込み層の基本構造を図 2.1 に示す。

入力として，サイズ S×S で N チャネル (S×S×N) の画像をとるとする。この時，フィルタのチャネル数は入力と等しくなければならないため，N となり，また，サイズは L×L とする。

図中の $*$ は畳み込みを表し，畳み込み層では，上記入力にフィルタを畳みこんでいく。具体的には，入力の各チャネルにフィルタの各チャネルを畳み込み，その結果を全チャネル

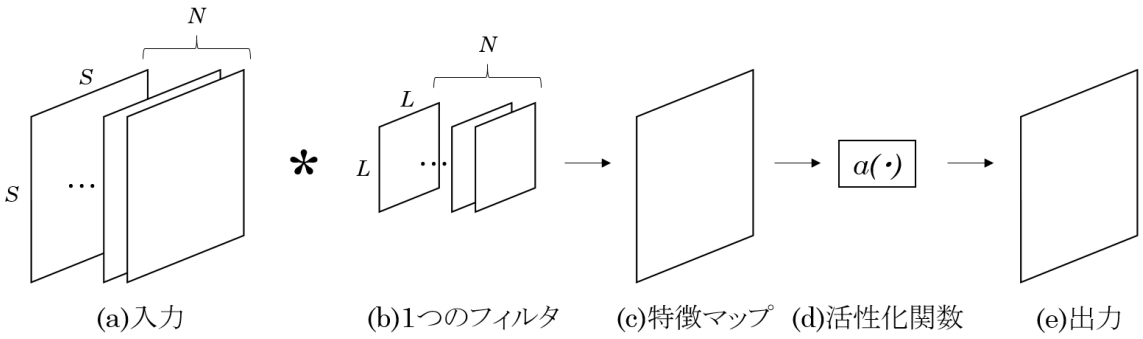


図 2.1　畳み込み層でのフィルタ 1 つに関する計算内容

に渡って加算する。この結果は 1 チャネルの画像の形式をとり，特徴マップと呼ばれる (図 2.1 の (c))。得られた特徴マップは，活性化関数 (図 2.1 の (d)) を経て畳み込み層の出力となる (図 2.1 の (e))。このように，フィルタ 1 つにつき入力と同サイズ，チャネル数 1(S×S×1) の出力が得られる。ここで，フィルタ数を N’ とすると出力の形は S×S×N’ となり，次の層への入力となる。

畳み込み層おけるパラメータはフィルタであり，畳み込み層 1 層のパラメータ数 p_{cnn} は以下の様になる。

$$p_{cnn} = L \times 2 \times N \times N' + N \times N' = L^2 NN' + NN' \tag{2.1}$$

ここで，NN' はバイアスであり，フィルタ 1 枚につき値を 1 つ持つが，フィルタ内のチャネル間で同一とする場合が多く，その場合上式は以下の様になる。

$$p_{cnn} = L \times 2 \times N \times N' + N' = L^2 NN' + N' \tag{2.2}$$

2.1.2 プーリング層

プーリング層の目的は，入力における微小な位置変化に対し頑健にすることである。プーリング層も，畳み込み層と同様に入力に対してフィルタを適用して出力を得る。ただし，プーリング層のフィルタはパラメータを持たず，学習によって変化する重みは無い。また，活性化関数を使用しないのが一般的である。図 2.2 に，フィルタサイズ 3×3，ストライド 2 のプーリングの様子を示す。

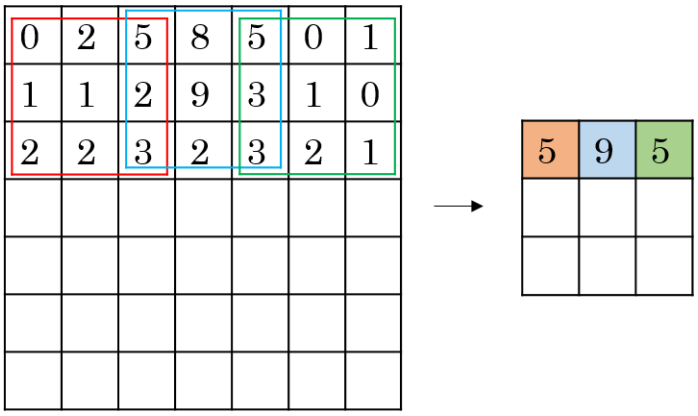


図 2.2 最大プーリング

上図では，フィルタが重なった部分における最大値を出力とする最大プーリングを行っている。ストライドとはフィルタの移動幅のことであり，今回の場合，ストライドが 2 であるため，フィルタは図のように移動していく。この処理を入力のチャネルごとに独立して行う。従って，入出力でチャネル数は変化しない。また，最大値ではなく，平均値を出力とする平均プーリング (average pooling) も存在するが，最大プーリングを用いるのが一般的である。

2.2 Long short-term memory

Long short-term memory(LSTM) とは，Recurrent Neural Netwaork(RNN) の拡張として登場した，時系列データに対するモデルである。LSTM は，中間層を LSTM ブロックに置き換えることで，従来の RNN では学習出来なかった長期依存を学習可能にしている。図 2.3 に LSTM ブロックの構造を示す。

図 2.3 の様に，LSTM ブロックは入力ゲート (Input gate)，出力ゲート (Outout gate)，忘却ゲート (Forget gate)，メモリセル (memory cell) から構成されている。ここで，x_t は現在のタイムステップでの入力，y_{t-1} は 1 つ前のタイムステップでの出力，$R_{in}, R_{out}, R_z, R_{for}$, $w_{in}, w_{out}, w_z, w_{for}, b_{in}, b_{out}, b_z, b_{for}$ は各ゲートのパラメータであり，学習によって変化する重みである。また，図中の破線矢印は前のタイムステップからの入力を表している。入力の次元数を N，メモリセルの次元数を N' とすると，各ゲートの R, w, b のパラメータ数 p_R, p_w, p_b は以下の式で表せる。

$$p_R = p_w = N \times N' \tag{2.3}$$

$$p_b = N' \tag{2.4}$$

従って，LSTM ブロック 1 つ当たりのパラメータ数 p_{lstm} は，以下の様になる。

$$p_{lstm} = (p_R + p_w + p_b) \times 4 = 8(N \times N') + 4N' \tag{2.5}$$

また，σ はシグモイド関数，tanh は tanh 関数の活性化関数である。x_t, y_{t-1} は各ゲートそれぞれに供給され，各重み行列 $R_{in}, R_{out}, R_z, R_{for}, w_{in}, w_{out}, w_z, w_{for}$ と乗算されたのち，加算され，各活性化関数を通過する。

入力ゲートでは，現在の入力 x_t と前の時間の出力 y_{t-1} を，それぞれどの程度メモリセル

に伝搬するかを制御しており，式で表すと以下の様になる。

$$i_t = \sigma(R_{in}y_{t-1} + w_{in}x_t + b_{in}) \tag{2.6}$$

$$z_t = \tanh(R_z y_{t-1} + w_z x_t + b_z) \tag{2.7}$$

メモリセルは過去の情報を保持しており，忘却ゲートによってその保持具合を制御している。メモリセルの状態を c_t とすると以下の様に表せる。

$$f_t = \sigma(R_{for}y_{t-1} + w_{for}x_t + b_{for}) \tag{2.8}$$

$$c_t = \sigma(i_t \odot z_t + c_{t-1} \odot f_t) \tag{2.9}$$

ここで，$\odot$ は要素ごとの積を表している。出力ゲートでは，メモリセルからの出力を制御

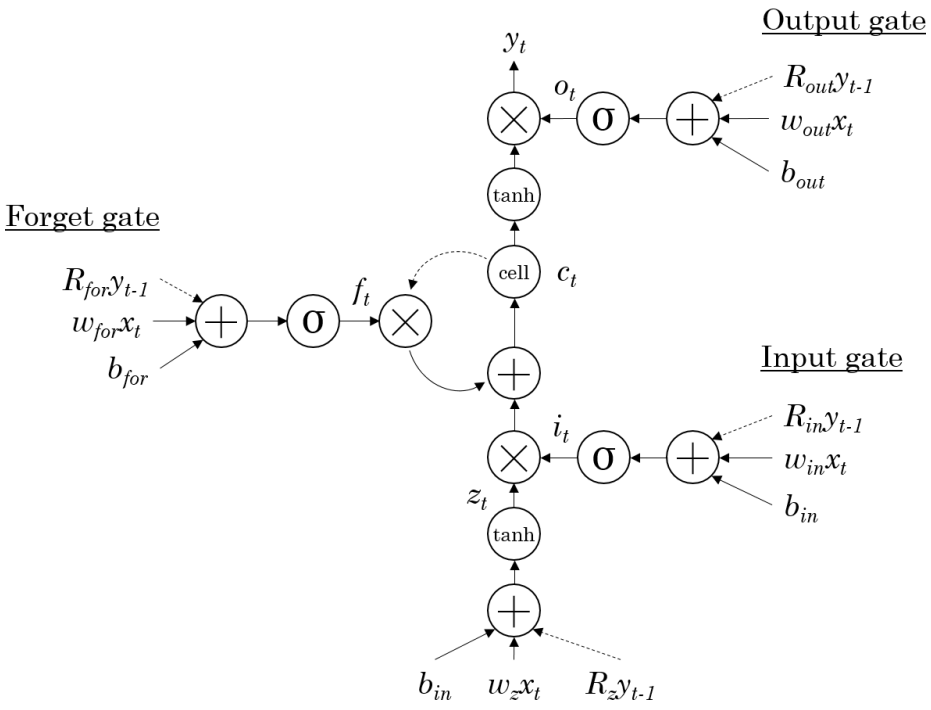


図 2.3　LSTM ブロックの構造

しており，LSTM ブロックの出力は以下の様になる。

$$o_t = \sigma(R_{out} y_{t-1} + w_{out} x_t + b_{out}) \tag{2.10}$$

$$y_t = \tanh(c_t) \odot o_t \tag{2.11}$$

第 3 章　提案法

従来の，動画を用いた手話認識では，動画をそのまま学習データとし，認識モデルへの入力としている。この方法の利点として，画像処理による特徴抽出などが必要無く，end to end の学習が行えることが挙げられる。しかし，特徴抽出を必要としない反面，高い認識精度の達成には非常に多くの学習データが必要となる。また，動画が学習データのため，データ 1 つ 1 つのサイズが大きい。従って，学習の際のデータサイズが膨大になり，それに伴うモデル規模の増大，学習の長時間化が欠点となる。

また，撮影環境が認識精度に影響を与えることが考えられる。手話認識の研究で用いられるデータセットは，スタジオなどの好条件で撮影された動画である場合が多く，実環境にそぐわない条件がみられる。例として，背景が壁のみであったり，手話者が常に画面の中央にいるなどが挙げられる。実環境下でこれらが変化したとき，無視できない影響を及ぼす懸念がある。

本章では，上記の従来手法における問題点を考慮し,「鼻」·「手首」·「手指」の座標情報による手話認識を提案する。

3.1　OpenPose

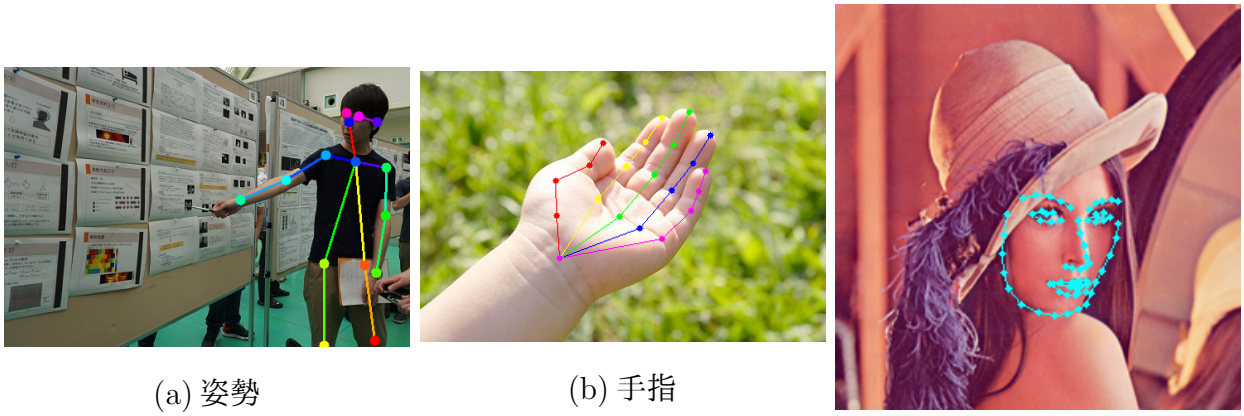
体の各部位の座標情報を取得する為，提案法では OpenPose を用いる。OpenPose とは Zhe らの人物姿勢推定手法 [10] を用いたライブラリである。Web カメラなどの簡単なカメラで，リアルタイム姿勢推定を行うことができ，その精度は非常に高い。また，姿勢推定に加え手指，顔の特徴点の検出も行うことができる。

OpenPose によって得られる特徴点は以下の通りである。

また，OpenPose を用いて，姿勢，手指，顔の特徴点を検出し，元の画像上に描画したも

表 3.1　OpenPose で得られる座標

	検出される部位	総座標数
姿勢	鼻　首　右肩　右肘　右手　左肩 左肘　左手　右腰　右膝　右足　左腰 左膝　左足　右目　左目　右耳　左耳	18
手指	手首　各指の付け根　各指の第一関節　各指の第二関節　各指先	42
顔	眉　目　鼻　口	69



(a) 姿勢　(b) 手指　(c) 顔

図 3.1　姿勢，手指，顔の特徴点

のを図 3.1 に示す。

3.2　座標について

3.2.1　使用する座標

OpenPose によって得られる座標は上述の通りである。提案法ではそれらの中から,「鼻」・「両手首」・「手指」の座標 45 点を学習データとして用いる。手話は手の動きと手の形で意味を表す。従って，手の位置情報として両手首の座標 (2 点) と，手の形の情報として，得られる全ての手指の座標 (42 点) を採用した。また，手の位置は上半身のいずれかであることがほとんどで，中でも顔の周りであることが多いため，顔と手の相対位置が重要であると考え，顔の中心である鼻の座標 (1 点) を採用した。

提案法で用いる学習データは,「鼻」・「手首」を合わせて 3 点,「手指」が 42 点と手指の情報が占める割合が大きい。手指の情報を減らすことができれば，更なるデータ量削減が行える。そこで，手指の大まかな形が分かれば手話認識が可能ではないかと考え，各指の第 1 関節と第 2 関節の座標を用いず，指の付け根と先端の座標のみを使用することも行う。これによって，手指に関する座標点数が 42 点から 22 点となり，使用する座標を 25 点まで削減できる。

以上の様に，用いる座標は 45 点と 25 点の 2 パターンとする。

3.2.2 座標の正規化

本章冒頭で述べた従来法の問題点である「撮影条件の影響」を受けないよう，提案法では座標の正規化を行う。手話認識に影響を及ぼし得る撮影条件として，以下の 3 点が考えられる。

1. 撮影場所，人物 (背景，服装など)
2. 画面上の被写体の位置
3. 撮影機器と被写体の距離

1. については，座標情報を用いていることで背景，服装などの情報を完全に排除できる。
2. は，被写体が画面の右端に写っている場合と，左端に写っている場合など，被写体の位置ずれによる影響を意味している。この影響を排除する為，提案法では「鼻」·「両手首」の 3 点に関して，以下の様に「鼻」の座標を原点とした絶対座標に変換している。

$$\begin{aligned}(x'_{nose}, y'_{nose}) &= (x_{nose}, y_{nose}) - (x_{nose}, y_{nose}) \\ (x'_{rwrist}, y'_{rwrist}) &= (x_{rwrist}, y_{rwrist}) - (x_{nose}, y_{nose}) \\ (x'_{lwrist}, y'_{lwrist}) &= (x_{lwrist}, y_{lwrist}) - (x_{nose}, y_{nose}) \end{aligned} \tag{3.1}$$

ここで (x_{nose}, y_{nose}) は鼻の座標，(x_{rwrist}, y_{rwrist}) は右手首の座標，(x_{lwrist}, y_{lwrist}) は左手首の座標を表している。この変換により，被写体の位置ずれが発生した場合でも，同様に,「手指」の座標は手首の座標を原点とした絶対座標へ，以下の様に変換している。

$$\begin{aligned}(x'_{rfin,k}, y'_{rlfin,k}) &= (x_{rfin,k}, y_{rfin,k}) - (x_{rwrist}, y_{rwrist}) \\ (x'_{lfin,k}, y'_{lfin,k}) &= (x_{lfin,k}, y_{lfin,k}) - (x_{lwrist}, y_{lwrist}) \end{aligned} \tag{3.2}$$

ここで，$(x_{rfin,k}, y_{rfin,k})$ は手指の座標を表しており，$k = 1, 2, ..., K$ が各指，各関節を表している。この処理によって，被写体の位置ずれに影響されない手話認識を行うことが出来る。
3. は，撮影機器と被写体の距離が変わることで生じる，画面上における被写体の大きさの変化による影響を意味している。この影響を排除する為，提案法では原点から各座標までの距離を，鼻から首までの長さで以下の様に正規化している。

$$(x'_{all}, y'_{all}) = \frac{(x_{all}, y_{all})}{l} \tag{3.3}$$

ここで，(x_{all}, y_{all}) は全座標を表しており，l は鼻から首までの長さを表している。この処理によって，撮影機器と被写体の距離が変化しても，座標間の距離を一定に保つことができ，撮影距離に影響されない手話認識を行うことができる。

第 4 章　実験

4.1　実験条件

20 種類の単語について手話単語認識を行う。訓練は GeForce GTX750Ti(2GB memory) を積んだ PC 上で行い，オプティマイザに Adam[12](学習率：0.001)，活性化関数に ReLU を用いた。全ての CNN 層にバッチ正規化，全結合層にはレイヤ正規化を取り入れている。また，CNN を除く全ての層で dropout(rate=0.5) を使用している。全てのネットワークでバッチサイズ 40，訓練回数は 1000epoch とした。

4.1.1　データセット

本論文で用いるデータセットは，自作したオリジナルのデータセットである。8 名の手話者に 20 種類の手話を行ってもらい，その様子を撮影した。撮影機器は SONY HDR-AS50 デジタル HD ビデオレコーダー 5 台で，手話者の正面，左右 15°，30 °からそれぞれ撮影した。撮影はフレームレート 30fps，RGB のカラーで行った。例として，撮影した動画の 1 フレームを図 4.1 に示す。

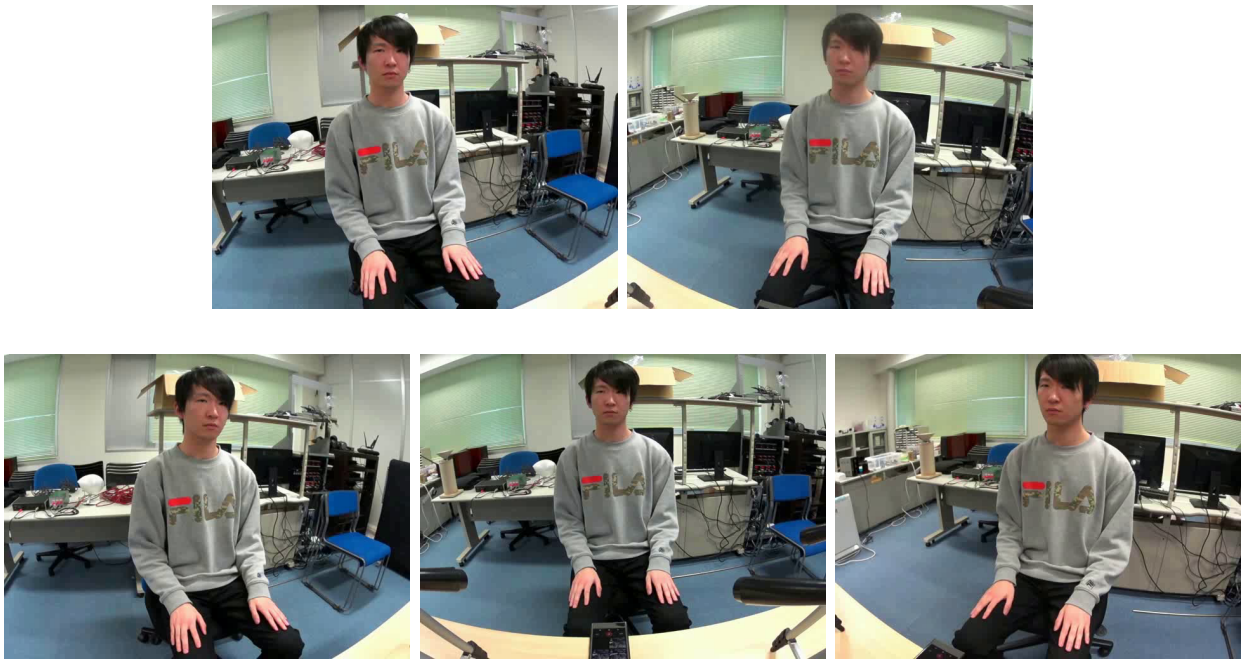


図 4.1　撮影した動画

表 4.1　使用した単語

a	1	おはよう，こんにちは， こんばんは
	2	春， 秋
b	1	クイズ， 黄色， 黒， 頭， 髪
	2	新しい， 影響を受ける， 映画
	3	悪い， 古い
	4	嘘， おいしい
c	1	違う， わからない， わかる

使用した単語は，手の動きに関して以下の 3 つのグループに分けられる。

a 動作の中で似た動きを含むもの

b 手が近い位置で異なる動きをするもの

c 手の位置，動きともに他の単語と異なるもの

また，使用した 20 種類の単語を表 4.1 に示す。

ここで，a-1 と b-2 に含まれる単語は，手の位置も動きも似ている単語である。

4.1.2　ニューラルネットワークの構成

本論文で使用したニューラルネットワークの構成を図 4.2 に示す。

基本の構成は，時系列データのための LSTM が 3 層と，全結合層が 2 層とシンプルである。動画の場合のみ，CNN を LSTM の前に挿入している。ここで，LSTM の次元数は入力データに依存した大きさとなっている。提案法の入力データは座標であるため，入力データが 45 点の場合は入力次元は 90 になる。また，全結合層への入力は LSTM 各層の出力を結合したものであるため，LSTM の次元数が 90 のとき，全結合層への入力は 270 となる。CNN のフィルタサイズ，チャネル数については VGG[11] を参考にした。VGG はシンプルな構成ながら，画像認識のクラス分類において高い精度を達成したことから，様々な研究で用いられるモデルである。VGG の構成の特徴は，フィルタサイズが小さく，基本的に 3×3 のフィルタを使用すること，同一チャネル数の層を数層重ねた後，max pooling を行い，その際にチャネル数を倍にすることが挙げられる。本論文における動画用のニューラルネットワークは，VGG と同様の特徴を保持しつつ，出力の特徴マップサイズが 1×1 となる様に構成されている。

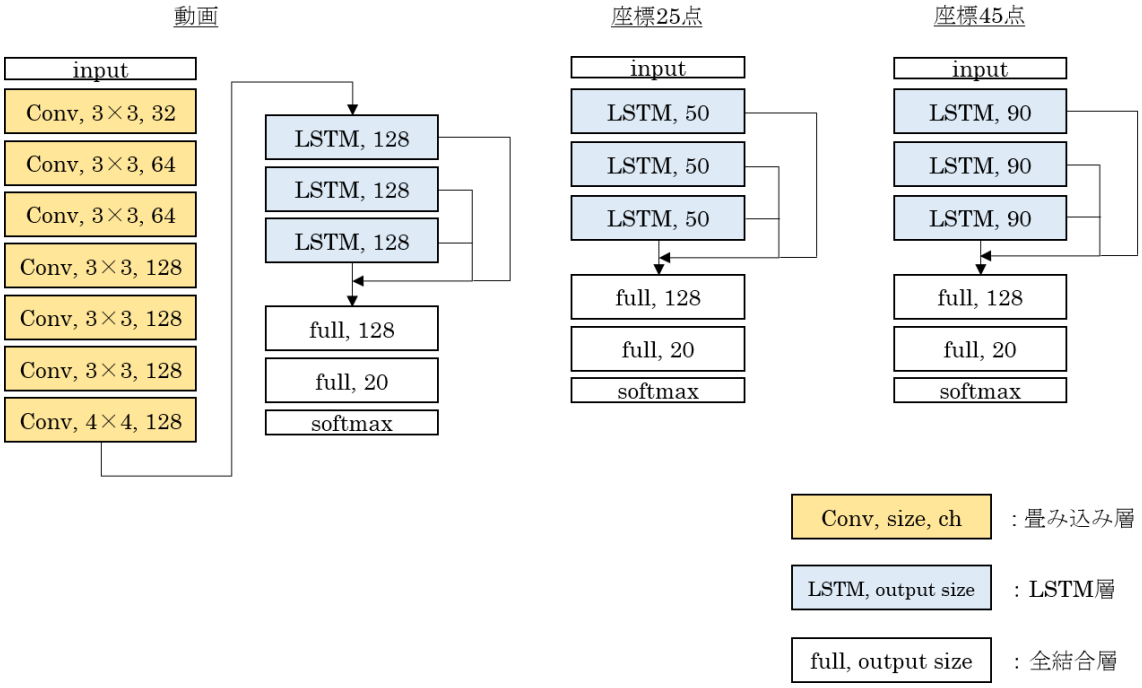


図 4.2　ニューラルネットワークの構成

4.2　提案法による手話単語認識

動画を使った場合と，提案法での手話単語認識の比較を行う。本実験では，データセット中の，正面から撮影したもののみを用いて訓練を行う。従って，学習データ数は 160，それらを 40 個ずつ 4 つのグループに分け，4 分割交差検証を行い，認識精度を検証する。動画を使った場合と提案法の比較のため，学習データサイズ，1 エポックあたりの学習時間，ネットワークのパラメータを調査する。また，単語ごとの認識のしやすさを評価するため，認識結果を混同行列を用いて表す。

4.2.1 結果

動画を使った場合と，提案法での実験結果を表 4.2 に，それぞれの混同行列を図 4.3，4.4，4.5 に示す。表 4.2 から，動画の場合が精度 25% なのに対し，提案法では座標 25 点，45 点の双方で 80% を超える精度を達成していることが分かる。提案法の座標 25，45 点間では，それぞれの認識精度が，83.4%，84.9% と，僅かではあるが 45 点の場合の方が良い結果となった。また，データサイズ，学習時間ともに動画の場合と比べて，大幅に削減できていることが確認できた。パラメータ数については，データサイズ，学習時間ほどではないが，大きく削減できている。

図 4.3 から，動画を用いた場合では，全単語に渡って認識精度が低いことが分かる。また，認識結果が行列内の左上，右下に偏っていることも確認できる。これは，縦軸の上部 10 単語と下部 10 単語で，撮影日時が異なることが原因だと考えられる。撮影日時が異なるため，照明や背景のずれといった変化が認識結果に影響を与えたものと考えられる。これに対して，図 4.4，4.5 から，提案法では撮影日時が異なる単語間で認識結果が偏ることは無い。従って，提案法は照明や背景のずれといった変化に影響を受けにくいことが確認できる。

表 4.2　実験結果

	データサイズ [MB]	学習時間 [/1epoch]	パラメータ数 [個]	認識精度 [%]
動画	1410	1m30s	1082K	25
座標 25 点	3.74	1s	82K	83.4
座標 45 点	6.8	1.5s	233K	84.9

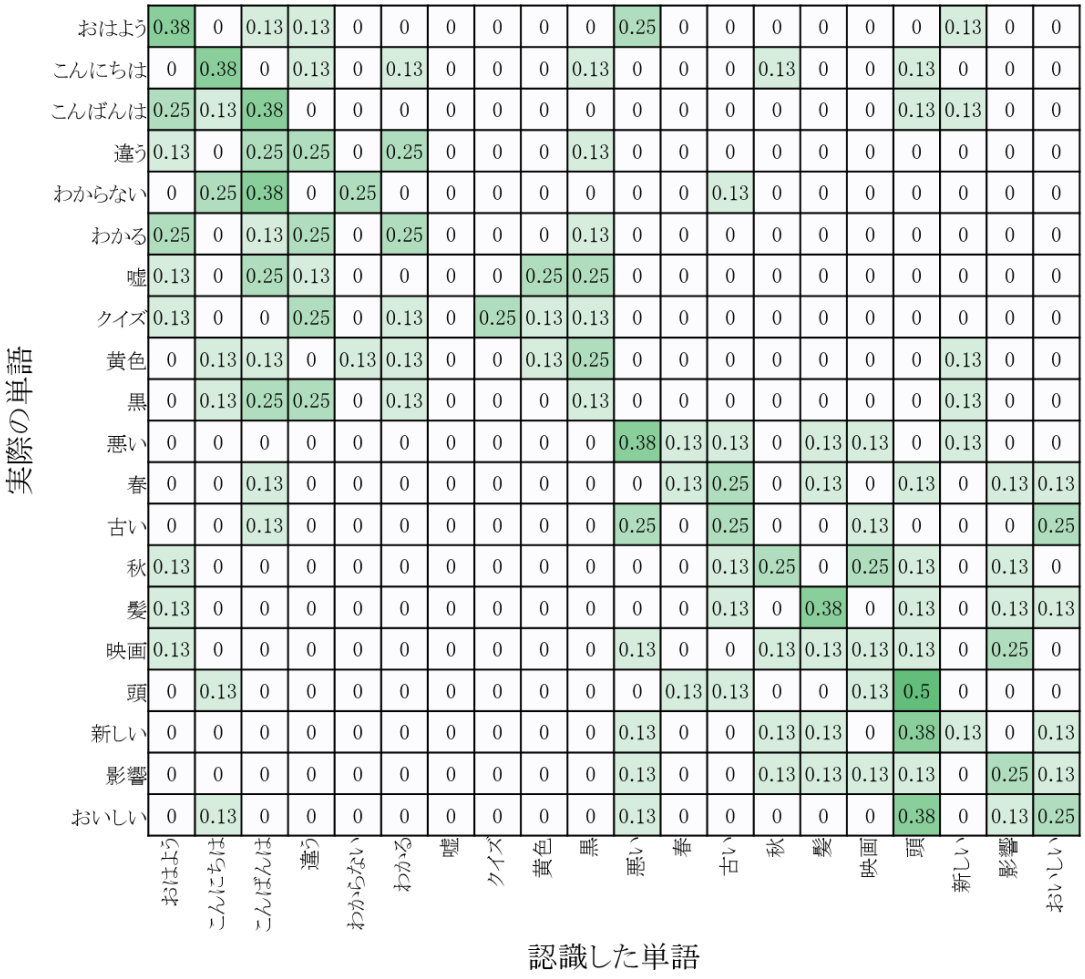

実際の単語
認識した単語
おはよう 0.38 0 0.13 0.13 0 0 0 0 0 0 0.25 0 0 0 0 0 0 0.13 0 0
こんにちは 0 0.38 0 0.13 0 0.13 0 0 0 0.13 0 0 0 0.13 0 0 0.13 0 0 0
こんばんは 0.25 0.13 0.38 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0.13 0.13 0 0
違う 0.13 0 0.25 0.25 0 0.25 0 0 0 0.13 0 0 0 0 0 0 0 0 0 0 0
わからない 0 0.25 0.38 0 0.25 0 0 0 0 0 0 0 0.13 0 0 0 0 0 0 0
わかる 0.25 0 0.13 0.25 0 0.25 0 0 0 0.13 0 0 0 0 0 0 0 0 0 0
嘘 0.13 0 0.25 0.13 0 0 0 0 0.25 0.25 0 0 0 0 0 0 0 0 0 0
クイズ 0.13 0 0 0.25 0 0.13 0 0.25 0.13 0.13 0 0 0 0 0 0 0 0 0 0
黄色 0 0.13 0.13 0 0.13 0.13 0 0 0.13 0.25 0 0 0 0 0 0 0 0.13 0 0
黒 0 0.13 0.25 0.25 0 0.13 0 0 0 0.13 0 0 0 0 0 0 0 0.13 0 0
悪い 0 0 0 0 0 0 0 0 0 0 0.38 0.13 0.13 0 0.13 0.13 0 0.13 0 0
春 0 0 0.13 0 0 0 0 0 0 0 0 0.13 0.25 0 0.13 0 0.13 0 0.13 0.13
古い 0 0 0.13 0 0 0 0 0 0 0 0.25 0 0.25 0 0 0.13 0 0 0 0.25
秋 0.13 0 0 0 0 0 0 0 0 0 0 0 0.13 0.25 0 0.25 0.13 0 0.13 0
髪 0.13 0 0 0 0 0 0 0 0 0 0 0 0.13 0 0.38 0 0.13 0 0.13 0.13
映画 0.13 0 0 0 0 0 0 0 0 0 0.13 0 0 0.13 0.13 0.13 0.13 0 0.25 0
頭 0 0.13 0 0 0 0 0 0 0 0 0 0.13 0.13 0 0 0.13 0.5 0 0 0
新しい 0 0 0 0 0 0 0 0 0 0 0.13 0 0 0.13 0.13 0 0.38 0.13 0 0.13
影響 0 0 0 0 0 0 0 0 0 0 0.13 0 0 0.13 0.13 0.13 0.13 0 0.25 0.13
おいしい 0 0.13 0 0 0 0 0 0 0 0 0.13 0 0 0 0 0 0.38 0 0.13 0.25
おはよう
こんにちは
こんばんは
違う
わからない
わかる
嘘
クイズ
黄色
黒
悪い
春
古い
秋
髪
映画
頭
新しい
影響
おいしい

図 4.3　混同行列 (動画)

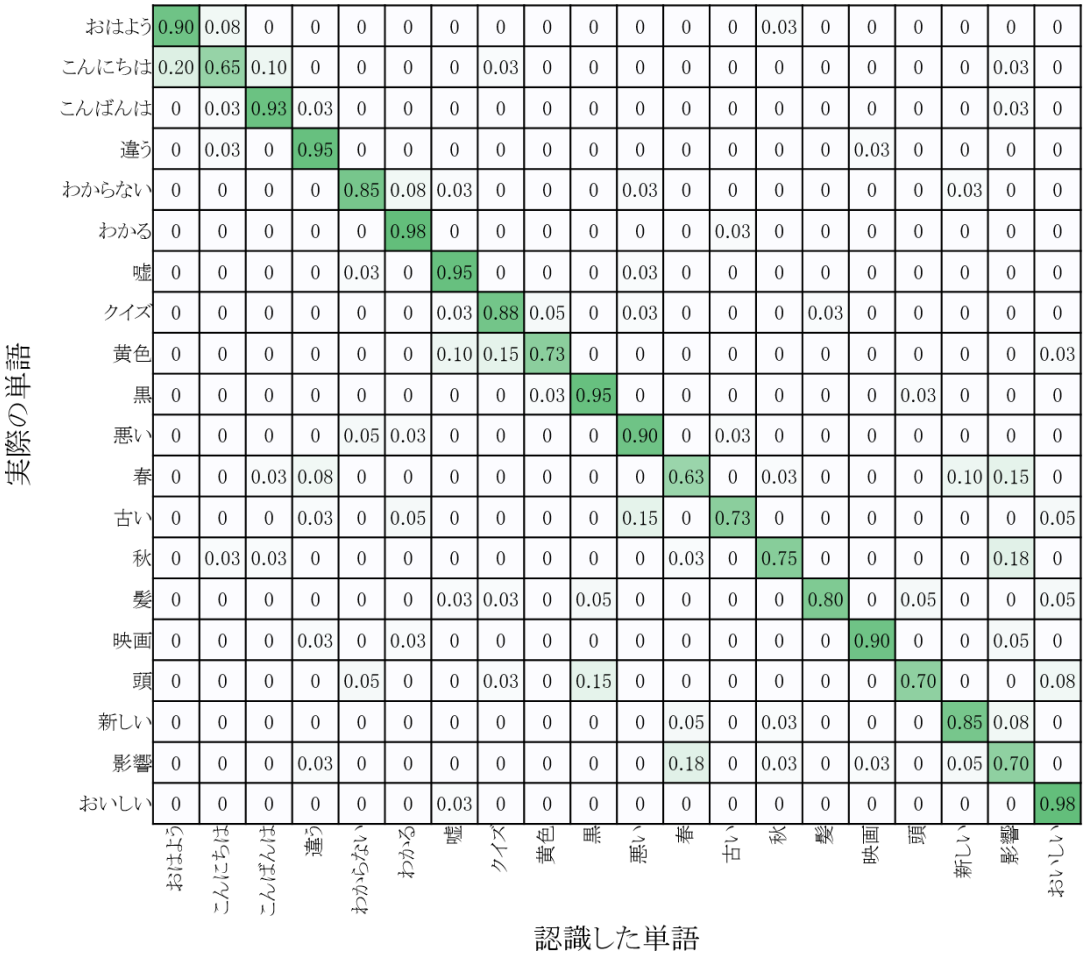


図 4.4　混同行列 (座標 25 点)

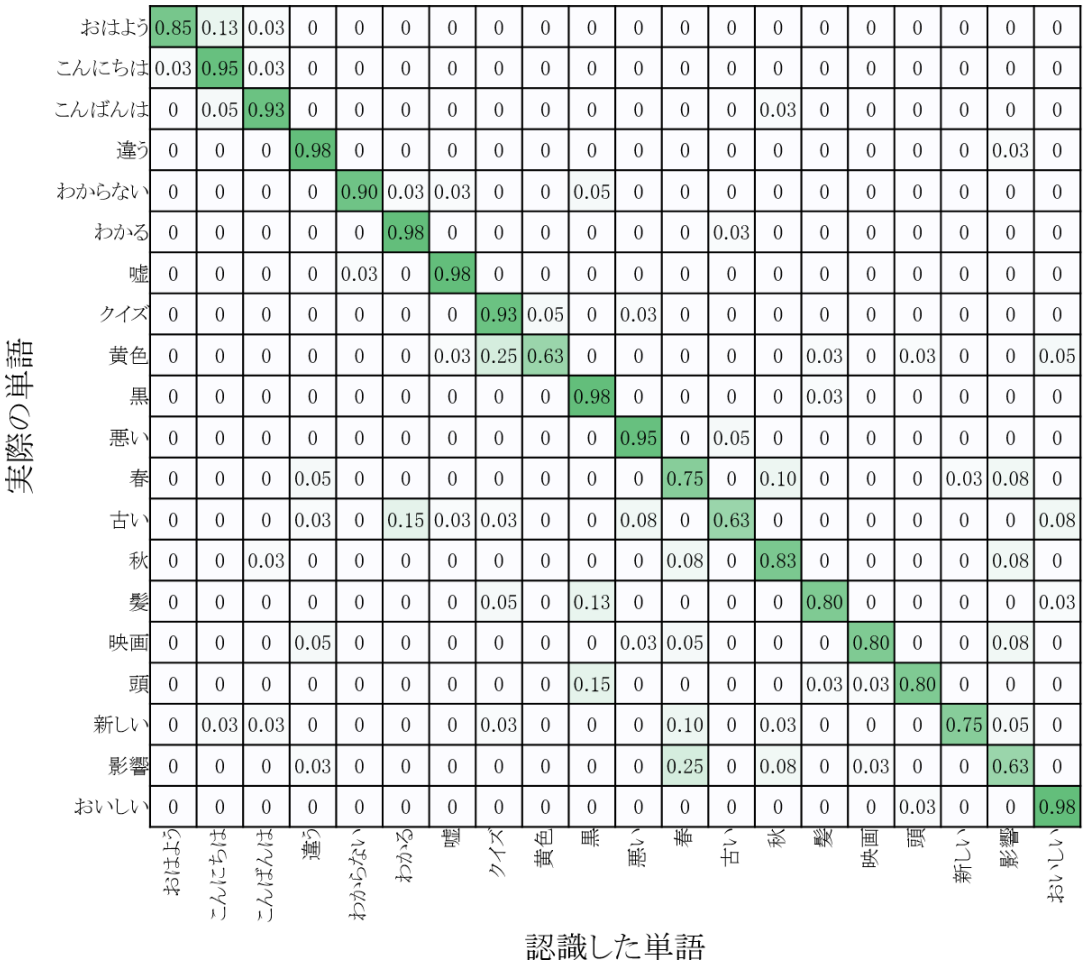


図 4.5　混同行列 (座標 45 点)

4.3　正規化による効果の検証

3.2.2 項で述べたように，提案法では,「被写体の位置ずれ」,「カメラと被写体の距離の変化」に影響されない認識のために，座標を正規化している。本項では，この正規化の効果を検証する。

検証のため，訓練データには正規化を行っていない座標データを用いる。また，テスト

データには，擬似的に被写体の位置ずれと，カメラとの距離の変化を擬似的に再現した座標データを用いる。それぞれの再現は以下の様に行っている。

- 位置ずれデータの再現：次式を用いて被写体を右に移動させる

$$(x'_{all}, y'_{all}) \quad = \quad (x'_{all}, y'_{all}) + (100, 0) \tag{4.1}$$

- 距離の変化データの再現：次式を用いてカメラと被写体の距離を近づける

$$(x'_{rwrist}, y'_{rwrist}) \quad = \quad (x_{rwrist}, y_{rwrist}) + \frac{x_{rwrist}, y_{rwrist} - x_{nose}, y_{nose}}{2} \tag{4.2}$$

$$(x'_{lwrist}, y'_{lwrist}) \quad = \quad (x_{lwrist}, y_{lwrist}) + \frac{x_{lwrist}, y_{lwrist} - x_{nose}, y_{nose}}{2} \tag{4.3}$$

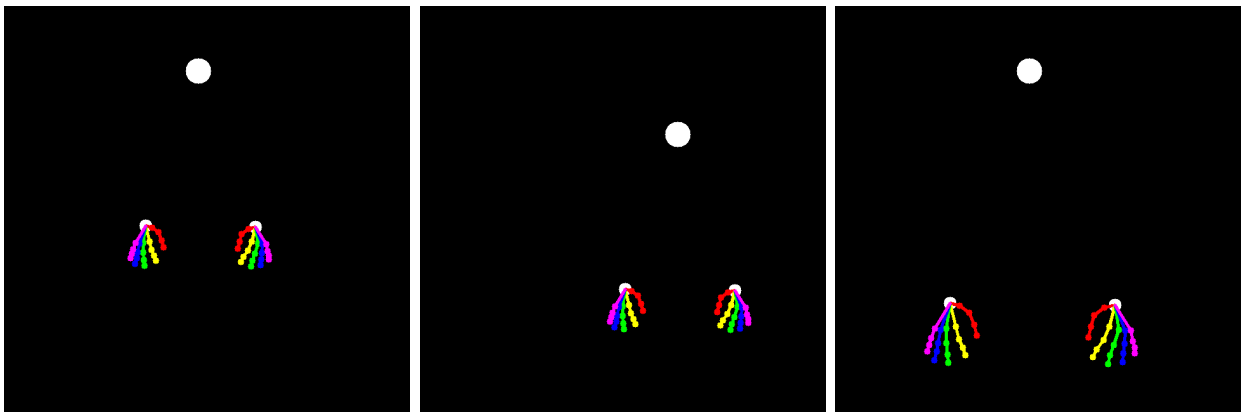
$$(x'_{rfin,k}, y'_{rfin,k}) \quad = \quad (x_{rfin,k}, y_{rfin,k}) + \frac{x_{rfin,k}, y_{rfin,k} - x_{rwrist}, y_{rwrist}}{2} \tag{4.4}$$

$$(x'_{lfin,k}, y'_{lfin,k}) \quad = \quad (x_{lfin,k}, y_{lfin,k}) + \frac{x_{lfin,k}, y_{lfin,k} - x_{lwrist}, y_{lwrist}}{2} \tag{4.5}$$

上記の方法で擬似的に再現したデータの一例を図 4.6 に示す。また，学習データは 4.2 節と同様，正面から撮影したもののみを用いて行う。

4.3.1 結果

被写体の位置ずれ，カメラとの距離の変化それぞれの認識精度を表 4.3 に示す。表 4.2 の正規化ありの結果と比較すると，被写体の位置ずれでは 2〜3 割，カメラとの距離変化では



(a) 元データ　(b) 被写体の位置ずれ　(c) 被写体との距離変化

図 4.6　再現データの描画

表 4.3　認識精度

	位置ずれ	距離変化
座標 25 点	52.5	68.1
座標 45 点	64.4	75.6

1〜2 割 程度の認識精度低下が見られる。従って，提案法の正規化なしでは，位置ずれ，距離変化の双方で，認識精度に悪い影響を受けてしまうことが確認できた。

4.4　撮影角度変化による影響の検証

提案手法では,「被写体の位置ずれ」,「カメラと被写体の距離の変化」という撮影条件の変化に影響されない認識を可能としている。しかし，他の撮影条件の変化として，撮影角度が変化することが考えられる。4.2，4.3 節では，被写体を正面から撮影したものに限って認識を行った。本節では，撮影角度の変化による認識精度への影響を検証する。

訓練データには，他の実験同様に正面から撮影したもののみを用い，テストデータには，正面以外から撮影したもの (左右 15°，30°) を用いる。また，全角度のデータを用いた訓練も行い，撮影角度の変化に頑健な認識が可能であるかも検証する。

4.4.1　結果

撮影角度毎の認識精度を表 4.4 に示す。表 4.4 から，左右ともに角度が大きくなるにつれて認識精度が下がり，30°ではおよそ　2 割低下することが分かる。また，左右で認識精度の低下に差がみられ，左に角度がついた場合の方が同じ角度でも精度が低下している。これは，今回の実験で用いた手話単語が両手を用いるもの，あるいは右手のみを用いるものであり，左に角度がつくと右手がカメラから遠ざかるためだと考えられる。

全角度のデータを訓練に用いた場合では，正面以外の角度についても，正面と同等の認

表 4.4　撮影角度毎の認識精度

		テストデータ				
		左 30°	左 15°	正面	右 15°	右 30°
訓練データ	座標 25 点 (正面)	56.9	71.1	83.4	82.7	66.3
	座標 45 点 (正面)	61.4	75.9	84.9	83.8	67.6
	座標 25 点 (全角度)	80.7	88.1	90	89.4	83.4
	座標 45 点 (全角度)	83.8	89.4	90.7	91.3	85.7

識精度を達成しており，撮影角度の変化に頑健な認識が可能であることが確認できた。座標 25，45 点間では，いずれの撮影角度においても 45 点の場合の方が良い結果となった。また，正面の認識精度について，正面データのみでの訓練時より高い精度を達成している。これは，撮影角度が異なるものも含めて，訓練データが増えたことにより，汎化性能が増したものと考えられる。

4.5　まとめ

本章では，提案法の有効性を確認するため，動画を用いる方法と，提案法で 20 種類の単語について手話単語認識の実験を行い，認識精度，学習時間，データサイズ，パラメータ数で比較を行った。結果として，提案法の方が認識精度は高く，学習時間，データサイズ，パラメータ数も大幅に削減できているため，提案法の有効性を確認できた。

また,「被写体の位置ずれ」,「カメラと被写体の距離の変化」という撮影条件の変化に頑健な認識のために，提案法で行っている座標の正規化についても，有効性を確認するための実験を行った。具体的には,「被写体の位置ずれ」,「カメラと被写体の距離の変化」を擬似的に再現したデータに対して，正規化を行わない場合，認識精度がどう変化するかを確認した。その結果，認識精度が 1〜3 割低下したため，提案法における正規化の有効性を確認できた。

さらに，撮影条件の変化として撮影角度の変化についても検証を行った。正面から撮影したデータのみで訓練した場合，撮影角度の変化に伴って，認識精度が 1〜3 割低下した。撮影角度が変化したものも含めて訓練を行うと，正面以外の角度についても認識精度が向上し，撮影角度の変化に対応した認識が行えた。

これらの実験を通して，学習時のデータサイズを削減した，撮影条件の影響を受けにくい手話単語認識を実現できた。

第 5 章　まとめ

本論文では，学習時のデータサイズの削減と，撮影条件の影響を受けにくい手話単語認識の実現を目的に，座標情報による手話単語認識を提案した。第 1 章では，手話認識システムへの期待，従来の，動画を用いた手話認識手法の問題について述べ，本論文の研究目的について述べた。第 2 章では，本研究で手話単語認識に用いた CNN と LSTM について，内部処理とパラメータについて説明した。第 3 章では，提案法である「鼻」·「手首」·「手指」の座標情報による手話単語認識について述べた。座標情報を抽出する OpenPose について紹介し，撮影条件の影響を受けにくくするための座標の正規化について説明した。第 4 章では，8 名の手話者で撮影した 20 種類の手話単語を用いて，認識精度，データサイズなどの評価実験を行うことで，提案法の有効性を確認した。まず，動画を用いた場合と提案法で評価の比較を行った結果，提案法がよりよい認識精度を達成し，データサイズなどが削減できていることを確認できた。次に，撮影条件の変化を擬似的に再現することで，提案法における座標の正規化を行わない場合，認識精度が低下することが確認できた。また，正面から撮影したデータのみで訓練した場合，撮影角度の変化は認識精度にどう影響を与えるかを検証した。結果として，撮影角度が大きくなるにつれて認識精度は低下したが，正面以外から撮影したデータも含めて訓練することで，撮影角度が変化した場合も正面と同等の認識精度を達成した。以上から，学習データサイズを削減し，撮影条件に影響されにくい手話単語認識を実現できたといえる。

第 6 章　今後の課題

本論文では，20 種類の手話単語に対して，単語レベルの手話認識を行った。しかし，実際の手話は単語の種類が膨大で，文章を表現するために動作が連続して展開される。従って，認識可能な単語数の増加と，文章レベルの手話認識が今後の課題である。また，本論文では撮影角度の変化に対して，各角度から撮影したデータを含めて訓練することで対応したが，この方法では対応させる角度の数だけ訓練データが増加してしまう。この訓練データの増加を抑えることも，今後の課題といえる。

謝辞

本研究を行うにあたり，適切なご指導ご鞭撻を頂いた，杉田泰則准教授に深く感謝いたします。また，本論文の審査において的確なご指示を頂きました、本学電気系岩橋政宏教授ならびに坪根正准教授に感謝いたします。そして，多くのご指摘をくださいました信号処理応用研究室の皆様に感謝いたします。

最後に、勉学に励む機会を与え，様々な面において支えていただいた両親に深く感謝いたします。

平成 31 年 2 月

参考文献

[1] Jie Huang, Wengang Zhou, Qilin Zhang, Houqiang Li, Weiping Li, “Video-based sign language recognition without temporal segmentation,” In Proceedings of the AAAI Conference on Artificial Intelligence, pp. 2257 - 2264, February 2018.

[2] Junfu Pu, Wengang Zhou, and Houqiang Li, “Dilated Convolutional Network with Iterative Optimization for Coutinuous Sign Language Recognition,” In International Joint Conference on Artificial Intelligence (IJCAI), pp. 885 - 891, July 2018.

[3] Bowen Shi, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Greg Shakhnarovich, and Karen Livescu, ”American Sign Language fingerspelling recognition in the wild”, In 2018 IEEE Workshop on Spoken Language Technology (SLT 2018), Desember 2018.

[4] Vo Hoai Viet, Nguyen Thanh Thien Phuc, Pham Minh Hoang, Liu Kim Nghia, “Spatial-Temporal Shape and Motion Features for Dynamic Hand Gesture Recognition in Depth Video,” International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.10, No.9, pp. 17-26, September 2018.

[5] Shubham Juneja, Chhaya Chandra, P.D Mahapatra, Siddhi Sathe, Nilesh B. Bahadure and Sankalp Verma, ”Kinect Sensor based Indian Sign Language Detection with Voice Extraction ,” In International Journal of Computer Science and Information Security (IJCSIS), Vol. 16, No. 4, pp. 135 - 141, April 2018

[6] T Handhika, R I M Zen, Murni, D P Lestariand I Sari, ”Gesture recognition for Indonesian S ign Language (BISINDO),” Journal of Physics: Conference Series, vol. 1028, no. 1, p. 012173, June 2018.

[7] D. Avola, M. Bernardi, L. Cinque, G. L. Foresti, and C. Massaroni, “Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures,” IEEE Transactions on Multimedia, Vol.21, No.1, pp. 234 - 245, January 2019.

[8] Biyi Fang, Jillian Co, Mi Zhang, ”DeepASL : Enabling Ubiquitous and Non-Intrusive Word and Sentence-Level Sign Language Translation,” In Proceedings of SenSys ’17, Delft, Netherlands, November, 2017.

[9] Teak-Wei Chong and Boon-Giin Lee, ”American Sign Language Recognition Using Leap Motion Controller with Machine Learning Approach,” Sensors, 3554 October 2018.

[10] Z. Cao, T. Simon, S. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1302 - 1310, July 2017.

[11] K. Simonyan, A. Zisserman, ”Very deep convolutional networks for large-scale image recognition”, In International Conference on Learning Representations (ICLR), May 2015.

[12] D. Kingma and J. Ba, ”Adam: Amethod of stochastic optimization”, Published as conference paper at ICLR, pp. 1-15, 2015.