距離画像を用いた
動きのある指文字認識に関する研究

平成 24 年度

筑波技術大学大学院 修士課程 技術科学研究科

産業技術学専攻

三宅 太一

目 次

図 目 次

表 目 次

筑波技術大学

修士（工学）学位論文

第1章　序論

1.1　研究背景

音声認識技術，およびその関連技術についての研究が進み，音声による入力機能を備えた情報機器が広く普及してきた．音声を入力として用いることによって，ユーザが手で直接インタフェースを操作する必要なく，非接触で操作を行うことが可能になるため，様々な場面での活用が期待されている．しかし，聴覚障がい者のなかには明瞭な発声が困難な者も多く，音声入力を利用できない場合がある．そこで，音声に代わる入力方法として，指文字を用いた入力方法について検討した．

主に音声入力が活用される場面としては，視線をそらしたり手を離したりできない状況や，直接触れるのが困難な状況で情報機器を操作するような場合である．前者の一例としては，車の運転中にカーナビゲーションシステム（カーナビ）を操作するときである．走行中に視線をカーナビの画面に向けて直接操作するのは危険がともなうため，音声入力が活用されることが多い．後者の例としては，料理などの作業中で手が濡れている，あるいは汚れているときである．端末に直接触れる事ができない状況においても，音声入力を用いることでレシピや手順を確認するアプリケーションを操作することができる．また，最近ではテレビやエアコンなどの家電機器に対するリモコンによる操作の代替として，音声入力が利用されつつある．その他にも，タッチパネルのような操作インタフェースに慣れていない一部の高齢者ユーザや，身体障がい者の入力手段としても注目されている．

しかし，音声入力はすべてのユーザにとって必ずしも有効な入力インタフェースにならない．特に，聴覚障がい者のユーザの場合は発話障がいを持ち，明瞭な発声が困難な者も多い．言語機能形成期に聴覚を失ったり，聴力が低下したりしていた者は，高性能な補聴器を使用して発話訓練を十分に積んでいても機器が十分に認識可能なレベルの音声を発声することは困難である．このように，聴覚障がい者の発声は健聴者と比較して不明瞭かつ個人差が非常に大きいため，先に述べた機器の操作等に音声入力をうまく活用できない場合がある．今後，さらに音声

認識とその関連技術が進歩するにつれて，音声入力による操作を前提としたインタフェースが普及し，聴覚障がい者にとって利用しにくい，あるいは利用できない機器が増えていく可能性が高い．

音声入力の代わりにデバイスを操作する方法として，身振りなどの映像入力による操作が考えられる．聴覚障がい者が互いにコミュニケーションをとる主な手段である手話は，腕や手指の動きで表現されるため，身振りと同様に映像入力として利用できる可能性がある．しかし，手話は音声言語や書記言語に比べて語彙数が少なく，手話単語に無い新しい単語や固有名詞の表現には向かない．そのため，これらの表現には指文字が利用されている．指文字とは手の型や動作を書記言語に対応させたものである．日本では図 1.1 のように平仮名 1 文字ずつに対応しており，各文字の手型を連続的に出すことで様々な単語や文を表現できる．また，上半身と両手を利用する手話表現と異なり，指文字は片手で小さな動きで表現することが可能である．

このことから，入力手段として汎用性が高く，利用できる場面も多いと考えられる．例えば，現状の音声入力の代替手段として，図 1.2 のように視線をそらさずにカーナビを操作したり，図 1.3 のように情報機器や家電機器を操作することが可能になると考えられる．その他，聴覚障がい者が指文字や手話の分からない健聴者と会話をする場合，指文字の表現速度が十分に早ければ，端末に指文字で入力して相手に見せることで手書きよりもスムーズにコミュニケーションがとれる場合も考えられる．

図 1.1: 日本の指文字一覧

図 1.2: 指文字でカーナビを操作する例

図 1.3: 指文字で情報機器を操作する例

これらの指文字を入力手段として利用するためには，対象の情報機器にユーザが連続的に出した指文字を読み取らせ，文や単語を認識させなければならない．先行研究では，カメラ画像や様々なセンサを用いて，手型が静止した指文字のみを対象にして 1 文字 1 文字の認識を試みた例が数多く報告されている．しかし，日本の指文字は濁音や半濁音などの手型をある決まった方向に動かして表現するものも含まれており，これらの動きのある指文字も対象とした認識方法がなければ指文字による入力手段は実現できない．そこで，本研究では濁音，半濁音，小書き文字などの手型に動きのある指文字を 1 文字ずつ認識するための基礎的な研究を行った．

1.2　指文字とは

日本の指文字の表現は合計で 81 文字存在する．そのうち清音の指文字については，手の形を静止させて表現する指文字が 41 文字，手指を動かす指文字が「の」「も」「り」「ん」の 4 文字，また，図 1.4 のように，濁音，半濁音，小書き文字，促音，長音，および「を」のように，手型を固定しながら平行移動させて表現する指文字が 36 文字ある．動きのある指文字は，例えば濁音の「ぎ」であれば手型を「き」の形にしたまま横方向に動かして表現する．半濁音の「ぷ」は指文字「ふ」の形のまま上方向に動かし，促音および小書き文字の「っ」「ょ」であれば「つ」と「よ」を手前の方向に動かす．清音指文字の「を」の場合は「お」の指文字を手前に動かして表現する．長音は「そ」の形のまま下方向に動かして指文字を表現する．

本論文では，手型を静止して表現する指文字のことを静止指文字,「の」「も」「り」「ん」のような指文字を手指を動かす指文字，濁音，半濁音，小書き文字，促音，長音，および「を」を表現する指文字を手型の動きのある指文字と呼ぶ．

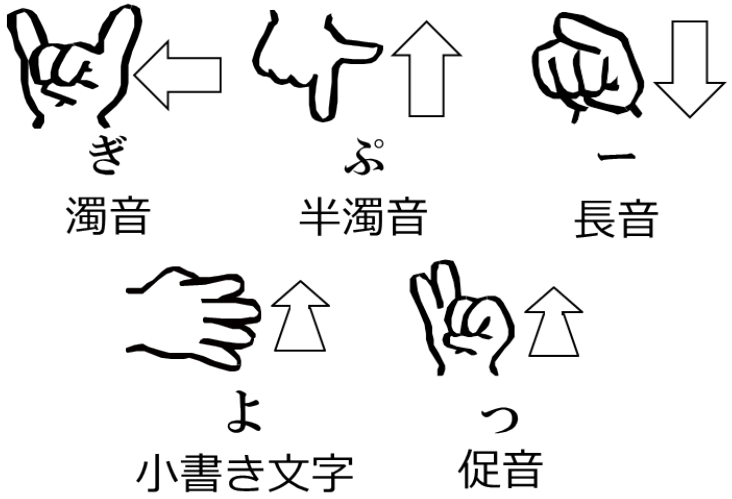

図 1.4: 手型の動きのある指文字の例

1.3　研究目的

指文字入力インタフェースを実現するためには，ユーザが表出した指文字を 1 文字ずつ読み取り認識し，単語や文脈を推定して，デバイス上で各種操作を実行させる必要がある．先行研究の多くは指文字の認識を 1 文字ずつ行う方法の提案を試みており，全ての指文字のうち，静止指文字に限定して精度よく認識することを目的にしている．しかし，指文字の表現方法の中には濁音などの手型の動きのあるものも含まれており，それらの指文字の認識を目的とした研究は例を見ない．

本研究では，日本の指文字のうち手型の動きのある指文字，つまり濁音，半濁音，小書き文字に対応した指文字を認識するための方法を提案することを目的とする．手型の動きのある指文字を効率よく認識するために，手領域をリアルタイムかつ正確に抽出し，動作方向と動作量を検出して認識を行う．

本論文ではまず，距離画像を用いて対象の体積を計算して手領域を精度良く抽出する方法と，指文字の手領域の位置の履歴から動作を検出して濁音，半濁音，小書き文字を識別する方法について述べ，認識実験を行った結果について考察する．次に，考察した結果を受け，手型の動きに起因する距離画像の特徴的なノイズが含まれる画像を，学習データとして識別器で認識させる方法を提案する．さらに，検出した手領域の動作と組み合わせて動きのある指文字の認識精度を向上させる方法について述べる．

1.4　本論文の構成

本章では，本研究の研究背景と目的について述べた．まず第 2 章で関連研究とその問題点について述べ，本研究の位置付けを明らかにする．次に，第 3 章で手領域の抽出，および手型識別と動作検出について述べ，認識実験を行った結果について考察する．第 4 章では第 3 章で得た知見から手型識別を改良し，動作検出を最適化する方法について述べ，認識率を改善させた結果について述べる．最後に，第 5 章でまとめ，今後の課題について述べる．

第2章　関連研究と本研究の位置付け

2.1　指文字認識の関連研究

指文字認識の関連研究では，静止指文字を対象とした報告が多く，動きを含む指文字を含めたすべての指文字の認識を試みた報告は例をみない．手型が静止した清音の指文字を認識する関連研究の方法は，2種類の方法に大別できる．1つはデータグローブなどの接触式センサを用いて得られるデータをもとに認識する方法，もう1つは指文字を撮影した各画素に輝度値を格納したカラー画像や距離値を格納した距離画像に対して画像処理を行い非接触で認識を行う方法である．

2.1.1　接触式センサ用いた方法

データグローブのような接触式センサを用いると，各関節の回転角から手指の屈伸情報を特徴量として得ることができる [1]．磁気センサを装着したグローブを用いた方法 [2] では，アルファベットの指文字を高い精度で認識できることが報告されている．このような接触式のデバイスを装着することで指文字を認識するような方法の多くは，システムを利用する度にユーザに対して着脱の負担を強いることになる．また，データ入力用のケーブルが計算機と繋がっているため，手型の動作範囲が制限されてしまい，自然な指文字表現を妨げる可能性がある．その他にも，入力インタフェースとして用いる場合，手が汚れる，料理をするなどの状況下では利用が困難になる．これらの問題点を改善するため，デバイスに触れずに指文字を認識させることが可能な非接触式センサを用いた方法が提案されている．

2.1.2　非接触式センサを用いた方法

非接触式センサを用いることで，接触式センサによるユーザの負担を大幅に軽減することができる．このセンサを用いた方法として，撮影したカラー画像から指文字を認識を試みた例が多い．一般的には，指文字を撮影したカラー画像を 2 値化して手の領域を抽出し，手型を認識する方法が知られている．抽出した手の領域から，輪郭や指の本数の情報，手のひらとの位置関係，手首の向きといった手指形状特徴をもとにして指文字の認識を行っている [3][4]．その他にも，2 値化したシルエット画像に対して細線化処理を行い，指の本数情報だけでなく長さや太さ，手指の曲率などの細かい形状特徴をもとにして指文字の認識を行う方法も報告されている [5][6]．

CCD カメラで指文字を撮影してニューラルネットワークを用いた識別器を構築することで指文字を認識する方法 [7] では，静止指文字のうち 15 文字を高い精度で認識している．カラーカメラを 2 台用意し，指文字提示者からみて正面やや左と左真横から撮影する方法 [8] では，一部の指文字を除く指文字 75 文字を高い精度で識別している．

カラー画像を用いたこれらの研究では，特徴量の種類と識別器の組み合わせを工夫して高い認識率で静止した指文字を識別する．先に述べた通りユーザの負担を軽減させることができるが，撮影環境が一定でなければ，手型の特徴量を安定して得ることができない．例えば，肌の色と似たような背景や，一様ではない複雑な背景で撮影する場合は，手領域をうまく抽出するのは難しい．また，照明条件が一定ではない環境では，特徴量も変化してしまう．したがって，識別器の学習データを取得するときの照明や背景の条件と，認識時の条件が異なる場合，認識率は著しく低下する．

上記のような環境光や背景による問題を改善するために，カラーグローブをユーザに装着させることで手領域と背景との分離を容易にした方法も報告されている [9][10][11]．さらに，カラーグローブを手指の部位ごとに色分けし，背景との分離のみでなく手型の部位の特徴も容易に得るための工夫もされている [12]．しかし，接触式センサと同様に，グローブの着脱の負担をユーザに強いる問題や，特別なグローブを装着して作業を行うことが困難な状況では，利用ができないといった問題が生じる．そこで，接触式センサやカラーグローブによる方法やカラー画像による指文字認識方法のもつ問題を解決するために距離画像を用いた方法が近年注目され，様々な研究が進められている．

2.1.3　距離画像を用いた方法

距離センサは，画素ごとに距離値を格納した距離画像を撮影することができる．距離画像を用いた指文字認識では，3 次元スキャナや RGB-D センサといった距離センサを用いて指文字を撮影する方法が提案されている．これらのセンサを用いることにより，カラー画像の方法における，指文字撮影時に適切な背景を選択しなければならない問題や，環境光により生じる手型の陰影などの影響をほとんど無視することが可能である．さらに，距離値によって背景と手領域を容易に分離できるため，認識対象となる指文字を抽出しやすいという利点がある．

3 次元スキャナを用いて指文字を撮影する場合，高解像度な距離画像が得られるため，より精度の高い認識が可能になる．手領域の抽出を行って特徴量を取得し，識別器に決定木 [13] や 3 次元テンプレートマッチング [14] を用いた方法が提案されている．その他にも，抽出した手領域に主成分分析をかけて特徴量の次元を圧縮し，認識速度の向上を試みた方法 [15] もある．これらの研究では静止指文字を高い精度で認識させることに成功しているが，手領域の抽出は手動で行っている．入力インタフェースに適用するためには，抽出の自動化が課題となる．また，解像度と認識率が高い距離画像を取得できる反面，撮影のフレームレートが著しく低く，動きを含む指文字への十分な対応が難しいといった問題点がある．より撮影速度の早いセンサを選択することでこの問題を解決できる可能性がある．

カラー画像と距離画像を同時に撮影できる RGB-D センサの 1 つである Microsoft 社の Kinect を用いて認識を試みた方法 [16] では，SDK として OpenNI[17] を用いて手領域の抽出を行い，リアルタイムな指文字の認識に成功している．この研究はアルファベットの指文字を対象に認識を行っているが，日本の指文字のように手型を動かして表現するものは存在しない．日本の指文字を認識させるためには，手型の動きのある指文字にも対応させる必要があるため，本研究ではこれらの指文字を認識するための方法を提案する．

2.2　指文字認識における本研究の位置付け

関連研究では静止指文字の認識精度の向上を目的としたものが多く，手型の動きのある指文字を対象とした報告は見られなかった．距離画像を用いて手領域の課題を回避した研究でも，静止指文字を対象としたものが多かった．本研究では赤外線 TOF カメラを用いて距離画像を撮影することで，手型の動きのある指文字を認識するための手法を提案する．本研究で採用す

表 2.1: 距離センサのデバイスの比較

センサの種類 名称	3 次元スキャナ ミノルタ社，VIVID300	RGB-D センサ Microsoft 社，Kinect	赤外線 TOF カメラ Mesa 社，SR-3000
計測方式	光切断方式	パターン照射方式	TOF 方式
解像度	400×400	640×480	176×144
距離分解能	1m の距離で 0.001m	1m の距離で 0.01m	1m の距離で 0.006m
撮影速度	約 1fps	最大 30fps	最大 50fps
問題点	動きへの対応が困難	小さな物の計測が困難	低解像度

るセンサと，距離画像を用いた指文字認識の関連研究で用いられた距離センサの比較を表 2.1 に示す．

3 次元スキャナは，1m 離れた場所で 0.001m 程度の高い分解能と，高い解像度の距離画像を撮影することが可能である．しかし，前述のとおり撮影速度が 1fps と遅いためリアルタイムな手領域の抽出および指文字認識ができない．撮影速度の速いセンサを採用すれば，動きへの対応が可能となる．

Kinect センサのパターン照射方式は，赤外線レーザーのパターンを対象に照射し，その歪みから距離を計測する．この方式は，パターンの歪みを十分に計測可能な大きさの対象に照射する必要がある．例えば，指先のような細いものに対して照射すると歪みが確認できず，その部分の正確な距離を計測することができない．また，1m 離れた場所における奥行き方向の分解能が約 0.01m であるために手型の距離画像の欠損が多くなり，認識が困難になると考えられる．これらの問題を解決するためには，より精度よく距離計測が行えるセンサを選択する必要がある．

赤外線 TOF カメラは，照射された赤外線光が反射して帰ってくるまでの時間をもとに，画素ごとに距離を計測する．そのため，指のように細い物体であっても距離値を測定可能である．また，距離分解能が 1m の位置で 0.006m 未満となっているため，Kinect センサよりも精度の良い計測が可能である．撮影速度も十分に早く，動きへの対応も可能である．一方，他のセンサよりも解像度が低いため，従来の静止指文字の認識に比べて認識率が低くなる可能性がある．しかし，動きのある指文字を精度良く認識するためには，解像度よりもフレームレートが重要であると考え，TOF カメラを採用した．

第3章　距離画像を用いた動きのある指文字認識[A1][A2][A3]

3.1　動きのある指文字認識の概要

本章では静止指文字41文字と手型の動きのある指文字34文字を対象とし，それらを認識するための手法についての検討を行った結果について述べる．距離画像を用いた指文字認識処理の流れを次に示す．

処理1 指文字の距離画像の撮影（3.2節）
処理2 距離画像中の手領域抽出（3.3節）
処理3 手領域の手型識別（3.4節）
処理4 手領域の動作検出（3.5節）

まず，処理1で各画素に距離値を格納した距離画像を赤外線TOFカメラを用いて撮影する．これにより，環境光による輝度の変化の影響を考慮する必要がほとんどなくなる．次に，処理2で距離画像中の指文字を表現している手領域の抽出を行う．カメラから手領域までの距離値から計算される体積をもとにして抽出を行うため，カラー画像を用いた方法のように背景を考慮する必要がない．処理3では，手領域の手型をパターン認識により識別し，処理4で手領域の3次元空間中の動作方向と移動量の検出を行う．手型の動きのある指文字については，処理3と処理4を組み合わせて認識を行う．

3.2　指文字の距離画像の撮影

距離画像とは各画素に距離値を格納した画像で，撮影には距離センサとして赤外線 TOF カメラを用いた．TOF とは Time of Flight の略である．カメラの周囲に付けられた LED から照射された赤外線が対象で反射し，カメラで観測されるまでの時間を計測することで物体までの距離を測定することが可能である [20].

本研究では，TOF カメラとして Mesa 社の Swiss Ranger SR-3000[19]-[21] を指文字の撮影に用いた．図 3.1 に製品イメージを，表 3.1 と表 3.2 に製品の概要と距離分解能を示す．同カメラの距離画像の解像度は 176 × 144，最大フレームレートは 50fps であり，関連研究で用いられていたセンサよりも低解像度であるが，高速な距離画像の撮影が可能である．指文字を撮影する際に手型をカメラの近くで出した場合，赤外線の反射光を検出するための時間分解能が不足し正確な距離値を測定することができない．したがって，本研究では手型をカメラから 0.5m〜0.7m の位置で手型を提示する．また，表 3.2 のように，距離によってフレームレートが異なるという特徴をもつ．実際に奥行き 8m 以上ある部屋の中で，照明をすべて点灯させて指文字が提示してみたところ，実測値は 25fps 程度であることがわかった．このことから，動きへと対応させるには十分の撮影速度が確保できていると考え，この環境で本 TOF カメラを使用した．

図 3.1: SR-3000 製品画像

表 3.1: SR-3000 の概要

型式	SR-3000
画素数	176 × 144
視角	47.5 × 39.6
フレームレート	最高 50fps
距離分解能	表 3.2 を参照
インターフェース	USB2.0
光学レンズ	f/1.4
レンズマウント	M12 × 0.5
照射強度 (光学)	$< 1W$
波長	850nm
復調周波数	Default:20MHz
カメラサイズ	50 × 67 × 42.3mm
ケース材質	アルミニウム
ネジ穴	2 × M4：1 × 1/4”
質量	162g
供給電力	12V
消費電力	12W
動作温度	-10℃～+50℃

表 3.2: SR-3000 の距離分解能

距離［m］	0	1	2	3
フレームレート [Hz]	29	20	15	12
分解能 [m]	0.003	0.006	0.013	0.022

赤外線 TOF カメラは照射された赤外線が物体に反射して戻ってくるまでの時間を計測するため，撮影対象となる物体表面の反射特性の違いによりノイズが発生する．SR-3000 の SDK はメディアンフィルタを備えており，ある程度のノイズを除去することができる．得られる距離画像は，画素ごとに距離値が格納されている．実際に指文字を撮影した距離画像について，距離が近い画素を明るく，遠い画素を暗くしたグレースケールの輝度画像で表示した例を図 3.2 に示す．またカメラの SDK を用いて各画素の距離値から 3 次元頂点座標値を計算することができる．得られた座標値を視覚的に表現した例が図 3.3 の画像である．

図 3.2: 距離画像の例（距離値を輝度値に変換して表示）

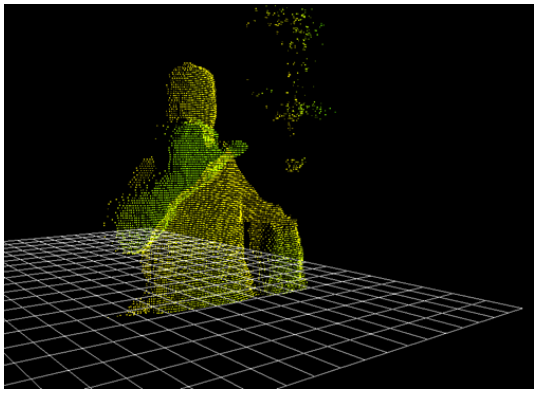

図 3.3: 3 次元頂点へ描画

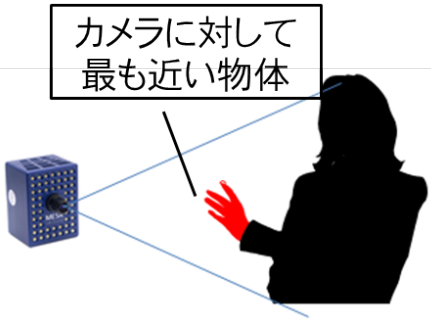

図 3.4: 指文字撮影の様子

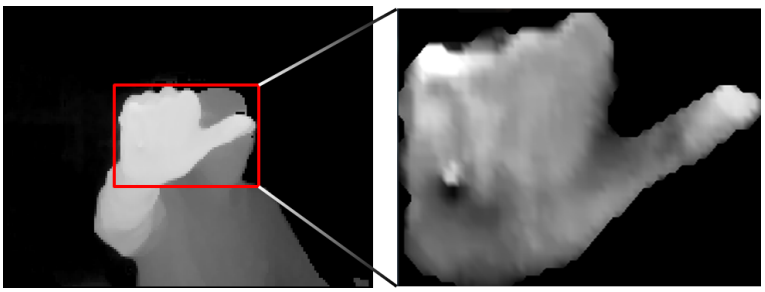

図 3.5: 手領域抽出例

3.3　距離画像中の手領域抽出

カメラに対して指文字を提示したとき，図 3.4 のように指文字を表現している手がカメラから最も近い物体として撮影されると考えられる．実際に抽出される手領域は，図 3.5 のような矩形領域になる．提示された手型をカメラでが撮影したとき，実際に撮影される範囲は手型の表面であり，カメラからは見えない裏側の情報を認識に用いることはできない．つまり，手型を横から見たときにカメラ側の半分よりも前の部分が抽出できれば，この部分の体積は手型の半分程度になると考えられ，体積を用いて手領域をすべて抽出できると考えた．そこで，本研究ではこれらの仮定をもとに，赤外線 TOF カメラを用いて撮影された物体の体積を手がかりにして手領域を動的に抽出する方法を考案した．以降は，図 3.6 で示すようにカメラの横方向を x，上方向を y，光軸方向を z としたカメラ座標系を基準にして説明する．

手領域の体積を計算するためには，距離画像中の各画素が占める面積を求める必要がある．

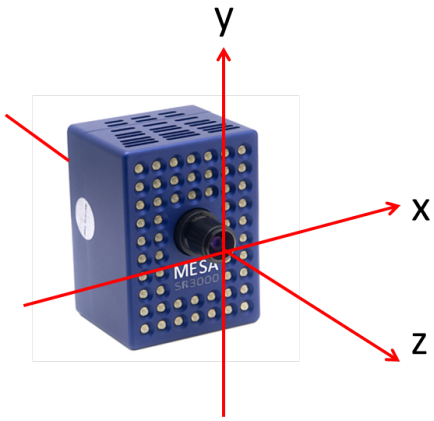

図 3.6: カメラ座標系の設定

カメラで距離画像を撮影したときの，画像のピクセル幅 w とピクセル高 h に対するカメラの画角を図 3.7 のように φ_h，φ_v と表す．図 3.8 で示すように，画像までの距離を z としたときに，距離画像の占める横方向の幅 l_w は，

$$l_w = 2z \tan \frac{\varphi_h}{2} \tag{3.1}$$

同様に，縦方向の幅 l_h についても，

$$l_h = 2z \tan \frac{\varphi_v}{2} \tag{3.2}$$

と表せる．このとき，距離画像中の z の位置にある画素のもつ縦方向の幅は l_v/h, 横方向の幅は l_h/w と表せるため，その画素の占める面積 $s(z)$ は

$$s(z) = \frac{l_w * l_h}{h * w} \tag{3.3}$$

で求まる．この式を展開すると，

$$s(z) = \frac{4z^2 \tan\{\frac{\varphi_h}{2}\} * \tan\{\frac{\varphi_v}{2}\}}{h * w} \tag{3.4}$$

になる．

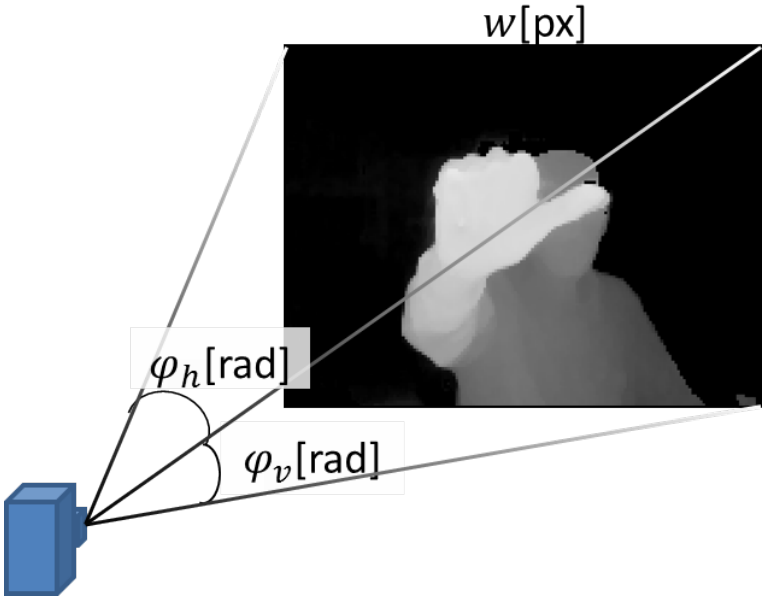

図 3.7: 距離画像の大きさとカメラの画角

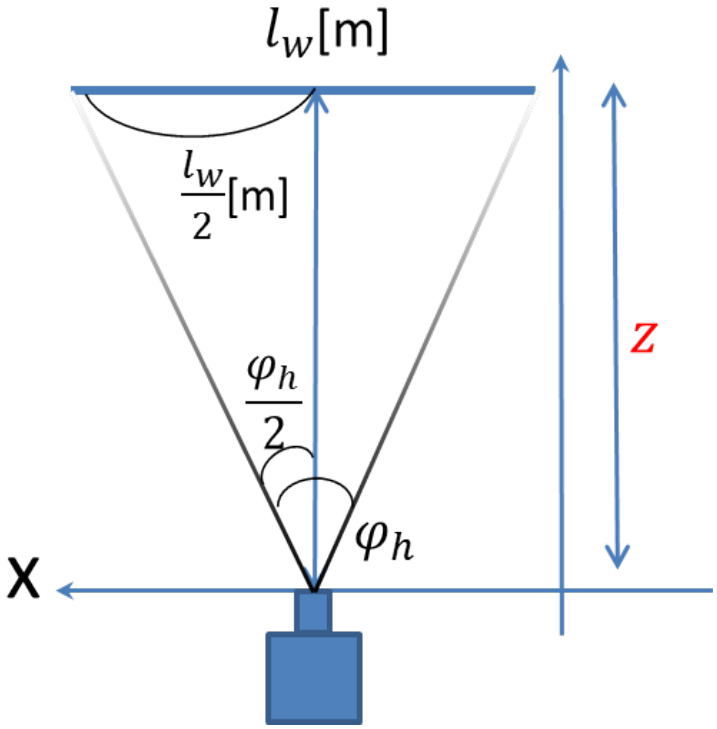

図 3.8: 距離画像の横幅 l_w の導出

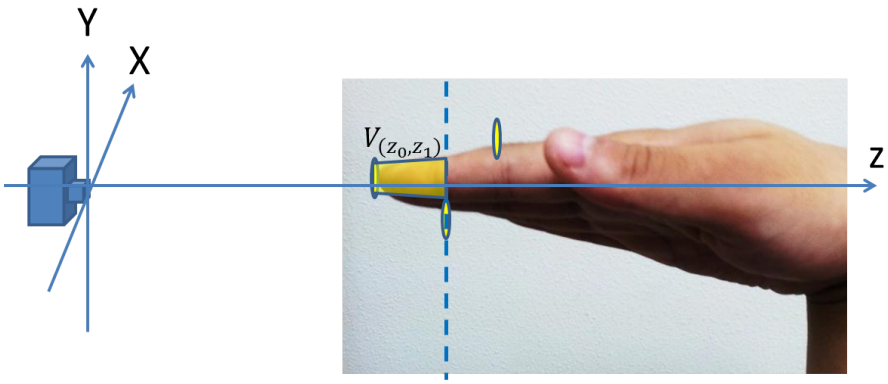

図 3.9: z_0 に対応する画素の四角錐台の体積計算

ある画素と，それよりもカメラから遠い位置にある画素までの z 値をそれぞれ z_n，z_f としたとき，これらの画素の間で形成される四角錐台の体積は

$$V(z_n, z_f) = \frac{\{s(z_f) * z_f - s(z_n) * z_n\}}{3} \tag{3.5}$$

で求めることができる．この体積をもとにして手領域の抽出を行う．

まず，距離画像中の各画素のもつカメラ座標上の z 値を，カメラから近い順にソートしたものを配列に格納する．この配列のインデックス番号を i とし，z 値を z_i と表す．カメラから最も近い位置にある画素までの z 値を z_0 とする．

次に，i を 0 から順に 1 ずつ増やしながら，z_i の位置の画素に対応する四角錐台の体積を計算する．図 3.9 の例では，まず z_0 に対応する画素を天面，z_1 の位置を底面とした四角錐台の体積 $V(z_0, z_1)$ を求めている．

その後，さらに i を増やしながら同様の計算処理を行う．図 3.10 では z_1 に対応する画素画素を天面として，z_2 の位置にある画素を底面とする四角錐台の体積 $V(z_1, z_2)$ を求めている．

このとき，図 3.11 のように，先ほど求めた z_0 の位置の画素を天面とする四角錐台の体積 $V(z_0, z_1)$ を距離 z_2 の位置を底面とした四角錐台の体積 $V(z_0, z_2)$ に更新する．

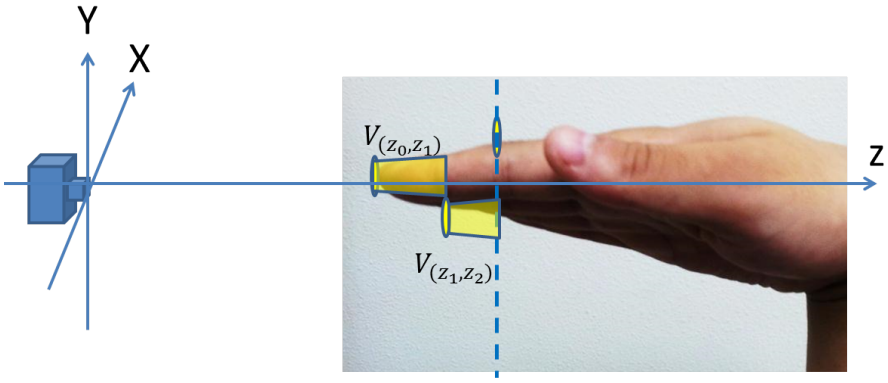

図 3.10: z_1 に対応する画素の四角錐台の体積計算

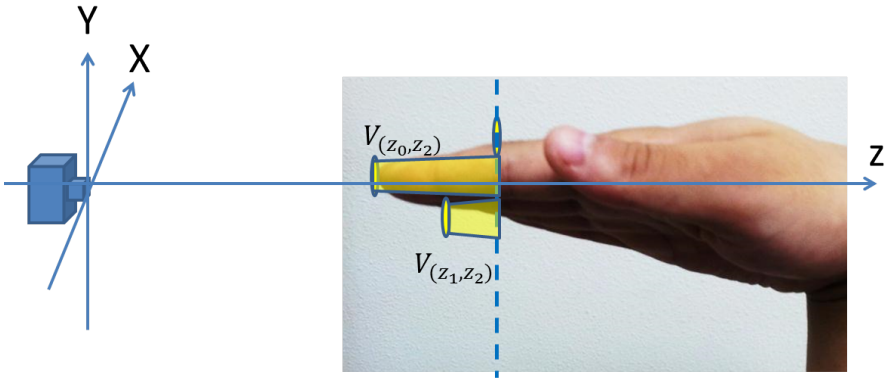

図 3.11: z_0 に対応する画素の四角錐台体積の更新

i を増やしながら計算を繰り返し，総和が図 3.12 のようにしきい値 V_t を超えた時点の距離 z_t の位置で止める．このように i を 1 ずつ増やしながら求めた各画素に対応する四角錐台の体積の合計を

$$V(i) = \sum_{j=0}^{i-1} V(z_j, z_{i+1}) \tag{3.6}$$

で計算する．本研究では，この $V(i)$ で求まる体積が実際の手の体積の半分程度になるときに手領域をうまく抽出できると仮定した．つまり，しきい値 V_t を手の約半分の体積に設定して，i を 1 ずつ増やしながら $V(i)$ と V_t を比較する．そして，$V(i) > V_t$ となった時に処理を停止して，z_0 から z_i に対応する画素を手領域として抽出する．この方法は物体の体積をもとに手領域を抽出するため，指文字を表現した際の手の位置によって抽出される領域が変化することがなく，距離による影響を受けにくいという利点がある．

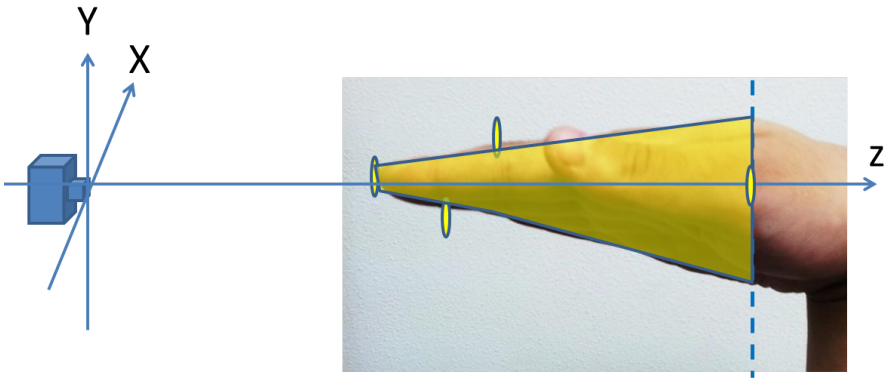

図 3.12: 四角錐台の体積の合計から手領域の体積計算

3.4　手領域の手型認識

3.4.1　手型認識の概要

抽出した手領域について指文字の手型を識別する．手型識別の流れは次の通りになる．各処理の具体的な内容についてを図 3.13 に示す．

1. 指文字の入力
2. 距離画像の前処理
3. 特徴量を計算
4. 識別器による手型認識
5. 認識結果の出力

指文字の入力は，抽出した手領域の距離画像とする．入力した画像に対し，前処理として 16×16 の大きさにスケーリングを行った後，特徴量を抽出する．距離画像は距離値を元に一度グレースケールの輝度画像に変換してあり，スケーリング後の画素に格納されている輝度値を特徴ベクトルとして認識に用いた．識別器は k 近傍法を採用し，学習データと照合し認識結果を出力する．

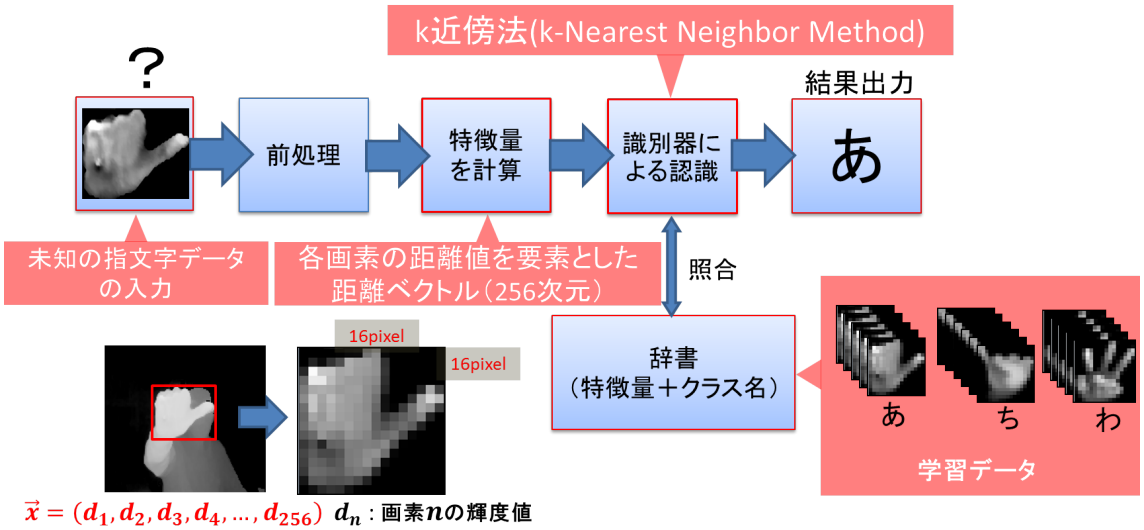

図 3.13: 手領域の手型識別の概要

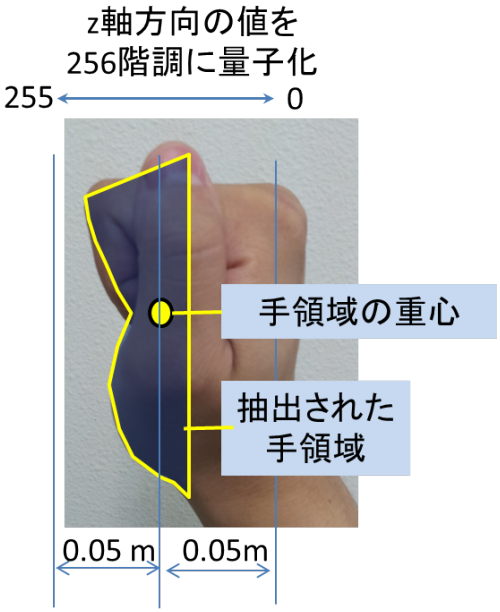

図 3.14: z 値の量子化による CG 面の輝度値の設定

3.4.2　特徴量の導出

特徴量を計算するため，距離画像に対して次のような処理を行った．まず，抽出した手領域の各画素の 3 次元頂点の座標値を取得する．次に，その頂点群から面を形成し，OpenGL を用いて 256 × 256 の CG 画像としてレンダリングを行う．手領域の座標群の中心である重心から z 軸方向の前後の範囲を設定し，指文字表出者の手が収まるようにする．このときの範囲の奥行き値，つまり，z 軸方向の値を 256 階調に量子化し，その数値をグレースケールの輝度画像の輝度値として設定した CG 画像を生成した．本研究では図 3.14 のように，重心の位置から前後 0.05m を z 軸方向の幅とした．

次に CG 画像全体を 16 × 16 の大きさのブロック領域に分割し，それぞれの領域の輝度値の平均を画素に格納した 16 × 16 の画像を生成する．このときの各画素の輝度値を要素とした，256 次元の特徴ベクトルを手型の識別に用いた．

3.4.3　k 近傍法を用いた手型の識別

3.4 節で述べた方法で抽出した手領域の特徴ベクトル $\vec{x}_n$ から手型を識別するために図 3.15 のような k 近傍法 [22] を用いた．k 近傍法はパターン認識の分野の中では最も単純な線形識別器である．本研究では距離画像を用いた指文字認識手法の基礎的な検討として，この識別器を用いたときの認識率の評価を行った．

抽出された手領域の特徴ベクトルに対し，その手型のクラスをタグ付けしたものを 1 つの学習データとする．その学習データの特徴ベクトルを $\vec{C}_n = (d_1, d_2, \ldots, d_{256})$ とする．このときの n は，そのクラスでタグ付けされている特徴ベクトルの数を表す．今，入力データとして未知のクラスの指文字の特徴ベクトル $\vec{x}_{input} = (f_1, f_2, \ldots, f_{256})$ が与えられたとき，その特徴ベクトルと辞書に登録されている特徴ベクトルとのユークリッド距離 D を次の式で計算する．

$$D = \sqrt{\sum_{i=1}^{256}(d_i - f_i)^2} \tag{3.7}$$

すべての特徴ベクトルに対して計算したとき，最も距離が小さかった順に学習データをソートし，先頭から上位 k 個までのクラスについて出現数が最も多かったものを，入力データのクラスの推定結果として出力する．図 3.15 の例では，ユークリッド距離が近かった上位 5 個のうち「あ」のクラスが最も多いので，それを指文字のクラスの推定結果としている．

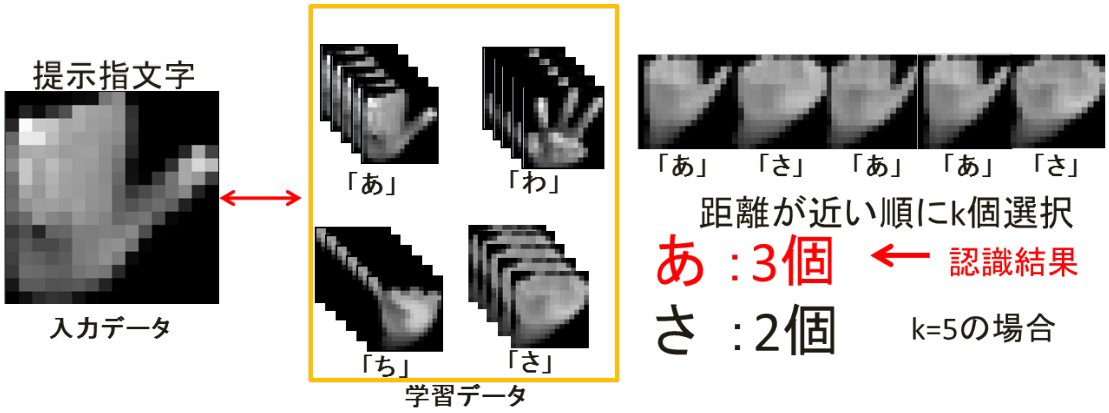

図 3.15: k 近傍法による手型の識別

3.5　手領域の動作検出

3.5.1　手領域の動作検出の概要

手領域の動作検出は，手領域が移動した方向の決定と，その方向に移動した量の計算を行う．移動方向と，移動量を組み合わせて，濁音，半濁音，小書き文字の，どの動きに該当するのか検出を行う．ここでは，ユーザが右手で指文字を出す場合についての処理方法について述べる．

3.5.2　手領域の移動方向の検出

手領域の移動方向を調べるために，主成分分析（PCA，Principal Component Analysis）を用いる．手領域の 3 次元座標群の重心を求め，連続した N_f フレーム分保存する．この N_f 個の重心群に対して PCA をかけると，図 3.16 のように分散が最も大きい方向が得られる．この方向は第一主成分と呼ばれ，手領域が移動した方向とする．図 3.17 で示すように，指文字を提示している手領域をカメラから見たときに，x 軸の左方向，y 軸の上方向，z 軸の奥方向をそれぞれ濁音，半濁音，小書き文字の動作方向であるとして $\vec{u}_i(i=0,1,2)$ を，該当する手型の動きのない指文字の動作方向として $\vec{u}_i(i=3,4,5)$ をそれぞれ定義する．ここでは，$\vec{u}_0$ は濁音方向，$\vec{u}_1$ は半濁音方向，$\vec{u}_2$ は小書き文字の移動方向と定義した．

PCA の第一主成分方向の単位ベクトル $\vec{v}$ とこれらの軸とのなす角が最小になるような軸 $\vec{u}'$ を選択する．ここで選択された軸を手型の動作方向の候補として決定する．$\vec{u}'$ を選択するために，

$$\max\{\vec{v} * \vec{u}_i\} \tag{3.8}$$

$$\vec{u}_i = \begin{cases} \vec{u}_0, \vec{u}_1, \vec{u}_2 & (\text{濁音，半濁音，小書き文字に該当する方向}) \\ \vec{u}_3, \vec{u}_4, \vec{u}_5 & (otherwise) \end{cases}$$

を満たす $\vec{u}_i$ を求め，この $\vec{u}_i$ を $\vec{u}'$ と置く．$\vec{u}_3$〜$\vec{u}_5$ が検出された場合は，動きがないと判定する．

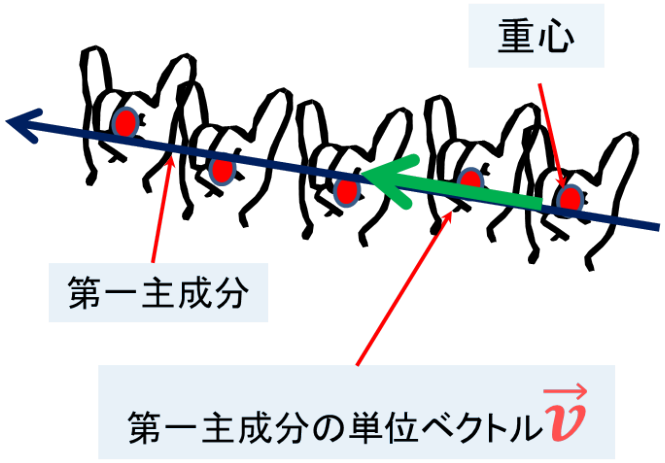

図 3.16: 手領域の移動方向の決定

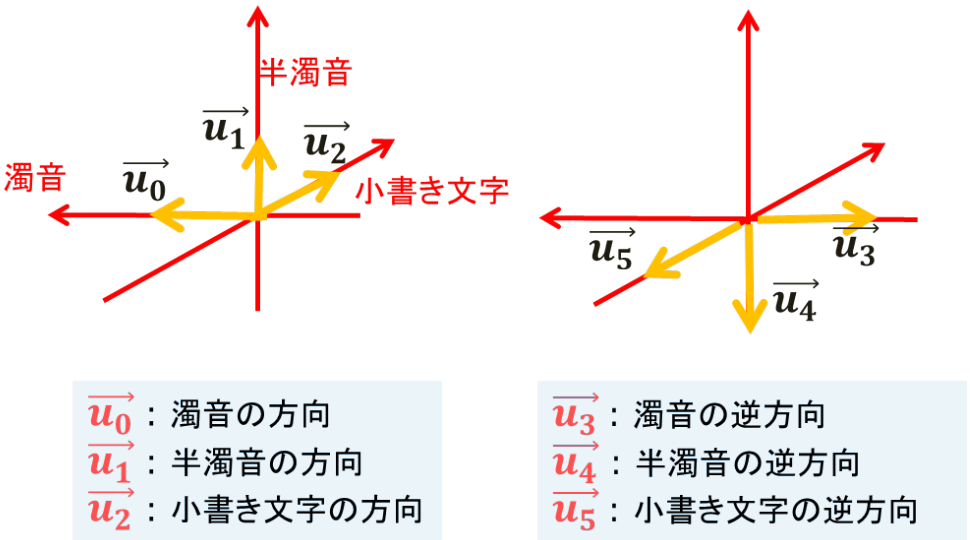

図 3.17: 手領域の動きのある指文字の軸

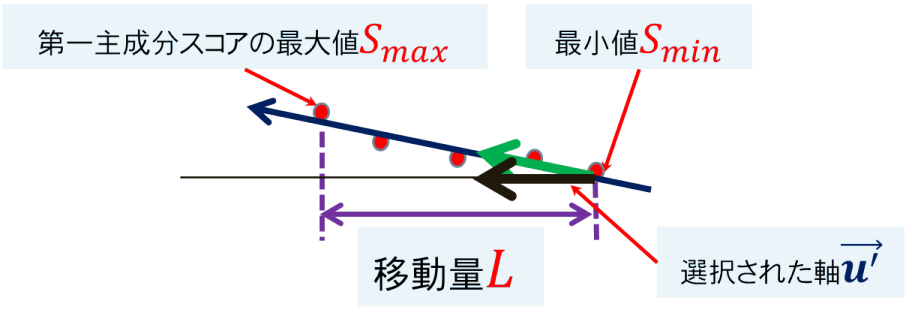

図 3.18: 手領域の移動距離の計算

3.5.3　手領域の移動量計算

手型を静止して表現する指文字「か」と，その手型を固定して横方向に動かして表現する「が」を区別するためには，抽出された手領域が動いているか否かを判断する必要がある．本研究では手領域が一定距離動いていれば手型の動きのある指文字を表現していると判断する方法を考案した．

式 3.8 で候補として選択された $\vec{u}'$ を用いて手型の移動量である L を求める．図 3.18 で示すように，第一主成分のスコアの最大値を S_{max}，最小値を S_{min} とすると，移動距離 L は，

$$L = (S_{max} - S_{min})\vec{v} * \vec{u}' \tag{3.9}$$

で計算できる．この移動距離 L が，移動距離のしきい値 L_t に対して $L < L_t$ の場合は表現している手領域が静止していると判定し，$L \geq L_t$ の場合は手領域が動いていると判定する．

3.5.4　手型の動きのある指文字の認識方法

手型の動きのある指文字を認識する際の処理イメージを図 3.19 に示す．まず最初に，ある指文字の表現をしたときにその手型の認識を行う．次に，過去 N_f フレーム分の重心群から動作検出を行う．この 2 つの結果を組み合わせることで動きのある指文字の認識を行う．例えば，手型が「ぎ」と認識されている間に，動作検出でうごきがないと識別されればその 2 つを組み合わせて「き」と識別し，横方向に動作があれば「ぎ」として認識する．

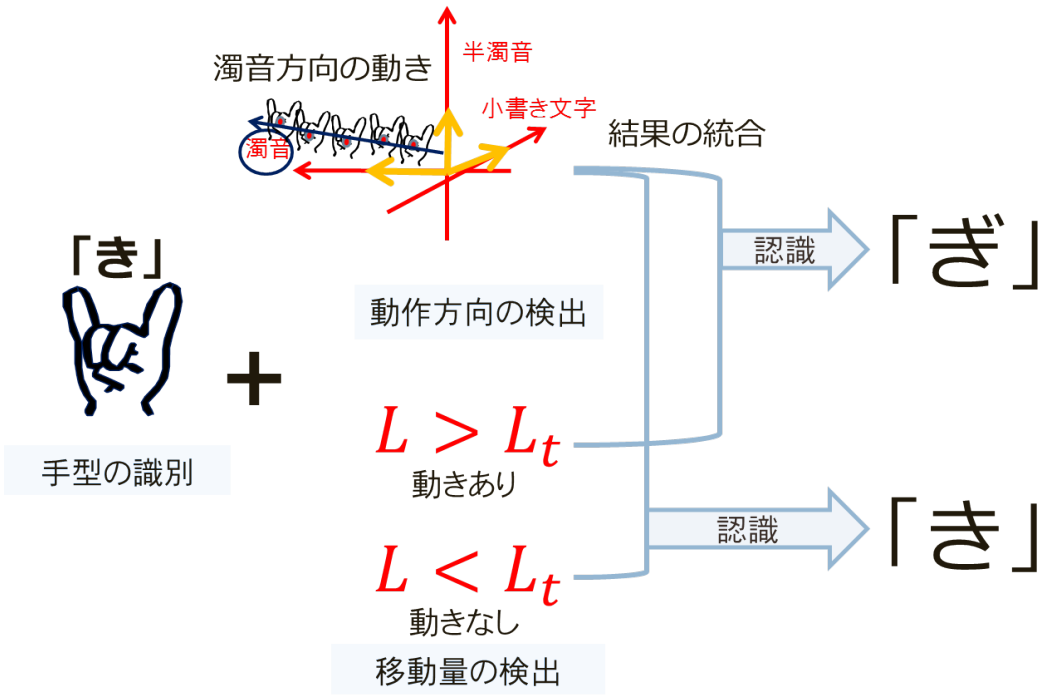

図 3.19: 手型の動きのある指文字の認識処理の例

3.6　認識実験 1

3.6.1　実験目的

手領域の抽出処理および手領域の動作検出処理が正常に適用されているかどうかを確認するために，図 1.1 で示した清音指文字のうち手型を静止して表現する指文字 41 文字と，動きのある指文字の濁音，半濁音，ならびに拗音に用いる小書き文字の指文字 34 文字の計 75 文字を対象として認識実験を行った．

3.6.2　実験の概要と条件

撮影は，昼間の明るい時間に室内の照明を付けた状態で行った．カメラは図 3.20 のように床から 1m の高さにして固定し，撮影環境の位置関係に変化が生じないようにした．指文字の撮

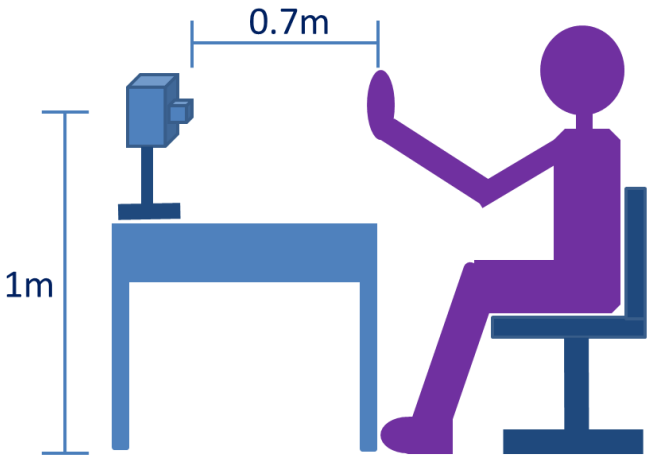

図 3.20: 指文字を撮影する環境

影を行う際に，被験者は，椅子に深く腰掛けた状態で指文字を提示する．指文字はカメラから 0.7m 程度離れた場所で，掌がカメラに対して正面を向くようにした．被験者 1 名が提示した指文字を各クラス 10 枚ずつ保存（計 750 枚）したものを k 近傍法の学習データとして使用した．学習データを取得した人と同一者が 1 クラスにつき 20 回の指文字の提示を行い，その認識率を調べた．保存した重心のフレーム数 $N_f = 5$[frame]，移動量のしきい値 $L_t = 0.1$[m] とした．k 近傍法の k は 7 とし，ユークリッド距離が近かった上位 7 個のデータから識別結果を決定した．

実験の実施には，OS は Windows，CPU は Intel Core i7-2620M（2.70GHz），メモリ 8.00GB を搭載した計算機を使用した．指文字認識のプログラムを実装し動作を確認した結果，1 回の認識処理を終えるのに約 0.18 秒かかった．

3.6.3　実験結果と考察

3.6.2 節で述べた実験環境条件で認識を行った結果を，表 3.3 に示す．対象とした指文字全体の平均認識率は約 0.85 となった．表中の※印がついている数値は，認識率が全体の平均認識率よりも低かったものを示す．

表 3.3 の結果のうち，特に認識率が低かったのは手型が類似した静止指文字である．具体例として,「ひ」と「ら」については図 3.21 のように相互に誤認識し合ってしまい認識率が低かった.「い」,「そ」,「ぬ」,「ま」については，図 3.22 のように，それぞれ「ち」,「は」,「ろ」,「ね」に誤認識され，認識率が低かった．従来の静止指文字認識を試みた方法でも，程度の差はある

表 3.3: 手型識別と動作検出による指文字認識結果

クラス	清音	濁音	半濁音	小書き	クラス	清音	濁音	半濁音	小書き
あ	1.00	−	−	1.00	な	1.00	−	−	−
い	0.40	−	−	0.00 ※	に	1.00	−	−	−
う	1.00	−	−	1.00	ぬ	0.20 ※	−	−	−
え	1.00	−	−	1.00	ね	1.00	−	−	−
お	1.00	−	−	1.00	は	1.00	0.40 ※	0.40 ※	−
か	1.00	1.00	−	−	ひ	0.35 ※	0.60 ※	0.80 ※	−
き	1.00	1.00	−	−	ふ	1.00	1.00	1.00	−
く	1.00	1.00	−	−	へ	1.00	1.00	1.00	−
け	1.00	0.80	−	−	ほ	1.00	1.00	1.00	−
こ	0.80 ※	1.00	−	−	ま	0.40 ※	−	−	−
さ	1.00	1.00	−	−	み	0.85	−	−	−
し	1.00	0.50 ※	−	−	む	0.90	−	−	−
す	0.60 ※	0.40 ※	−	−	め	1.00	−	−	−
せ	1.00	1.00	−	−	や	1.00	−	−	1.00
そ	0.00 ※	0.80 ※	−	−	ゆ	1.00	−	−	1.00
た	0.80 ※	0.80 ※	−	−	よ	1.00	−	−	0.80 ※
ち	1.00	0.50 ※	−	−	ら	0.65 ※	−	−	−
つ	0.85	0.40 ※	−	−	る	1.00	−	−	−
て	1.00	1.00	−	−	れ	1.00	−	−	−
と	0.80 ※	1.00	−	−	ろ	0.85	−	−	−
					わ	1.00	−	−	1.00
					Avg	0.86	0.81	0.84	0.87

全体の認識率平均：0.85

が同様な誤認識は発生し課題となっている．

これらの類似した手型の指文字で誤認識が生じる原因として，識別器の識別能力の限界，学習データ不足や，特徴ベクトルの特徴を表す性能不足が考えられる．本報告で使用した識別器は最も単純かつ基本的なｋ近傍法であり，特徴ベクトルは画素の輝度値を単にベクトル化したものであるため，先に述べた手型を十分に識別できなかった可能性がある．静止指文字の手型を精度良く識別できる従来の方法を活用することによって，誤認識を低減することができると考える．

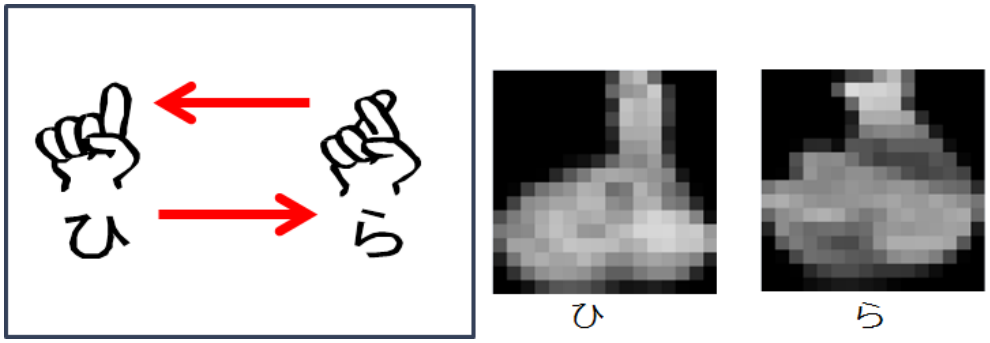

図 3.21: 相互に誤認識した例

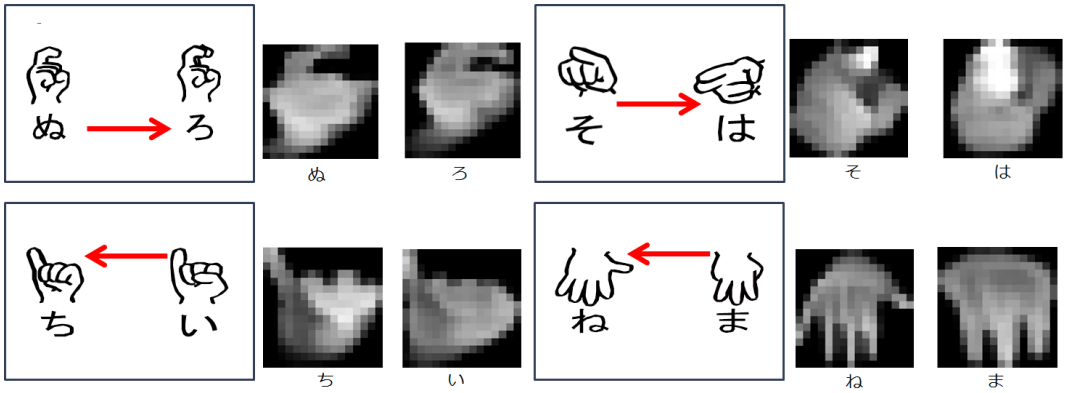

図 3.22: どちらか片方に誤認識した例

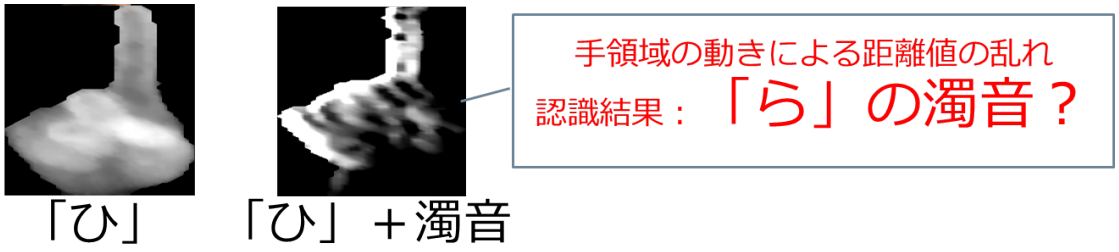

図 3.23: 手領域の動きによる誤認識

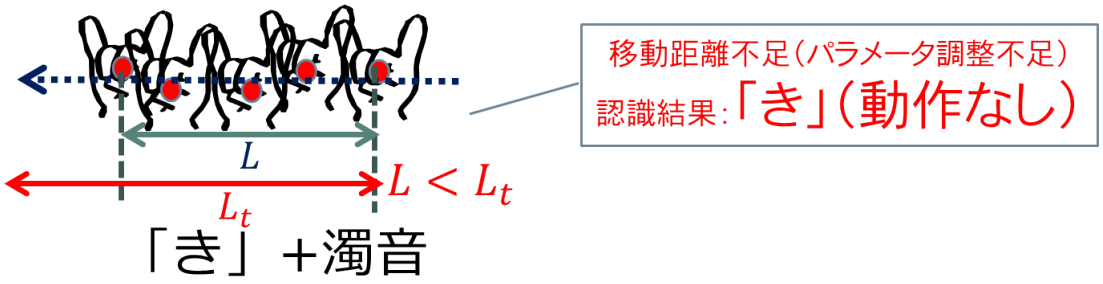

図 3.24: パラメータ調整不足による誤認識

手型の動きのある指文字を誤認識してしまうパターンは，図 3.23 のように動作中の手型が誤認識されてしまう場合と，図 3.24 のように移動距離が足りず静止指文字として認識されてしまう場合の 2 つがあった．前者は，手型の動作中に移動方向に対して距離値が乱れてしまうことが原因である．これは，TOF カメラで距離を計測するときに手が高速に動いていると発生し，光学式のカラーカメラにおけるぶれと同様なものと考えられる．例えば「び」は，人差し指 1 本を立てた「ひ」の手型を横方向に動作させるが，人差し指と中指を立てて交差させる「ら」に誤認識されることがあった．後者は，手型の移動距離のしきい値 L_t の調整不足が原因である．

これらの誤認識を減らすために，各問題について次のような解決方法を考えた．動きによる距離値の乱れについては，図 3.23 の 2 つの画像を比較してみると，静止指文字と動きのある指文字は明らかに異なる距離画像になっていると言える．そこで，この乱れの特徴を考慮した特徴量を用いれば，静止指文字および動きのある指文字の 2 つを区別できるのではないかと考え，改良を試みた．この距離値の乱れのことを，動作ノイズと呼ぶことにする．手型の動作検出で手型の動きのある指文字を表現する際の移動量は，5 フレームあたり 0.1m 程度であると予想してパラメータ L_t を設定していたが，実際は最適なパラメータを実験的に設定しなければならな

かったことが考えられる．保存する重心のフレーム数と移動量のパラメータの組み合わせは動作検出の精度に大きく影響を与えていると考えられるため，この 2 つのパラメータの組み合わせをいくつか試すことで，一番安定して動作を検出できる数値を決めることで解決する．

その他に認識率を改善する方法として，特徴ベクトルに対し主成分分析をかけることで次元圧縮して最適化を行うことによって認識を行う方法が考えられる．実際にこの方法を試して認識率を評価した結果については，付録 A.1 で述べる．

3.7　まとめ

本章では，距離画像を用いた動きのある指文字の認識方法について述べた．まず，TOF カメラから最も近い物体の体積をもとにして動的に手領域を抽出する方法について述べた．次に，距離値にもとづく特徴量を計算し，手型識別と動作検出を組み合わせた指文字の認識方法による認識率を評価した結果について述べた．全体の認識率は 0.85 となり，手領域の抽出および動作検出の処理が正常に行えていることが確認できた．しかし，手型の動きのある指文字については，動作ノイズおよびパラメータ調整不足が原因で誤認識を起こしてしまうことがわかった．

次章では，本章の結果を踏まえて動きのある指文字の認識率を改善させるため，動作ノイズを考慮した特徴量とより精度の高い識別器を採用した手型識別の改良，およびパラメータを最適化した動作検出と改良した手型識別とを組み合わせた場合の認識率を評価した結果について述べる．

第4章　動きのある指文字認識の改良[A4]

4.1　概要

第 3 章で述べた手法では，次の 2 つの課題があった．

- 移動方向に対する動作ノイズが原因で指文字の誤認識が生じる
- パラメータの調整不足により動作検出に失敗する

そこで，図 4.1 に示す特徴量と識別器の部分を変更することでこれらの課題の解決を試みた．パターン認識の性能に優れる高次局所自己相関特徴（HLAC）を採用し，動作ノイズを含んだ距離画像も特徴として学習に用いることで，手型の動きのある指文字の誤認識の問題を改善する．この特徴量に対し，非線形なパターンにも強く，物体認識の分野で精度が高いと言われるサポートベクターマシン（SVM）を用いて手型の識別を行う．動作検出に失敗する問題には，パラメータの数値の組み合わせを試し最適化を行うことで対応する．動作検出精度を向上させた上で手型識別と合わせることで，動きのある指文字の認識精度の向上を目指す．

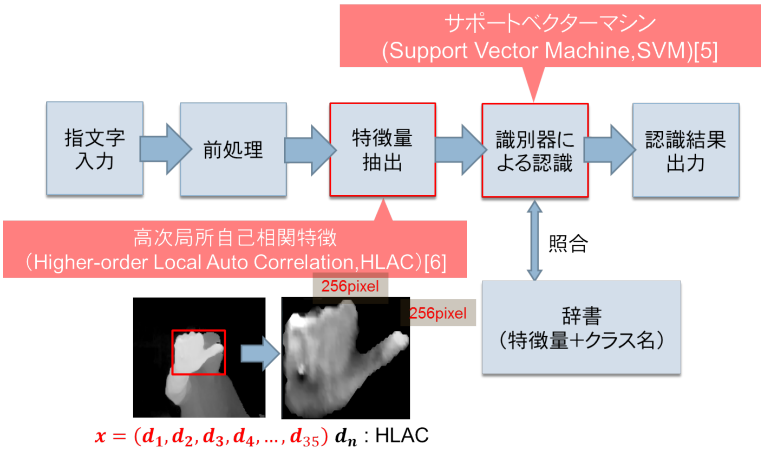

図 4.1: HLAC と SVM を用いた手型識別

4.2 動作ノイズを考慮した特徴量による動きのある指文字認識

4.2.1 動作ノイズを考慮した特徴

第 3 章では静止指文字の特徴量と動作検出を組み合わせて手型の動きのある指文字を認識した．しかし，図 3.23 で示した通り，手型が同じであっても動きがある場合は，動作ノイズの影響で距離画像に違いが生じていることが明らかになった．本研究では，動作ノイズを含んだ距離画像も指文字表現の特徴と考え，動作ノイズを考慮した特徴量を識別器の学習データとして利用した．具体的には，静止指文字「か」の特徴量と，動きのある指文字「が」の特徴量をそれぞれ別のクラスと考えて識別器の学習データとして登録し，動作検出の結果を用いずにそれぞれの指文字が識別可能かどうか実験を通して調査した．

動きのある指文字を表現する際，カメラと手領域の間の距離は常に変化し，距離画像中の手領域の大きさも変化してしまう．特に小書き文字については，手型をユーザから見て手前に動かすため，手領域の大きさの変化が顕著である．一方，HLAC は認識対象の位置不変性および加法性を満たす特徴量だが，大きさについては特徴量が不変ではない．つまり，手型の動きによって特徴量が大きく変化してしまう可能性がある．そこで，抽出した手領域の距離画像を図 4.3 のように 256×256 のサイズで CG 画像としてレンダリングし，サイズを正規化した上で HLAC を求めた．

4.2.2 高次局所自己相関特徴（HLAC）

指文字の特徴量は，高次局所自己相関特徴 (Higher-order Local Auto Correlation, HLAC)[23] を採用した．HLAC は，パターン認識において重要視されている位置不変性および加法性を満たす統計的な特徴量として知られている．対象となる画像の領域内の位置 $r = (x, y)$ における画素値を $f(r)$ とすると，その周囲への N 次の変位 $a_1, a_2, \ldots, a_N$ に対して次のような式で表現する事ができる．

$$x^N(a_1, \ldots, a_N) = \int f(r) f(r + a_1) \cdots f(r + a_N) \tag{4.1}$$

N 次の変位は図 4.2 のような 25 個のパターンで表現される．あるパターンについて見たときに，1 つの画素とその周囲の対応する画素の画素値を足しあわせ，画像全体を走査しながら結果を

掛けていったものを特徴ベクトルの 1 つの要素として扱う．濃淡画像画像では変位方向の重複を考慮すると全部で 35 個のパターンが存在するため，ベクトル $\vec{x} = (d_1, d_2, \ldots, d_{35})$ を特徴量として認識に用いることができる．

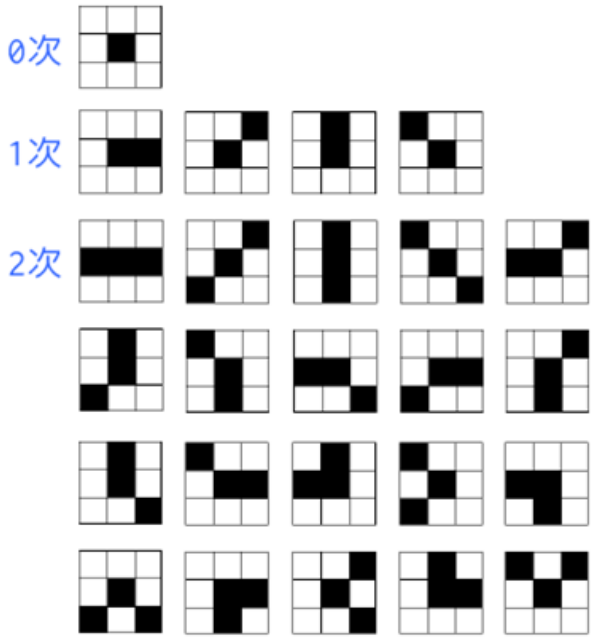

図 4.2: HLAC の局所パターン

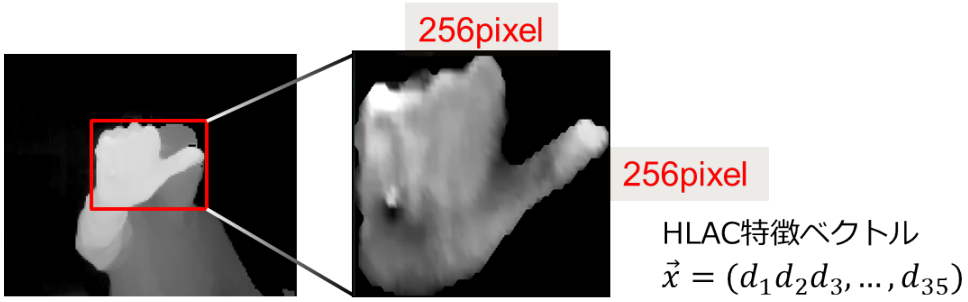

図 4.3: 256 × 256 にスケーリングした画像から HLAC 導出

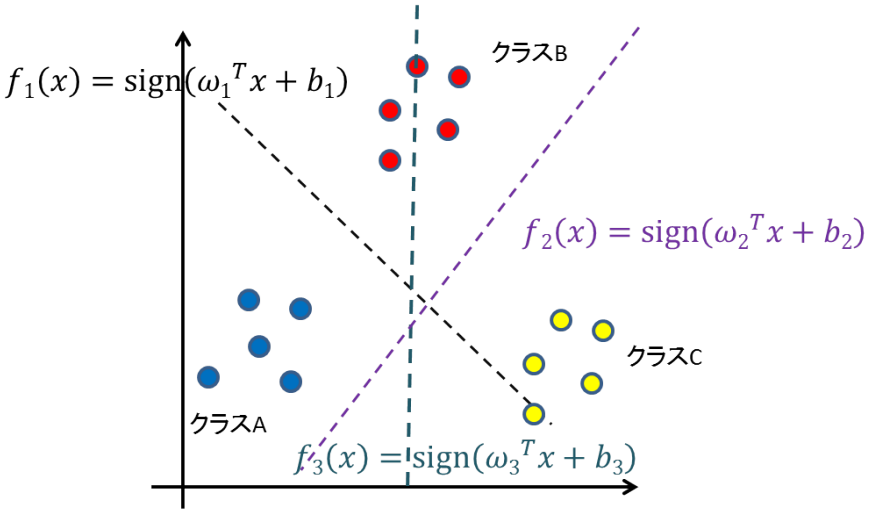

図 4.4: 1 対 1SVM（3 クラス分類）の例

4.2.3　サポートベクターマシン

サポートベクターマシン (Support Vector Machine, SVM)[24] は，現在知られている多くの手法の中で最も認識性能が優れた学習モデルの一つで，線形入力素子を利用して 2 クラスのパターン識別器を構成する手法である．

ある学習サンプル群 $\mathbf{x}(x = x_1, x_2, \ldots, x_n)$ があり，各サンプルは $y_i = +1$ と $y_i = -1$ の信号で分類されているとする $(i = 1, 2, \ldots, n)$ と，未知のクラスの入力データを分類する式は $f(x) = \mathrm{sign}(\omega^T * x + b)$ で表される．$g(x) = \omega^T * x + b$ は学習サンプル群を 2 つに分ける識別面であり，$g(x) \leq 0$ のときに $f(x) = -1$，$g(x) \geq 0$ の時 $f(x) = +1$ に識別ができる．$g(x) = -1$，$g(x) = +1$ となる面は識別面との距離が等しくなり，これらの面の上に存在する点をサポートベクターと呼ぶ．また，ω は重みベクトル，b はバイアスを表しており，ω と b の 2 つのパラメータは，識別面とサポートベクターまでの距離が最大になるように二次計画問題を解くことによって求める．

通常，SVM は 2 クラスの分類にしか用いることができないため，本研究の対象とする指文字を識別させるためにマルチクラスに拡張させる必要がある．そこで，図 4.4 のように複数クラスの学習データ群のうち 2 クラスの組み合わせすべてについて識別器を構成する．未知の入力データが与えられたとき，すべての識別器にかけた結果に対して多数決を行いクラスを決定す

る，1 対 1SVM（One-versus-One SVM[24]）を用いて認識を行った．

4.3 認識実験 2

4.3.1 実験目的

3.6.3 節で述べたように，距離センサで動きのある指文字を撮影した場合に，その動き方向に動作ノイズが生じてしまい，誤認識の原因となっていることがわかった．そこで，本実験では動作ノイズが現れている距離画像についても HLAC を計算し，濁音，半濁音，小書き文字の指文字のクラスを設けて学習データとして使用した．

動作ノイズを考慮した特徴量を用いて指文字を認識させた場合，静止指文字と手型の動きのある指文字の 2 つを区別が可能であると考えられる．そこで，本実験では「あ」行,「か」行,「さ」行,「は」行,「や」「ゆ」「よ」の 23 文字，濁音と半濁音の,「が」行,「ぱ」行の 10 文字，および小書き文字の「ゃ」「ゅ」「ょ」の 3 文字の，計 36 文字を対象として認識実験を行った．「か」行と濁音の「が」行,「は」行と半濁音の「ぱ」行,「や」行と小書き文字の「ゃ」,「ゅ」,「ょ」を手型識別のみで区別できるかどうかを調べた結果について述べる．

4.3.2 実験の概要と条件

指文字の撮影を行う際に，被験者は椅子に深く腰掛けた状態で指文字を提示する．カメラは地面から約 1m の位置に設置し，指文字は 0.5m 程度離れた場所で，手のひらがカメラに対して正面を向くようにした．被験者 1 名が提示した指文字の距離画像を連続的に撮影し，指文字ごとに 20 フレームを 1 セットとして HLAC を計算した．10 セットずつ作成した各指文字の HLAC 群のうち，5 セットを SVM の学習データとして使用し，残り 5 セットを評価用のデータとして識別器にかけ，クラスごとの平均認識率を求めた．

4.3.3 実験結果と考察

手型を認識させた結果を表 4.1 に示す．対象とした指文字の全体の認識率の平均は約 0.85 となった．そのうち，特に認識率が高かった指文字は「き」と「ぎ」や「く」と「ぐ」の組み合

表 4.1: 動作ノイズを考慮した手型識別による認識結果

クラス	認識率	クラス	認識率	クラス	認識率
あ	0.92	か	0.63	が	0.98
い	0.91	き	0.91	ぎ	1.00
う	0.98	く	0.96	ぐ	0.96
え	0.98	け	0.78	げ	0.90
お	1.00	こ	0.84	ご	0.96
さ	0.84	は	0.78	ぱ	0.88
し	1.00	ひ	0.88	ぴ	0.89
す	1.00	ふ	0.54	ぷ	0.81
せ	0.77	へ	0.93	ぺ	0.81
そ	0.67	ほ	0.65	ぽ	0.81
AVG	0.91	や	0.61	ゃ	0.81
		ゆ	0.78	ゅ	0.75
		よ	0.77	ょ	0.75
		AVG	0.77	AVG	0.81

全体の認識率平均：0.85

わせである．特に認識率が低かったのは「か」「け」「は」「ふ」「ほ」「や」であった．いずれも認識率が 0.8 を下回っているが，対応する動きのある指文字である「が」「げ」「ぱ」「ぷ」「ぽ」「ゃ」については認識率が高いことがわかる．つまり，動作ノイズを考慮した特徴量が有効に働いていると考えられる．

認識率が低かった原因は，SVM で指文字を学習させる際のパラメータの調整不足が原因であると考えられる．しかし，動作検出による動きの有無の判定と組み合わせれば，識別器に関係なく認識率の改善が可能であると考えた．そのためには，計算に用いるパラメータの調整を行い，動作検出の精度を改善させることが必要となる．次節では，実際にいくつかの数値の組み合わせをパラメータとして設定し，動作検出検出のためのパラメータを調整する方法について述べる．

4.4 動作検出パラメータの調整

4.4.1 パラメータの調整方法

3.5 節で述べた手領域の動作検出では，フレーム数 N_f と移動距離のしきい値 L_t の 2 つのパラメータを筆者の経験から十分に認識可能であると予想した数値に設定していたために，指文字を表現した際に移動量がしきい値に満たない場合が生じてしまい，誤認識を引き起こしてしまうことがわかった．そこで，これらのパラメータの値を調節したときの動作検出の結果について調査を行った．図 4.5 のように，手型を動かす指文字を提示している最中のあるフレームを見たときに，そのフレームを含めた過去 N_f フレームの指文字の重心を保存し，移動方向および移動量を計算した後に移動量 L とそのしきい値 L_t を比較して動きの有無を判別する．

4.3 節の実験で学習に使用した連続で指文字を撮影したすべてのフレームのうち 1 セットを元に動作検出の精度について調査した．調査するフレームは，1 セットのフレーム全体の中でそれぞれのクラスの指文字を表現している連続する 20 フレームとし，その各フレームについて動作検出処理を行う．動作検出に成功した場合は，検出できたフレームが連続して表れるため，その最大フレーム数を調査した．N_f と L_t の数値を変化させると，その最大フレーム数も変化するため，最も安定して検出できるような数値の組み合わせについて検討した．

4.4.2 最適なパラメータの予想

距離画像で指文字を撮影した際の 1 フレームあたりの移動距離 l_u[m/frame] は，手領域の移動速度 s[m/s] とカメラのフレームレート f_c[frame/s] によって，

$$l_u = \frac{s}{f_c} \tag{4.2}$$

で表される．一方，手領域の動作検出のしきい値 L_t を N_f で割ると，1 フレームあたりの移動距離のしきい値として見ることができる．この L_t/N_f は，l_u に置き換えることができ，手領域の移動速度 s とカメラのフレームレート f_c から，最適なパラメータ L_t/N_f を導き出すことができると考えられる．4.3 節の実験ではカメラのフレームレートの実測値は約 25fps，手型の移動速度の平均は約 0.08m/s であった．これらを式 4.2 に代入すると．$lu = 0.08 \ / \ 25 = 0.0032$ と

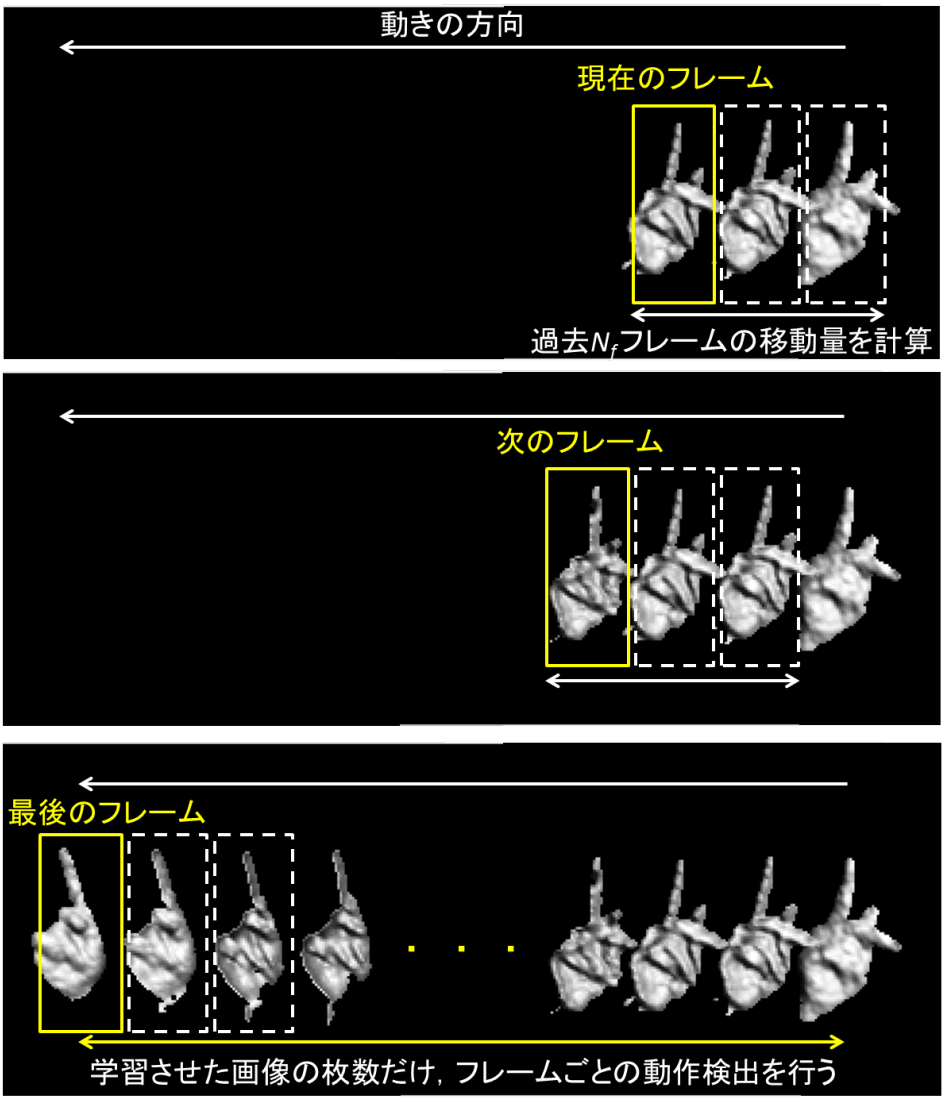

図 4.5: 動作検出のためのフレーム選択

なる．つまり，L_t/N_f がおよそ 0.0032 となるパラメータが設定されたとき，最も動作検出の精度が高くなると予想した．

4.4.3　パラメータ調整結果

表 4.2 のように，設定した Nf と Lt のパラメータの組み合わせそれぞれについて，手型に動きのある指文字表現の 20 フレームに対して動作検出を行い，連続的に正しく動作検出された最大フレーム数の平均を同表に示す．

パラメータの項目は，動作検出実験を行った際の N_f と L_t の組み合わせを示している．動作検出したフレーム数の項目は，濁音，半濁音，小書き文字の指文字をそれぞれすべて提示した際に，動作検出に成功した連続最大フレーム数を平均した数値を示す．N_f＝ 10 と L_t＝ 0.03 や，N_f＝ 6 と L_t＝ 0.02 の組み合わせを選択すれば濁音，半濁音，小書き文字を十分に認識可能であった．なお，いずれの組み合わせにおいても静止手型に動きがあると誤認識される場合はなかった．

これらの結果から，パラメータ N_f と L_t が次の条件を満たしたときに最も安定して動きを認識することができた．

$$\frac{L_t}{N_f} = 0.0030 \sim 0.0033 \fallingdotseq 0.0032 \tag{4.3}$$

この結果は，式 4.4.2 で述べた最適な結果が得られると予想した条件と合致する．認識結果が安定しなかったときのパラメータは次の通りになる．

- $L_t/N_f \gg 0.0032$：手領域が動作していても静止していると誤認識
- $L_t/N_f \ll 0.0032$：手領域が静止していても動作していると誤認識

前者の場合は指文字を提示したとしても認識に成功したフレーム数が他の数値設定よりも少なくなってしまい，後者の場合は逆に手型を少し動かしただけでも過敏に反応し動きがあると認識してしまう．式 4.3 の条件に近いパラメータ設定の場合に認識フレーム数が多く，かつ安定した認識が可能であることが確認できた．

3.6 節では，パラメータはそれぞれ $N_f = 5$，$L_t = 0.10$ に設定して認識実験を行った．L_t/N_f は 0.02 となり，式 4.3 で求めた条件よりも大きな数値となった．これは，認識結果が安定しなかったパラメータの条件のうち，手領域が動作していても静止していると誤認識するパターン

表 4.2: パラメータ調査結果

パラメータ			動作検出したフレーム数		
N_f[frame]	L_t[m]	L_t/N_f	濁音	半濁音	小書き文字
10	0.05	0.0050	5.2	0.3	5.3
10	0.03	0.0030	5.8	2	5.6
10	0.02	0.0020	6.5	3	6.6
6	0.05	0.0083	3.2	0.3	5.2
6	0.03	0.0050	5.4	0.5	5.4
6	0.02	0.0033	6.8	2	5.9
5	0.05	0.0100	5.2	1	6.3
5	0.03	0.0060	6.3	2	6.3
5	0.02	0.0040	6.5	3	6.7

に当てはまる．つまり，3.6 節で行った実験では条件に適さないパラメータを設定したことが原因で，動きのある指文字の誤認識が生じてしまったことが明らかになった．

4.4.4 考察

表 4.2 より，半濁音の指文字を表現した場合のみ他の指文字に比べて認識に成功したフレーム数が少ないことがわかる．例えば，最適なパラメータの 1 つである $N_f = 6$，$L_t = 0.02$ のとき，濁音と半濁音は平均で約 6 フレーム以上は動作検出に成功しているのに対し，半濁音は平均で 2 フレームしか成功していない．これは，濁音，半濁音，小書き文字を表現する際の移動量がそれぞれ異なるためである．例えば，半濁音の表現が他の指文字の表現よりも移動量が極端に少なかった場合は，同じ移動量で表現したつもりでも識別に成功するフレーム数が少なくなってしまう．より識別精度を高めるためには，濁音，半濁音，小書き文字の軸の方向それぞれで，移動量のしきい値の最適化を行うことができれば改善できると考えられる．

4.4.5 動作ノイズを考慮した手型識別と動作検出による指文字認識

3.6 節での認識実験の結果では，パラメータの調整が不十分であったために，動きのある指文字の動作検出が正常に行われなかった．例えば,「が」のような動きのある指文字を提示して

も,「か」の静止指文字と誤認識されてしまう場合があった．これらの指文字は手領域の動作検出と組み合わせて認識結果を決定すれば認識率を改善することができる．本節では，手型識別の結果と動作検出の結果が得られたときに，動作検出の優先度を高くして最終的な認識結果を出力する方法を提案する．例えば図 4.6 のように，手型が「が」と認識されているときに動きの認識では動きがないとされた場合は，最終的な認識結果を「が」ではなく「か」に修正する．一方，手型が「か」と認識されている場合でも，動きがあるとされた場合は「が」に修正することで，双方の認識率を向上させることができる．本節では，4.2 節で述べた手型識別に動作検出の方法を組み合わせて対象とした指文字の認識率を再評価を行った．

動作検出結果 / 動作ノイズを考慮した識別結果	動作あり	動作なし
か	が	か
が	が	か

図 4.6: 動作ノイズを考慮した手型識別結果

表 4.3: 動作ノイズを考慮した手型識別と動作検出の認識結果

クラス	認識率	クラス	認識率	クラス	認識率
あ	0.92	**か**	**0.65**	が	0.98
い	0.91	**き**	**0.96**	ぎ	1.00
う	0.98	く	0.96	ぐ	0.97
え	0.98	け	0.78	げ	0.90
お	1.00	**こ**	**0.86**	**ご**	**0.98**
さ	0.84	**は**	**1.00**	**ぱ**	**0.88**
し	1.00	**ひ**	**0.98**	**ぴ**	**0.89**
す	1.00	**ふ**	**0.77**	**ぷ**	**0.81**
せ	0.77	**へ**	**0.99**	**ぺ**	**0.81**
そ	0.67	**ほ**	**0.72**	**ぽ**	**0.81**
AVG	0.91	や	0.76	**ゃ**	**0.98**
		ゆ	**0.80**	**ゅ**	**0.76**
		よ	**0.97**	**ょ**	**0.83**
		AVG	**0.86**	AVG	0.89

全体の認識率平均：0.90

4.4.6 認識結果と考察

手型識別と動作検出を組み合わせた方法で認識実験を行った結果，表 4.3 のようになった．太字で表したクラスと数値は，認識率が改善されたものを示す．4.3 節の実験結果で特に認識率が低かった指文字に「か」「け」「は」「ふ」「ほ」「や」があるが，認識率が向上していることが確認できる．また，静止指文字と，その手型に対応する動きのある指文字の認識率の大半が向上し，全体の認識率平均が 0.05 改善され 0.90 となった．

しかし,「け」や「ぽ」のように認識率が改善されていない指文字が存在するという課題がある．これは，抽出された手型の矩形領域を 256×256 の大きさの画像に正規化してレンダリングしたことが原因であると考えられる.「け」や「せ」のように手指を立てて表現する指文字の場合，実際に抽出される領域は縦幅の比率が大きい矩形領域になる．この領域を縦横比が一定になるように変換すると手指が潰れてしまい，指の長さや広がりなどの特徴が損なわれてしまうことが原因であると考えられる．

その他にも，SVM の識別関数のパラメータである ω と b を決定するための調整が不足していたことが，認識率の低い原因の 1 つとして挙げられる．最適な ω と b を設定するためには，画像を学習する際のカーネル関数の選択や関数内のパラメータの厳密な調整が必要となる．

動作検出の精度向上により，手領域の動きの認識はほぼ確実に行えることが明らかになった．つまり，従来の手型識別の方法と組み合わせることで精度の向上が期待できることがわかった．手型の誤認識に起因する認識率の低さについては，他の指文字認識の関連研究の方法を活用することで改善することが可能であると考えられる．この方法と動作検出方法を組み合わせれば，全体の認識率の更なる向上が期待できる．

参考として，手領域抽出結果を縦横比を維持したままレンダリングし，その画像から HLAC を計算することで特徴の損失を抑え，特徴量の改善を試みた．同方法を用いてすべての指文字の認識を行った結果について，付録 B で述べる．

4.5　まとめ

本章では，まず 3 章で提案した手法を改良するため，動作ノイズを含めた画像から HLAC を抽出し，静止指文字と手型の動きのある指文字の差別化を行うことで認識率の向上を試みた結果について述べた．その結果，動作検出を用いなくても手型に動きのある指文字を認識可能であることを示し，動作ノイズの特徴を活かした認識が行われていることが確認できた．

次に，動きのある指文字および類似手型の誤認識を防ぐために，最適なパラメータ設定することで精度を向上させた動作検出と手型識別とを組み合わせた手法を提案した．その結果，静止指文字と，手型の動きのある指文字において，互いに誤認識し合う問題が改善され，全体の認識率の平均は 0.05 改善し，0.90 となった．以上のことから，本研究で提案した動作ノイズを考慮した学習データで手型識別を行う方法，ならびに動作検出の方法は，手型の動きのある指文字の認識率を向上させることができることを明らかにした．

3.6 節や 4.3 節の実験結果は，1 名の手型とその動きを識別させた結果である．そのため，動作検出に用いたパラメータは一意に求めることができる．しかし，複数の被験者を対象に指文字の認識を行った場合，実際は図 4.7 のように指文字を表現する際の手型の移動量には個人差があることがわかった．聴覚障がい者が指文字を用いてコミュニケーションを行う場合，その習熟度および使用頻度から，表現速度に大きな差が生じる．生まれつき聴覚に障がいを持っている，または指文字を主なコミュニケーション手段として利用している人は，指文字を表現する速度が非常に速く，読み取るのが難しい．また，指文字を手話の補助的な手段として用いている人，例えば，手話での表現がないもしくは分からない単語を表現する際に用いている人や，覚えたばかりで慣れていない人の場合は，非常にゆっくりとした表現になる．そのため，指文字入力インタフェースへと導入する場合は，複数の被験者の指文字データから動作検出のためのパラメータの最適化を行う必要があると考えられる．

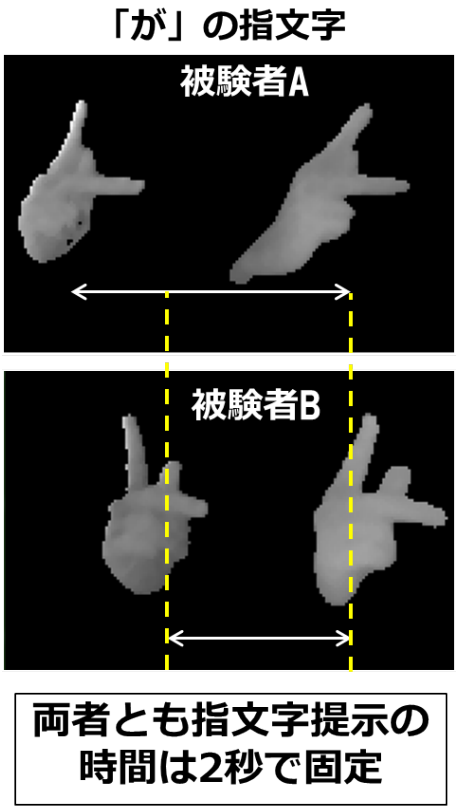

図 4.7: 個人差による手型の移動量の違い

第5章　結論

5.1　まとめ

音声による機器の操作が必要となるような場面において聴覚障がい者は音声入力の利用が困難である．そこで，音声と同様に非接触な入力手段である指文字をその代用とするため，指文字を認識させる必要があると考えた．指文字認識の関連研究では静止指文字の認識精度の向上を試みたものが大半であり，動きのある指文字の認識を試みた例はみられない．また，一般的に用いられるセンサでは手領域の正確な抽出や，動きのある指文字への対応も困難であるという問題点があることが判明した．そこで本研究では，距離画像を用いて手型に動きのある指文字を認識するための方法を提案し，評価を行うことを目的とした．体積による手領域の動的抽出，主成分分析を用いた手領域の動作検出，ならびに動作ノイズを考慮した手型識別を考案し，各方法を実装して認識率の評価を行った．本研究では，撮影速度の速い赤外線 TOF カメラを用いることで，手型の動的な抽出や，動きのある指文字への対応を可能にした．また，抽出された手領域に対する距離ベクトルと k 近傍法による手型識別，手領域の動作検出の方法と組み合わせて，動きのある指文字を認識する方法を提案した．清音の指文字，濁音と半濁音の指文字，および小書き文字で表される指文字について，合計 75 文字を認識させた結果，認識率平均が約 0.85 となった．この結果から，指文字の動きにともない発生する距離画像のノイズが誤認識の原因であることがわかった．

そこで，動きに起因する誤認識を減らすために，この動作ノイズを含んだ画像から HLAC を計算し，SVM で手型識別を行う方法を提案した．静止指文字と手型の動きのある指文字の区別を試みた結果，対象とした指文字の認識率平均は 0.85 となった．このことから，動作検出を用いなくても手型の動きのある指文字を認識可能であることがわかった．

上述の方法と動作検出を組み合わせた方法を提案するため，パラメータ N_f と L_t の組み合わせの調整を行い精度を向上を試みた．その結果，動作検出の精度は 1.0 となった．動作検出と手型識別を組み合わせて認識率を評価したところ，認識率が改善され，0.90 となった．このこ

とから，指文字認識の関連研究の方法と本研究で提案する動作検出を組み合わせることで，認識率の更なる向上が期待できることがわかった．

5.2 今後の課題

指文字入力インタフェースを実現する上で解決するべき課題として，ユーザが入力に用いる状況を想定したときの誤入力を低減するため，1 文字を認識するための精度の向上させることが挙げられる．本研究で対象としていなかった，手指を動かす指文字である「の」「も」「り」「ん」の指文字の認識にも対応させる必要がある．単語レベルでの認識を行う際に，指文字と指文字の間の「わたり」の動作の認識も大きな課題として挙げられる．

動作ノイズを考慮した方法では，全体認識率が 0.90 という結果であった．しかし，この状態で指文字入力インタフェースをユーザが使用した場合は 10 文字中 1 文字で誤認識が起こるため，十分な結果であるとは言えない．その他の識別器として AdaBoost[26] などを用いることで，認識精度の向上がより期待できると思われる．また，今回採用した赤外線 TOF カメラ以外のセンサのデータを使用することで同様の効果が期待できる．例えば，カラーカメラを用いて撮影した輝度画像から撮影し新しく特徴量を計算することで，認識対象の特徴を記述する特徴量の種類を増やす，といったことが挙げられる．

「の」「も」「り」「ん」の手指の動きのある指文字を認識させるには，HLAC を時間軸方向に拡張して動画に適用できるようにした立体高次局所自己相関特徴（CHLAC，Cubic Higher-Order Local Auto Correlation）[25] を用いることで解決できると考えられる．

「わたり」の部分を本研究で提案した方法で解決する場合は，指文字と指文字の間の表現をしている部分の画像を計算機に学習させる必要があり，学習データ数および識別器の計算コストが増大してしまうため，リアルタイムな入力インタフェースとしての実用化は難しい．文字レベルの認識精度を動作検出と組み合わせて向上させることができれば，音声認識と同様に指文字を 1 つの音素として処理することが可能である．自然言語処理の技術をそのまま応用することで解決が期待でき，単語レベルでの認識も可能になると考えられる．

本研究で提案した方法において，3 章，および 4 章で述べた距離画像の撮影から手型認識と動きの認識までを含めたフレームレートは約 25fps であり，パラメータ $N_f = 10$ のときは動きの有無を認識するまでに約 0.33 秒，$N_f = 6$ のときは約 0.20 秒かかった．指文字インタフェースを構築する際には，これらのディレイを考慮した設計についても考えていく必要があるだろう．

謝辞

本研究に関して終始ご指導ご鞭撻を頂きました若月大輔准教授に心より感謝致します．また，貴重な時間を割いて本論文をご精読頂き有用なコメントを頂きました修士論文主査の皆川洋喜教授，副査の西岡知之教授に深く深謝致します．大学院の講義における熱心な指導をしてくださった内藤一郎教授，学生生活における悩みに真摯に対応してくださった河野純大准教授に心から御礼申し上げます．研究を通じて活発な議論をさせて頂きました，同じ指文字をテーマとして研究をしている，同期の瀬戸山浩平さんに感謝致します．また，指文字データの収集に関しては，同じ617研究室で共に研究活動を行っている4年生，院生の皆様のご協力を頂きました．ありがとうございました．最後になりましたが，修士課程に進学する機会を与えてくださり，ありとあらゆる場面で私を温かく見守り続けてくれた両親に深く深く感謝いたします．

参考文献

[1] 黒田和宏，後藤忠敏，生田裕樹，“手形を認識するデータグローブStrinGlove R”，情報処理学会研究報告 Vol.2008，No.3，Vol.CVIM-161，p.343-344，2008.

[2] 福島大志，宮崎文夫，西川敦，“磁気データセットを用いた指文字入力インタフェース「Fingual」の開発”，計測自動制御学会論文集，Vol.48，No.3，pp.159-166, 2012.

[3] 長嶋祐二，藤井昌紀，長嶋秀世，“カラー画像による指文字認識に関する基礎検討”，テレビジョン学会技術報告，vol.17，No.14，pp.19-24，1993.

[4] 舟川政博，平山亮，“指文字画像からの手指形状特徴量抽出方法の検討”，FIT2006（第5回情報科学技術フォーラム），p87-88，2006.

[5] 広瀬健一,“細線化画像を用いた指文字認識”，コンピュータビジョン 84－1，pp1-6，1993.

[6] 慶島淳一，前薗正宜，小野智司，中山茂,“濃淡画像や距離画像を用いた決定木による指文字認識”，システム制御情報学会論文誌, Vol.19, No.4, pp.166-168，2006.

[7] 岩崎聡，朝倉俊行，広瀬謙治,“ニューラルネットワークを用いた指文字認識”，日本機械学会講演論文集，No.2002，Vol.39, pp.239-240, 2002.

[8] 平山亮，舟川政博,“ニューラルネットによる静止画像からの指文字認識”，情報処理学会全国大会講演論文集 第72回平成22年 (2), pp.13-14, 2010.

[9] 新澤真郷，大矢誠，“画像処理による指文字認識”，日本機械学会第46期総会・講演会 講演論文集，Vol.2009，No.46, pp.419-420, 2009.

[10] 渡辺賢，岩井儀雄，八木康史，谷内田正彦,“カラーグローブを用いた指文字の認識”，電子情報通信学会論文誌. D-II，情報･システム，II-情報処理 No.J80-D-2，Vol.10, pp.2713-2722, 1997.

[11] 大里宗之，鈴木基之，伊藤彰則，“カラーグローブを用いた指文字認識における特徴量の統合法”，電子情報通信学会技術研究報告，Vol.105，No.375，pp.73-78，2005.

[12] 糸井清晃，久保田哲也，小林幸雄，”色手袋を用いた指文字認識”，電子情報通信学会 情報・システムソサイエティ大会，p.219, 1997.

[13] 慶島淳一，小野智司，中山茂，“距離画像を用いた決定木による指文字認識”，システム制御情報学会論文誌，Vol.19，No.4，pp.166-168，2006.

[14] 東山和弘，小野智司，王宇，中山茂，“3 次元テンプレートマッチンクによる指文字認識”，電気学会論文誌 C, 電子・情報・システム部門誌，No.125，Vol.9, pp1444-1454, 2005.

[15] 王宇，板井聖治，小野智司，“PCA と 3 次元スキャナによる指文字認識”，情報知識学会誌，Vol.1，No.16，pp.51-60，2004.

[16] Pugeault.N and Bowden.R,“Spelling it Out:Real-Time ASL Fingerspelling Recognition.”, 1st IEEE Workshop on Consumer Depth Cameras for Computer Vision, Jointly with ICCV 2011，pp.1114-1119，2011.

[17] “ OpenNI ”，http://openni.org

[18] S.Ong and S.Ranganath，“ Automatic sign language analysis: A survey and the future beyond.”，IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.27，No.6，pp.873-891，2005.

[19] T.Oggier，B.Buttgen, F.Lustenberger，G.Becker，B.Ruegg and A.Hodac，“ Swissranger SR3000 and first experiences based on miniaturized 3D-ToF cameras ”，In Proc, of the First Range Imaging Research Day at ETH Zurich，2005.

[20] T.Oggier，M.Lehmann，R.Kaufmannn，M.Richter，P.Metzle，G.Lang，F.Lustenbergr and N.Blanc，“ An all-solid-state optical range camera for 3D- real-time imaging with sub-centimeter depth-resolution (SwissRanger)”，Proc. SPIE Vol.5249，pp.634-545. 2003.

[21] B.Buttgen，B.Oggier and T.Lehmann，“CCD/CMOS Lock-in pixel for range imaging:challenges, limitations and state-of-the-art”In proceedings 1st range imaging research day，pp.21-32，2005.

[22] Paredes.R,“Learning weighted metrics to minimize nearest-neighbor classification error”，Pattern Analysis and Machine Intelligence, IEEE Transactions on，Vol.28，No.7，pp.1100-1110，2006.

[23] 小林匠，大津展之，“画像特徴量-高次局所自己相関に着目した画像特徴量と画像認識への応用”，電子情報通信学会誌，vol.94，No.4，pp.335-340，2011.

[24] 栗田哲平，近山隆，“多クラスSupport Vector Machineを用いた一般物体認識での複数候補提示下における分類性能の傾向”，情報処理学会研究報告，
CVIM，vol.2008，no.115，pp.251-258， 2008.

[25] 南里卓也，大津展之，“複数人動画像からの異常動作検出”，情報処理学会論文誌 コンピュータビジョンとイメージメディア 45(SIG 15)，pp.43-50，2005.

[26] 三田雄志,“AdaBoostの基本原理と顔検出への応用：CVIM研究会 チュートリアルシリーズ（チュートリアル2）”，情報処理学会研究報告．CVIM，vol.2007，no.42，pp.265-272，2007.

本研究に関する成果・発表等

[A1] 三宅太一，若月大輔，内藤一郎，“距離画像を用いた動きのある指文字認識の検討-濁音・半濁音・拗音の認識”，電子情報通信学会　2012 年総合大会 情報・システムソサイエティ特別企画学生ポスターセッション予稿集，p.129，2012.

[A2] 三宅太一，若月大輔，内藤一郎，“距離画像を用いた動きをともなう指文字認識に関する基礎的検討 ”，筑波技術大学テクノレポート，vol.20，No.1，pp.7-13，2012.

[A3] 三宅太一，若月大輔，内藤一郎，“距離画像を用いた動きのある指文字の非接触認識 ～指文字入力インタフェースの実現を目指して～ ”，第 8 回日本聴覚障害学生高等教育支援シンポジウム・ランチセッション「聴覚障害学生支援に関する機器展示」，p50，2012.

[A4] 三宅太一，若月大輔，内藤一郎，“距離画像を用いた動きのある指文字の非接触認識手法の検討 ”，電子情報通信学会 HCG シンポジウム 2012，pp.270-275，2012.

著者のその他の研究成果

[B1] 若月大輔，内藤一郎，三宅太一，元西洋平，“ マイクロプロジェクタを用いた聴覚障害者のための学習支援システムに関する基礎的検討 ”，電子情報通信学会技術研究報告. WIT，Vol.111，No.58，pp.19-24，2011.

[B2] 若月大輔，内藤一郎，三宅太一，元西洋平，“聴覚障害者の講義受講支援のためのプロジェクタを用いた情報保障の基礎的検討 ”，筑波技術大学テクノレポート，Vol.19，No.2，pp.1-6，2012.

[B3] 若月大輔，内藤一郎，三宅太一，河野純大，加藤伸子，塩野目剛亮，西岡知之，皆川洋喜，村上裕史，三好茂樹，元西洋平，“ 卓上投影した文字通訳による聴覚障害者の講義受講支援の基礎的検討 ”，電子情報通信学会技術研究報告：信学技報，Vo.111，No.472，p[p.39-44，2012.

付録

A　特徴量の改良とその結果

A.1　主成分分析による特徴量の次元圧縮

3 章で使用した特徴量は，距離画像を距離値をグレースケールの輝度値に変換し，16 × 16 の大きさにスケーリングした画像の各画素をそのまま用いた 256 次元ベクトルである．しかし，あまり特徴を記述していない成分が含まれており，冗長であった可能性がある．そこで，主成分分析を行うことで最適化した特徴ベクトルについて追加実験を行い，その認識率について調査を行った．元の 256 次元の特徴ベクトルに対し主成分分析を用いて次元圧縮し，20 次元にして認識に用いた．

3.6 節の実験結果のうち，特に認識率の低かった指文字を中心に全く同じ条件で認識率を求めた．主成分分析で次元圧縮した特徴量を用いて認識を行った結果を表 A.1 に示す．

表 A.1: 特徴量改良後の認識結果

クラス	次元圧縮前 (256 次元)	次元圧縮後（20 次元）
い	0.40	0.80
す	0.60	0.75
そ	0.00	0.50
ぬ	0.20	0.00
ひ	0.35	0.70
ま	0.40	1.00
ら	0.65	0.70
認識率平均	0.37	0.64

A.2 考察と課題

特徴ベクトルの次元削減を行う前では，認識率が低かった静止指文字の平均認識率は 0.37 であった．次元削減後では 0.64 となり，認識率の向上が確認できた．しかし，次元削減を行った後でも「ぬ」の指文字ついては認識率の改善は見られなかった．追加実験の結果から，次元削減によって全体の平均認識率の向上が見られたが，一部のクラスの認識率は改善されないことがわかった．

3 章で提案する方法では，特徴量を取得する際に抽出した手領域のスケーリングを行っている.「ち」,「い」,「ぬ」,「ろ」といった手指を立てて表現する指文字の場合は，スケーリングを行う前は縦方向の比率が大きい画像が抽出される．これらの画像を縦横比が 1 : 1 になるようにスケーリングした場合，立てた手指が潰れてしまい，手型の特徴をよく表している部分の情報が損なわれてしまう．特に,「ぬ」と「ろ」の場合はこの情報の損失が顕著に表れ，TOF カメラから手型までの距離，奥行き情報や手指の屈伸の形といった情報が損なわれてしまったため，どちらの手型も近い形であると判別されてしまったと考える．

主成分分析による方法は，強く特徴を表している要素を特徴量として選択し認識に用いる．類似する手型のある指文字や特徴の損失がある指文字であっても，表 A.1 に示すように，誤認識を低減できる効果があることが示された．しかし，対象としたすべての指文字の認識率が改善されておらず，この方法で認識精度の向上させるには限界があることがわかる．特徴量の損失の少ない特徴量や，手型の識別能力に優れた特徴量を選択して認識に用いる必要があることが，今後の課題として挙げられる．

B 動作ノイズを考慮した方法によるすべての指文字を対象とした手型識別

B.1 概要

4 章で使用した特徴量は，抽出された手領域を 256×256 にスケーリングを行った後に計算した HLAC である．動作検出と組み合わせた方法で認識実験を行った結果，認識率が改善されていない指文字が生じる原因として，スケーリングを行ったことによる特徴の損失が原因であると仮定した．そこで，縦横比を維持したまま HLAC を計算することで，認識率の改善を試みた．

B.2 実験環境と条件

撮影時の環境は 4.3 節と同様の状態にした．認識対象となる指文字は，清音の静止指文字 41 文字に加え,「の」「も」「り」「ん」といった手指を動かす指文字の 4 文字と，濁音，半濁音，小書き文字，および「を」の手型の動きのある指文字 35 文字の 80 文字を対象として認識実験を行った．被験者 1 名について，すべての指文字を連続で提示して撮影したものを 1 セットとし，合計 10 セット撮影した．動作ノイズを考慮した HLAC を，クラスごとに連続 20 フレーム分計算し辞書に登録した．撮影したセットの内，2,4,6,8,10 番目のセットを学習に用い，1,3,5,7,9 番目のセットを認識率評価に用いた．

B.3 実験結果と考察

認識実験を行った結果について，表 B.1 に示す．指文字全体の認識率平均は約 0.28 となった．表 B.1 の結果のうち，静止指文字と，その指文字に対応する手型の動きのある指文字は，お互いに認識し合うことが確認できた．

手型識別に加え，動作検出を組み合わせて改めて認識率を評価した．その結果，認識結果は 0.1 改善されて全体で約 0.38 となり，動作検出との組み合わせには認識率向上の効果があることが確認できる．しかし，全体的に認識率が低い結果となり，スケーリングを行わなかったことによる効果が得られなかった．

認識率が低くなってしまった原因として，HLAC では 80 文字すべてのクラスを分類できるような特徴が表現できていないことが考えられる．クラス数を減らし,「あ」行と「か」行の 10 クラスに限定した上で同様に認識させた結果，認識率の平均は約 0.80 となった．1 つのクラスを表現するための特徴量を増やし，より多次元の特徴ベクトルを用いることで解決できると考えられる．その他には，スケーリングを行わなかったことによる，手領域の大きさの変化が影響しているものと思われる．手型を提示する位置がカメラから近ければ手領域は大きくなり，遠ければ小さくなるため，同じ手型でも得られる特徴量が変化する．HLAC で大きさの変化に対応するためには，3×3 のマスクパターンだけでなく，5×5，7×7 のようなパターンの計算を行い，すべての結果を統合した 105 次元のベクトルを用いる，といった方法が考えられる．

表 B.1: すべての指文字の認識率

クラス	認識率	クラス	認識率	クラス	認識率	クラス	認識率
あ	0.72	ぁ	0.49	や	0.35	ゃ	0.35
い	0.12	ぃ	0.33	ゆ	0.21	ゅ	0.69
う	0.20	ぅ	0.43	よ	0.37	ょ	0.66
え	0.00	ぇ	0.54	-	-	っ	0.44
お	0.16	ぉ	0.24	-	-	-	-
か	0.05	が	0.28	な	0.11	わ	0.22
き	0.16	ぎ	0.23	に	0.00	を	0.40
く	0.14	ぐ	0.26	ぬ	0.00	ん	0.49
け	0.00	げ	0.33	ね	0.52	該当無	0.80
こ	0.38	ご	0.15	の	0.25	AVERAGE	
さ	0.00	ざ	0.09	ま	0.24	0.28	
し	0.16	じ	0.47	み	0.40		
す	0.36	ず	0.28	む	0.12		
せ	0.00	ぜ	0.35	め	0.32		
そ	0.25	ぞ	0.37	も	0.19		
た	0.12	だ	0.40	ら	0.05		
ち	0.11	ぢ	0.33	り	0.36		
つ	0.11	づ	0.34	る	0.33		
て	0.06	で	0.53	れ	0.20		
と	0.12	ど	0.28	ろ	0.07		
は	0.59	ば	0.19	ぱ	0.34		
ひ	0.30	び	0.28	ぴ	0.33		
ふ	0.65	ぶ	0.16	ぷ	0.13		
へ	0.37	べ	0.30	ぺ	0.17		
ほ	0.00	ぼ	0.14	ぽ	0.21		