

公立はこだて未来大学 2017 年度 システム情報科学実習
グループ報告書

Future University Hakodate 2017 System Information Science Practice
Group Report

プロジェクト名

AI するディープラーニング

Project Name

AI Love Deep Learning

グループ名

手話翻訳グループ

Group Name

Sign Language Translation

プロジェクト番号/**Project No.**

13-A

プロジェクトリーダー/**Project Leader**

1015031 黒川大智 Daichi Kurokawa

グループリーダ/**Group Leader**

1015086 好本大地 Daichi Yoshimoto

グループメンバ/**Group Member**

1015242 岩田聖 Sho Iwata

1015007 佐野将真 Shoma Sano

1015129 東浦聡太 Sota Higashiura

1015019 正木大雅 Taiga Masaki

1015085 山本大智 Daichi Yamamoto

指導教員

竹之内高志 寺沢憲吾 香取勇一 片桐恭弘

Advisor

Takashi Takenouchi Kengo Terasawa Yuichi Katori Yasuhiro Katagiri

提出日

2018 年 1 月 19 日

Date of Submission

January 19, 2018

概要

近年、ロボットやボードゲームなど様々な例で人間を模倣出来る人工知能が注目を集めている。人工知能とは機械学習等を用いて、普段人間が行っている学習能力と同様の知能をコンピュータで実現しようとする技術のことである。その中でもディープラーニング(深層学習)はさまざまな分野に応用され、特に注目されている。ディープラーニングとは、脳のネットワークモデルを応用した機械学習の手法で、脳の神経細胞にあたるノードを多数に組み合わせることによって従来よりも複雑な学習が可能となっている。特に画像処理、音声処理、自然言語処理などに応用されている。しかし、ディープラーニングの実用化はごく最近で、活用範囲はいまだ未知数である。そこで、本プロジェクトではディープラーニングの新しい応用方法を提案しシステムとして実用化することを目標に活動を行った。最初にプロジェクトメンバー間でディスカッションを行い、何を実用化するかを目標の設定を行った。

ディスカッションの結果、本プロジェクトでは2つの目標を設定し、メンバーを分けた。グループAでは手話翻訳システムの開発、グループBでは詰将棋自動生成システムの開発を目標とした。本報告書ではグループAの活動内容について述べる。まず手話翻訳システムをテーマに選択した理由としては、手話をつかったことのない健常者と手話を使うろう者の間にはコミュニケーションの障壁が出来ていると考えたからである。理由としては手話の学習コストが高いこと、健常者の習得者人口が少ないこと、手話を習得していない人には意味が伝わらないことなどが挙げられる。そこで本グループでは、この課題を解決するためにディープラーニングで手話動画を学習させ、それを翻訳できるシステムの開発に取り掛かった。しかしこのシステムの開発にはいくつかの解決しなければならない問題点があった。その問題の1つが画像や映像などから学習に必要な特徴のみをどのように抽出するかである。今年度私たちはこの問題点を重視し、手話の1種である指文字の認識に取り組んだ。学習の手法としては、主に画像認識に応用される畳み込みニューラルネットワーク、(以下CNNとする)を採用した。また、制作過程でグループ分けを行い、

- 収集班 学習に用いるデータセットの作成および関連プログラムの作成
- 実装班 Pythonを用いた機械学習プログラムの作成

の2グループに分かれ作業を行った。

前期はディープラーニングに関しての学習および手話認識の前準備としてグー、チョキ、パーのハンドジェスチャの認識を行った。収集班はグループメンバー全員から手のみの画像を収集し、プログラムを用いてデータ拡張を行った。実装班は教科書やwebサイトからディープラーニングを用いたサンプルプログラムを調べ、それを活用しながらPythonを用いてハンドジェスチャ分類を行えるプログラムを作成した。

後期では、残りの活動時間などからどんなシステムを開発するのかを細かく考え、結果として指文字の認識を目標に活動した。収集班はプロジェクトメンバーからあ～た行までの指文字を行っている画像を収集し、前期同様データ拡張作業および新しく肌色抽出作業を行った。実装班では指文字の認識、分類が可能なプログラムの提案、作成を行った。また、最終発表に向けたスライド、ポスター等の発表資料の作成も並行して行った。

キーワード Deep Learning, 機械学習, 画像認識, 手話, 翻訳, 指文字

(※文責：正木大雅)

Abstract

In recent year, artificial intelligence ability of imitating human beings attracts. For example, robots, board game, and so on. Artificial intelligence is a technique to realize intelligence similarity to learning ability normally done by humans by computer using machine learning. Among them, deep learning has been applied to various fields, and it has received particular attention. Deep learning is a machine learning method applying a network model of the brain, which makes it possible to learn more complicated than before by combining a number of nodes in the neurons of the brain. It is generally applied to image processing, speech processing and natural language processing. But, the practical use of deep leaning is very recent and utilization range is still unknown. Therefore, in this project we proposed a new application method of deep learning and worked on the goal of practical application as a system. First we discussed with project members and set goals as to what to put into practical use.

As a result of the discussion, this project set two goals and divided the members. Group A targets development of sign language translation system. Group B targets development of tsume-shogi automatic generation system. This report describes the activities of Group A. First of all, we chose the sign language translation system as the theme because we thought that communication barriers were created between healthy people and deaf people. There are three reason. First, high learning cost of sign language. Second, there are few population who acquire sign language. Third, the meaning does not convey to those who have not learned sign language. Therefore, in order to solve this problem, we began developing a system that lets sign language images learn by deep learning and can translate them. However, there were some problems to be solved in the development of this system. One of the problems is how to extract only the features necessary for learning. This fiscal year we focused on this problem and worked on recognition of finger character. As a learning method, a convolution neural network (CNN) was adopted. Additionally, we divide us into two groups such as,

- Group of collecting: Taking video of sign language movies and division into images, padding, skin color extraction work and creation of related programs
- Group of implementation: Making program that implements machine learning using Python.

In the first semester, we learned about deep learning and recognized the hand gesture as preparation for sign language recognition. Group of collecting gathered images of only hands from all group members and augmented them using originally developed program. Group of implementation examined sample programs using deep learning from textbooks and web sites, and created a program that allows hand gesture classification using Python while utilizing it.

In the second semester, we reconsidered what kind of system to develop with the restriction of the remaining activity time etc. Group of collecting gathered images that doing finger character from project members, and did data augmentation and skin color extraction work. Group of implementation proposed and created a program that can recognize and classify finger character. In addition, we made preparations for presentations such as slides and posters for the final presentation in parallel.

Keyword DeepLearning, machine learning, image processing, sign language, translate, finger character

(※文責：正木大雅)

目次

第 1 章 はじめに

1.1 背景

手話は手の位置, 手の形, 手の動きなどを組み合わせて意味を伝える表意記号である。これを母語とし, コミュニケーション手段として用いる人をろう者と呼ぶ。ろう者は先天的, もしくは幼いころに失聴した人と日本語を母語として獲得した後に失聴した人とで母語として用いる手話が異なる。前者が用いる手話は日本語の音声を手の動作や顔の表情に変換しているものではなく, 独自の語彙と文法構造を持つ異なる言語であり, 日本手話と呼ぶ。後者が用いる手話は日本語手話の単語を日本語文法の形に再配置したもので日本語対应手話と呼ぶ。それに加えて, 名前や地名などの手話単語で表せない単語を表現するための仮名文字の各文字に対応した指文字がある。手話は日本手話, 日本語対应手話に指文字を加えた言語の総称となっている。基本的にろう者は日本語手話と日本語対应手話の両方を学び, ろう者同士での会話には日本手話を用い, 聴者との会話では日本語対应手話に切り替える。そのため, ろう者は 2 つの言語を日常的に使用する 2 言語使用者である。手話にはいくつかの課題点がある。それらの課題点としては手話の学習に費用が掛かるという点, 手話を習得していない人に対して意味が伝えられない点などが挙げられる。まず初めに手話の学習費用について, 聴者が手話を学ぶ際は手話サークルや手話講習会を利用する。しかし, 手話サークル, 手話講習会では主に日本語対应手話のみを学ぶ。そのため, 日本手話を学べる場所は少なく, 金銭的な費用が掛かる場合が多いので, 習得難度が高い。次に手話を習得していない人に対して意味が伝えられない点について, 手話は日本語から派生した言語ではないので推測で意味を翻訳することは難しい。これにより, 手話でしか話せない人と日本語でしか話せない人との間に, コミュニケーションの障壁が出来てしまっているという問題がある。そこで私たちはディープラーニングや機械学習を用いた先行研究の手法を組み合わせることで, 手話翻訳に応用できると考えた。

(※文責: 山本大智)

1.2 目的

手話を使ったことのない健常者と手話を使うろう者同士がもっと気軽にコミュニケーションを取れるようなシステムを開発する。そのために前期では特殊機器を用いずに日本語手話, 日本語対应手話, 指文字のそれぞれに対して映像からの翻訳を可能にすることを目標に設定し, 活動を行った。しかし, 期間内に日本手話と日本語対应手話に対応させるために不可欠な時系列のデータから特徴量を抽出する手法を確立することが難しいと判断した。よって, 後期では指文字のみの翻訳に絞り, 50 音に翻訳システムを対応させることを最終目標として活動を行った。

(※文責: 山本大智)

1.3 従来例

機械での手話翻訳の従来例がいくつか挙げられる。1つは図 1.1 に示す LeapMotionController[1] (以下 LMC) からの座標を取得して手の形を推定する手法である [2]。この手法では LMC によって取得した各座標間のユークリッド距離を利用して手の形を推定している。2つ目に図 1.2 に示す SignAloud[3] と呼ぶ手袋型のデバイスが存在する。デバイスに内蔵されたセンサから取得した情報を計算用のコンピュータに送信し、翻訳した文章をスピーカーを用いて音声として出力するシステムとなっている。その他に、図 1.3 に示す Kinect for Windows[4] (以下 KfW) の骨格追跡機能を用いた手法が挙げられる [5]。この手法では KfW によって肘を基点とした手の相対座標を時系列に取得し、取得した時系列データを確率モデルの 1 つである隠れマルコフモデルを利用するツールである Hidden Markov Model Toolkit[6] を用いて学習する。これらの従来例に共通する問題点としてカメラ以外の特殊機器が必要になること、学習データの作成に多くの時間がかかることなどが挙げられる。

(※文責：山本大智)



图 1.1 LeapMotionController[2]



图 1.2 SignAloud[3]



图 1.3 Kinect for Windows[4]

1.4 手話翻訳実装の課題点

手話翻訳を実装するにあたり、以下の点が課題となることが予想された。

1. 顔や手の特徴が手話の意味識別に必要となること。単語によっては手の動きに加えて、顔の表情が意味に影響を与えるものも存在する。それに加えて、顔と手が重なった場合に各領域が正確に判断できないと認識の精度に影響を与える。
2. 手話映像には無駄な情報が多量に含まれること。背景や服装が認識に影響を与えないようにする。また、肌色の領域が服装によって変化する部分がある。この部分についても、差異が出ないような仕組みが必要になる。
3. 単語の連続した手話動作をどのような基準で一つの単語に分割するのか機械で判断させることが困難であること。一連の手の動作の組み合わせによって成り立つ単語もあるために単語の区切りが間違った状態でも正しいと認識する場合がある。

(※文責：山本大智)

1.5 課題点への対応策

1.4 で記載した現在の問題点を解決するため私たちはブレインストーミングを行い、意見交換を行うことで具体策を考えた。

1 の問題の解決策としては OpenCV によって顔の表情を認識させるシステムを別に実装し、それぞれの認識結果の組み合わせによって判断する方法を考えた。それに加えて、顔と手に関する衝突判定を実装することを考えた。

2 の問題の解決策としては肌色検出によって肌色以外の部分を削除し、顔や手の特徴を容易に取得できるデータを学習に用いることで認識精度を上げられると考えた。加えて、顔と手の領域を抽出して、それぞれ専用の学習データを作る手法も効果があると考えた。

3 の問題の解決策としては手の動きを抽出できるアルゴリズムを応用することによって、動画から直接手話を認識させる手法を考えた。また、単語を区切る際に文法として誤りがあるかを判断するシステムが必要となる。

(※文責：山本大智)

第 2 章 プロジェクトの概要

2.1 課題設定

手話をディープラーニングで学習させるためには解決しなければならない課題がいくつかあった。その 1 つは手話の映像には無駄な情報が多量に含まれている点である。無駄な情報とは例えば手話ニュースの場合、対象が着ている服や映像中の背景の情報などである。この情報は手話に直接関係なく不必要であるため、ディープラーニングの認識を行うためには除去することが望ましい。よってこの課題点は言い換えれば図 2.1 のような学習に必要な特徴のみをどのように抽出するのかである。そこで本年度の AI するディープラーニングのプロジェクト学習ではこの課題を解決することとした。また調査した先行事例では特殊機器を用いた事例が多かった。しかし、ろう者の方々が手軽に使うには、特殊機器を必要としないシステムの方が好ましい。よって今回は、web カメラから手の特徴を抽出し、翻訳を行うこととした。

以下に具体的な課題を記載する。まず、グループ全体においての課題を示す。

- ゼロから作る DeepLearning[7] を使用して、ディープラーニングに関する基本学習を行い、理解を深める
- 機械学習を用いて、特殊機器を必要とせず、web カメラ等の一般的な機器のみで使用可能な手話翻訳システムを作成する
- web カメラの画像から無駄な情報を除去し、手領域と顔領域のみを抽出するアイデアを考える

次に、プロジェクト円滑に進めるために、学習データおよびテストデータ等のデータセットを作成する収集班と手話認識を行うためのプログラムを作成する実装班の 2 つの班に分け、班ごとに課題を設定した。それぞれの班のメンバーは以下の通りである。

- 前期
 - － 収集班 東浦, 正木
 - － 実装班 岩田, 佐野, 山本, 好本
- 後期
 - － 収集班 東浦, 正木, 山本
 - － 実装班 岩田, 佐野, 好本

収集班においては、以下の課題を設定した。

- ディープラーニングの学習に使用する画像のデータセットを作成する
- 効率的および適当なデータ拡張方法の模索およびそのプログラムを作成する
- 無駄な情報を除去するアイデアの 1 つである肌色抽出を行うための方法の模索および実行するためのプログラムを作成する
- 画像データの観点から、効率的なディープラーニングの学習方法を検討する

次に、実装班において以下の課題を設定した

- ディープラーニングを行うパソコンに開発環境を構築する
- Chainer による実践深層学習 [8] を使用して、ニューラルネットワークを実装するためのライブラリである Chainer を学習する
- web サイト [9] 上に掲載されているディープラーニングのプログラムを参考に、簡易的な機械学習を用いたプログラムを作成することで、完成形のイメージをつかみ、機械学習に対する知識や理解を深める
- 機械学習の手法を検討する
- 作成したデータセットからディープラーニングを行うことで、手話の判別が可能な学習済みモデルを作成する
- 学習済みモデルに訓練データ、テストデータをそれぞれ適用させ認識精度を調査する

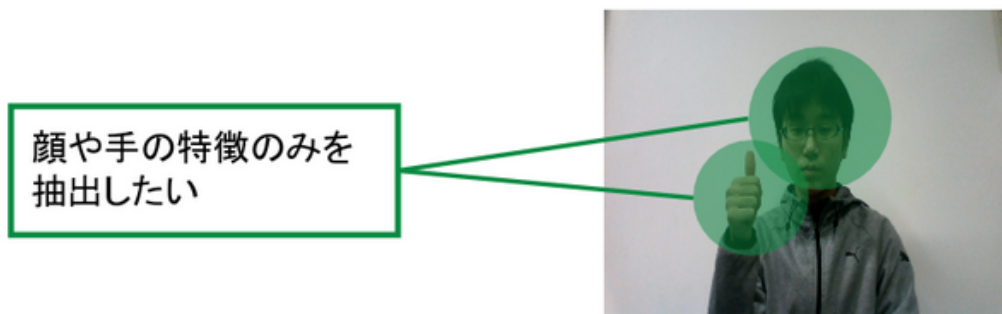


図 2.1 必要な特徴のみ抽出する例

(※文責：正木大雅)

2.2 前期における到達目標

前期のプロジェクト活動時点では最終到達目標として広く知られている手話ニュースを翻訳するシステムの開発を掲げた。よってその前段階として無駄な情報がなく、分かり易い画像から特徴量を取得し、ディープラーニングすることを到達目標に設定し活動した。目的としては実際にディープラーニングを用いて画像認識するプログラムを作成することで知識や理解を深め、到達可能な目標を明確にすることや後期のプロジェクト活動へ生かすことなどが挙げられる。方法としてはグー、チョキ、パーのハンドジェスチャを分類できるプログラムの作成とした。それぞれの班の目標としては実装班が web サイト上に記載されているディープラーニングを用いた画像認識を行うプログラムのソースコードを参考に、3 種類の手形状認識が可能なプログラムの作成とした。収集班はグループメンバーから手の撮影を行い、データ拡張、厳選等を行うことでディープラーニングに用いるデータセットの作成とした。

(※文責：正木大雅)

2.3 後期における到達目標

後期のプロジェクト活動では、残された期限やグループメンバーの技術力等を考慮し、初期の到達目標とは異なる到達目標を新たに設定した。新たな目標としては畳み込みニューラルネットワーク (以下 CNN とする) を用いた指文字認識とした。指文字は手話の 1 種であり、固有名詞を表すときなどに用いる。班ごとの活動目標を以下に記載する。実装班は web サイト上に記載されているプログラムのソースコードを参考に、指文字を認識し判別できるプログラムの作成を目標とした。その際ディープラーニングやその他の機械学習の手法から何を用いるのかの吟味や収集班から受け取ったデータからどのデータを使用するかの厳選、訓練データ数とテストデータ数の決定、プログラムに対応させるための編集作業を行うことも同時に目標とした。収集班はメンバーから指文字を行っている動画を撮影し、サイズ変更、画像への分割、データ拡張、肌色抽出、厳選を行うことでディープラーニングする際のデータセットを作成することや効率的に学習する方法の模索を目標とした。

(※文責：正木大雅)

第 3 章 課題解決のプロセス

3.1 技術・知識取得

3.1.1 Python の学習

Python には画像処理やニューラルネットワークを実装するライブラリがそろっていることや、ディープラーニングに関する基礎学習に採用していたゼロから作る Deep Learning[7] では、Python をプログラミング言語として用いていることから、本グループでは Python をプログラミング言語として用いプログラムの実装を行うことに決定した。しかし、本グループメンバーの Python に対する知識が無かったため、事前学習として基本的な Python の学習を行うことにした。プロジェクト活動の時間外に各個人で学習を行い翌週からはディープラーニングの学習を行える状態にした。オンラインの学習サイトの Progate[10] を Python の基礎学習に利用した。

(※文責：東浦聡太)

3.1.2 ディープラーニングの学習

Python の基礎学習を終えた後にディープラーニングについての学習を行った。プロジェクト活動時間にその日の学習目標を決め、個人で学習を行い決められた部分の学習が完了してからグループメンバー内での理解度の確認を行った。ネットワークの構造やデータの学習の部分など、個人の学習だけでは完全に理解ができなかった部分はメンバー同士で共有し解決できるようにした。メンバー内で解決できない場合はプロジェクト教員に積極的に質問するようにした。この学習と先行研究についての情報収集などから、本グループの目標である手話翻訳については、CNN（畳み込みニューラルネットワーク）を基に実装を行うことも決まった。ディープラーニングの基礎学習を終えてから、Python でのプログラムの実装を行う際にニューラルネットワークのライブラリである Chainer、画像処理のライブラリである OpenCV を用いることが決まり、Chainer による実践深層学習 [8] と OpenCV による画像処理入門 [11] を使用しプログラムの実装に活用した。

(※文責：東浦聡太)

3.1.3 手話、指文字についての調査

手話に対する基礎知識を身につけるために、手話についての学習を行った。手話は日本手話と日本語対応手話に大きく二つに分かれており、日本手話は日本語と比べ、文法も異なっている。それに対して日本語対応手話は日本語と文法を統一させた言語である。また、手話は音声言語や書記言語より語彙の数が少ないことから、指文字と呼ばれる手の形を書記言語の文字に対応させた視覚言語が存在しており、この指文字を用い手話単語にない単語を表現する時などに用いられている。翻訳において日本手話では文法を日本語対応へと変換するための自然言語処理が必要となる。今年度のプロジェクト活動では始めに指文字についての翻訳を目標にすることになった。この目標が実現した時点で期間が空いていた場合に、最終目標である手話翻訳に向けて目標の転換をできるように

した。

(※文責：東浦聡太)

3.2 中間発表

3.2.1 中間発表に関する活動

2017年7月14日、公立はこだて未来大学にてプロジェクト活動の中間発表が行われた。中間発表に向けてポスターの製作や発表用のプレゼンテーション用のスライドの製作を行った。グループ内でポスター班とスライド班に分かれて中間発表に向けた活動を行った。ポスター班は Adobe Illustrator を用いて作成した。スライド班は前期のグループ活動や進捗状況、後期に向けて目標などについてのスライドを作成した。



図 3.1 中間発表 手話グループポスター

(※文責：東浦聡太)

3.2.2 中間発表における評価シートの内容と考察

中間発表で受けた評価は、以下の表 3.1 に示す結果となった。発表技術、発表内容ともに平均が 7 を上回っていた。中間発表に向けた活動期間を多くとり、より良い発表を行うために努力した結果このような評価を受けることができた。

表 3.1 中間発表での評価

評価	発表技術	発表内容
10	2	1
9	0	1
8	7	5
7	2	5
6	4	3
5	1	1
4	0	0
3	0	0
2	0	0
1	0	0
無回答	0	0
平均点	7.6	7.2

中間発表では多数の意見や質問が寄せられた。まず、発表内容に関しての意見、質問を以下に示す。

- スライドが大変わかりやすかったです。
- スライドを読み上げているだけ。
- 声が大きくてよい。
- 声が聞こえやすかった。
- 目標が決まっただけなので、これから問題解決などが楽しみです。
- 展望が伝わりにくかった。
- 声がハッキリと聞き取りやすい。
- 発表中落ち着きがなかった。
- 作ったポスターをもっと活用すれば良いと思った。
- 文字が多い印象だったので文字を少し減らしては。
- 発表の段落分かりやすかったが、具体的な手法の効果が自分には伝わらなかったスライドがまぶしかった。

次に発表内容に関しての意見、質問を以下に示す。

- どこまで実現できるか楽しみにしています。
- 先行研究について触れていたりしてよい。
- 目標はよかったが計画が曖昧だと思った。
- 手順が分かりやすく内容が入りやすかった。
- スライドの色合いや見た目がシンプルで見やすい。
- 手話の今後が楽しみだった。
- 概要が先行研究について詳しく説明されていた。
- 今後の課題についての記述が少し薄かった。
- 目標設定という見方だと素人には難しい。
- 完成したらおもしろいと感じた。
- 趣旨がよく整理されており分かりやすい。
- 質疑応答について答えがピントはずれで質問に答えてないことが、複数見受けられた。

発表技術については、声の大きさや聞こえやすさに対して良い評価が多かった。発表練習の時間を多くとったことでメンバー全員の発表技術が上がったと考えられる。発表用のスライドやポスターに関しては、細かい部分ではあるがいくつか指摘があったので、次の最終発表に活かせるようにした。発表内容に関しては先行研究に触れている点、今後の目標や課題に関しての説明に関して良い評価だった。しかし、質疑応答の時間に関してはより丁寧な説明を行ったり、想定される説明を事前に考えるべきだった。

(※文責：東浦聡太)

3.3 開発

3.3.1 手話画像の収集

手話翻訳の前段階として、まず静止画のグー、チョキ、パーの画像を判定できるプログラムを作成することとなりそれぞれのデータは、各グループメンバーが自身の web カメラを用いることにより作成した。その後、それぞれの画像にコントラスト調整、ガンマ変換、平滑化、ノイズ付加などの加工を加えることによりデータの拡張を行いデータセットとした。次に最初の目標である指文字の認識に用いるデータセットの作成を行った。この作業はデータ班が行った。パソコンの web カメラから指文字を示している状態のメンバーの動画を撮影し、撮影した動画データを動画編集ソフトを用いてフレーム毎の画像に分割処理を行った。作成した画像をグー、チョキ、パーの画像と同様に加工処理を行いデータのデータ拡張を行った。次に作成した画像から指文字を示している部分と対象の顔のみを抽出する方法について検討した。情報収集を行った結果、肌色抽出処理を用いることで、作成した画像から対象が着ている服装や背景の部分を取り除くことができた。しかしこの処理の場合、対象の服装や背景が肌色に似ている場合に、指文字と顔に加え無駄な部分が抽出されてしまうためデータセットに使うことができなかった。よりデータセットの種類を増やすために、Python で書かれたオープンソースニューラルネットワークライブラリである Keras の `keras.ImageDataGenerator` クラスを用いてデータのデータ拡張を行った。この処理では対象をランダムに変形、加工することができより豊富なデータセットを作成することができた。最終的には、指文字の分割画像を Keras を用いてデータ拡張処理を行い、その画像に肌色抽出処理を行った画像が学習に使われるデータセットになった。次の目標のためにいくつかの語彙を表している手話

の画像も作成したが、結果的には学習に使われることは無かったが、指文字の学習において、不正解データであるその他画像として利用することができた。

(※文責：東浦聡太)

3.3.2 プログラムの開発

プログラムの開発ではデータ収集をする班と実装をする班でメンバーを分けて開発を行った。データ収集班ではディープラーニングで学習させるための指文字画像を拡張するプログラムと指文字画像の肌色部分のみを抽出した画像を生成するプログラムの2つのプログラムを開発した。開発環境はどちらも同じでOSはWindows10、言語はPycharmでPython3.4を用いて開発を行った。指文字画像を拡張するプログラムでは、主にKerasというPythonのライブラリを用いて開発した。このプログラムではある1枚の画像にぼかしやずらしを入れ、別の画像として保存するプログラムである。今回は1枚の画像につき約20枚ずつのデータ拡張画像を生成した。指文字画像の肌色部分のみを抽出した画像を生成するプログラムでは、主にOpenCVというライブラリを用いて開発した。このプログラムではデータ拡張した指文字画像を読み込み、その読み込んだ画像を指定した一定のRGB値のみの部分を残し、新しい画像として画像を生成した。図3.2が肌色抽出する前の画像で図3.3が肌色抽出した後の画像である。

実装班ではディープラーニングを用いた指文字を認識するプログラムを開発した。開発環境はOSをUbuntu14.04、プログラミング言語はPycharmでPython3.4を用いた。この開発では主にChainer, OpenCVというPythonのライブラリを用いた。また、ディープラーニングの手法として畳み込みニューラルネットワークを用いた。今回は入力層と4層の中間層と出力層の計6層のネットワークを構築したこのプログラムはデータ収集班が集めたデータを読み込み、多値分類問題として学習し訓練とテストを行い、その訓練とテストの正答率を結果として出力し、学習モデルの保存を行うプログラムである。今回の多値分類では6つの値を分類した、内訳としては50音1行とその他の計6値での分類(例えば「あ」、「い」、「う」、「え」、「お」、「その他」で6値)であった。しかし、これだけでは訓練とテストの正答率しかわからないため、学習済みモデルを用いて何か画像を読み込ませると6つのうちのどれなのかを出力させるためのプログラムを新たに開発した。

(※文責：佐野将真)



図 3.2 抽出前の画像

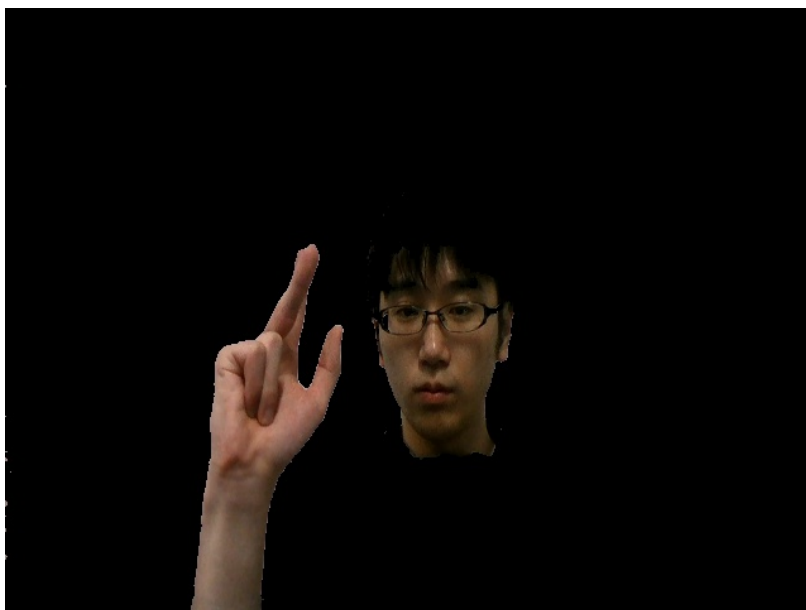


図 3.3 抽出後の画像

3.4 課題と到達目標の修正

3.4.1 手話の認識における困難さ

手話を認識するにはまず手領域を認識させなければならないこと、手は形に変化がありその変化を含めると認識が困難になる、次に単一のカメラのみでは奥行きを判断しづらいため手話単語の連続した動作をどのような基準で1つの単語として分割するのか機械に判断させるのは困難である。

(※文責：佐野将真)

3.4.2 本年度解決すべき課題と到達目標の修正

今期解決すべき課題はいくつかあった。まず一つは手話をカメラから撮った画像には無駄な情報が多く含まれるということである。今回私達はこの課題を肌色抽出という方法で回避した。3.4.1 で述べたものを実装するのは難しいと判断し今回は動きのない手話の一種である指文字の認識を行うことにした。

(※文責：佐野将真)

3.5 最終発表

3.5.1 最終発表に関する活動

2017年12月8日、公立はこだて未来大学内で最終成果発表が行われた。最終成果発表に向け、スライドやポスターの制作と発表用のプレゼンテーション資料の制作また発声練習などを行った。

(※文責：佐野将真)

3.5.2 評価シートの内容と考察

最終発表で50人から受けた評価は、表3.2に示す結果となった。

最終発表では多くの質問や意見が寄せられた。まずは発表内容についての質問と意見を以下に示す。

- 目線がPCやスクリーンにばかり向いていた。
- 声が大きく聞きやすかった。
- わかりやすい説明だった。
- CNNが何かわからなかった。
- 他の発表に声が負けて聞こえない時があった。
- 内容を主軸にした発表のため、分かりやすい。
- 面白い題材を扱っていると思います。
- 抑揚がなく何が重要かわからない。
- 説明がもっと欲しかった。
- ポスター前でももう少し積極的に説明する方が良いと思います。

表 3.2 最終発表での評価

評価	発表技術	発表内容
10	6	10
9	7	6
8	14	16
7	10	10
6	4	3
5	6	2
4	2	2
3	0	0
2	0	0
1	0	0
無回答	1	1
平均点	7.4	7.9

- スライドを使用して詳しく説明してくれていたのが良かった
- 発表練習がよくなされている印象です。スライドの情報量も不足ない印象です。
- 手話を流れでとらえられるのがもう出来ているかどうかわからなかった。
- 発表者に落ち着きがないと思います。
- 早口で聞きとれないことが何回かあった。

次に発表技術についての質問と意見を以下に示す。

- 評価結果についてもう少し何か考察があれば面白いと思いました。
- 指文字画像を読み取った結果をデータとして示せ低他のが良かった。
- テストデータの結果が 70 % 以上と高い数値になっていたので、プロジェクトの目標は達成出来てると思う。
- 画像認識に興味があったのでとても参考になった。
- 目的と方法がはっきりしてる。個人的に非常に有意義な研究であると感じた。スライドも丁寧で、今何を話しているかがわかりやすかった。研究が成功していることも分かりやすく、聞いていて面白かった。
- データ収集やそれに対する評価が分かりやすく丁寧なため成果内容が分かりやすかった。
- 「学習」を使うということと、その技術を生かすこととの効果が十分に整理されていない様に思います。
- 自分達の目標とその問題点をしっかり説明しており自分達の行ってきたことをわかりやすく説明していた。
- 専門用語が多すぎる、CNN の説明が「畳み込みニューラルネットワークです」という内容で分かる人がどれだけいるだろうか？
- 誰でも手話を理解できるといいね！
- ディープラーニングを実装することを試みた面白い内容でした。
- 結果が得られていたので目標設定と結果が適正であると思った。
- AI の研究は広く行われているので、このプロジェクト独自の目標がもう少し強調されてい

るといいと思います。

- プロジェクトの進行スケジュール等についての言及があるとつたわりやすかった。
- ディープラーニングを用いた手話 (指文字) の認識は役に立つもので良かった。
- 正直よくわかりませんでした。自分のような人のために「何が結局どうなのか」を最後に示していただけると嬉しいです。
- 動画での検出ができると、色々な所で使用ができそう。
- ディープラーニングの応用方法についてしっかり考察出来ていると感じた。
- 着眼点がすごいと思いました。
- 顔から認識してる情報は現在ないのか

技術に関して言えば、声が大きく聞きやすかったというコメントが比較的多く見受けられたため、発声練習の成果が得られたと考えられる。しかし、PC やスライドを見ないで話せたら良かったというコメントも多く見られたため、発表練習自体が少し不足していたと考えられる。内容に関して言えば、説明不足とのコメントが多く見受けられた。特に CNN に関する説明がないとのコメントが多く、用語に対する説明を分かりやすく簡潔に説明出来ていなかった。

(※文責：佐野将真)

第 4 章 グループメンバーの活動

4.1 各人の活動

- 好本大地 (グループリーダー, 実装班)

前期の活動では, 参考プログラムを web 上から見つけ, そのプログラムに変更を加えることによってシステム開発を行う環境下で実行ができるようにした.

後期の活動では, 前期に変更した部分に加え Linux にインストールした Chainer のバージョンに合わせた記述に変更することで参考にしたプログラムの課題点であった GPU に対応させることができた. また, 最終発表用のスライドの作成を行った.

グループリーダーとしては, 全体をまとめ, 誰がどのような作業を行っているのかを把握しながら前期, 後期通してのプロジェクト活動を進めることができた. また, システム開発のスケジュールを立てることの難しさや, 作成したスケジュールに沿ってシステム開発を進めていくことの難しさなどを学ぶことができた.

(※文責: 好本大地)

- 佐野将真 (実装班)

前期では, web サイトに掲載されているコードを実際に動かすことで, 必要になるであろうライブラリーの導入方法や使い方を調べた. また, Python でディープラーニングを使うために必要なソフトの導入, 手話翻訳を実装するために必要であろう Python のライブラリーをインストールし, Linux の環境構築を行った. 後期では, 新しい実装方法の提案や, より効率の良いプログラムへの改良やなど主にプログラム開発を行った.

(※文責: 佐野将真)

- 山本大智 (収集班)

前期では CNN というニューラルネットワークの学習と課題解決のための手法を模索した. 後期では画像データの収集を主な活動として, 学習の際に画像データを認識精度を上げられるような形式に加工した. 具体的には画像データから肌色抽出を行うためのプログラムの作成した. また, 試験的に顔認識によって画像データの中から顔の部分を削除し, 肌色抽出に加えて手の部分のみを抽出できるようなプログラムを作成した.

(※文責: 山本大智)

- 岩田聖 (実装班)

前期では画像認識による手検出を行うために, 手話翻訳の先行研究および画像からの特徴点抽出に関する文献の調査を行った. 中間発表に向け, グループのポスター制作を行った. 後期では主に学習のための CNN プログラムの改良とデバッグを行い, 環境を整えた. また, データ班が用意した画像データを各行ごとに画像の振り分けとデータセットの作成を行った. 後期の最終発表用のポスターの作成を行った.

(※文責: 岩田聖)

- 東浦聡太 (収集班)

主に手話画像のデータセットの作成とデータ収集のプログラム開発を行った。また、手話翻訳の実装の前に、SignAloud による手話翻訳の先行研究について調べた。手話を学習させる際に少ないデータ数で効率よく学習をさせる必要があったため、データのデータ拡張技術について調べた。Keras を用いたデータ数を増やすプログラムを見つけ、手話を学習させるための画像が作成できるようにプログラムの改善を行った。プロジェクト中間発表、最終発表においてデータ収集班の活動をまとめた草案を作成した。

(※文責：東浦聡太)

- 正木大雅 (収集班)

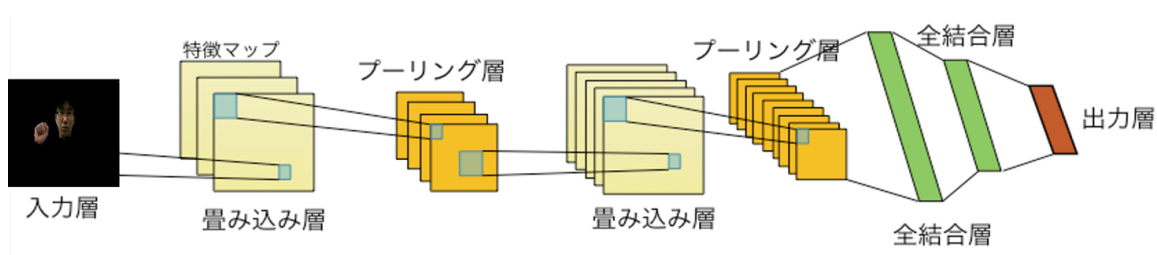
1 年を通して収集班としてディープラーニングで学習する画像データの作成を行った。前期では教科書を用いてのディープラーニングの学習や手話翻訳に関する先行事例の調査の他、メンバーからグー、チョキ、パーのハンドジェスチャ画像を様々な方向から撮影した画像を、分割、データ拡張等の作業を行い静止画像から手形状の認識を行うためのデータセットの作成を行った。後期では前期から引き続きデータセットの作成を行った。メンバーから挨拶や感謝、謝罪を示す手話の動画や指文字を行っている動画を撮影し、動画編集ソフトを用いて 60fps で取り込み画像への分割を行った。その後、収集班で協力し複数のデータ拡張方法の模索や実行、よりよく学習させるためにはどんな工夫が必要かなどを相談した。相談の結果としては様々な案が提案されたが、最終的に画像の hsv の値を限定し肌色抽出を行うことが決定し、データ拡張を行った指文字の画像全てにこれを適用した。プログラムに関しては web サイト上に掲載されているプログラムのソースコードを参考にフォルダの中身を全て読み込み、複数のデータ拡張を実行するプログラムを作成した。中間発表及び最終発表では全体の概要、A グループの発表を一回ずつ担当した。

(※文責：正木大雅)

第 5 章 実験

5.1 開発したもの

我々は画像認識の分野で広く用いられている学習手法の一つである CNN を使用し、手話を学習させ翻訳を行うシステムの開発を行った。CNN は主に画像認識の分野で幅広く使われている。これは、膨大な特徴を抽出するために畳み込みというフィルターを適用させ特徴抽出を行う畳み込み層と、これらの特徴をまとめ上げるプーリング層がある (図 5.1 を参照)。この層を繰り返す行うことにより、大きな画像の特徴量を小さな特徴量へ変換していき、画像認識を実現している。これらの特徴量を繰り返し学習を行うことで、各指文字に対する学習モデルを構築し、手話翻訳を実現する。



学習モデルの構築!!

図 5.1 Convolutional Neural Network[12]

では、具体的にどのようにして手話翻訳を行うのかを説明する。先ほど説明した CNN を用いて、対象者が手話を行っている画像を多数与え、その手話の特徴を学習させる。ディープラーニングによって作成されたモデルを用いて、実際に画像を読み込ませた時に正しい意味を理解し、翻訳を行うシステムの作成を行った。だが、手話は手や顔の動きなど複雑に絡み合っており、意味をなす物であり、容易に認識させることが難しい。そこで、今回は手話の中でも比較的手の動きが少ない、指文字を学習させ翻訳を行うことにした。指文字とは、手の形を文字言語に対応させた視覚言語の一要素である。手話は音声言語や書記言語より語彙数が格段に少ない為、手話単語にない単語、人名・地名・固有名詞などは、指文字を使って一字一字ずつ表現する。(図 5.2 を参照)

(※文責：岩田聖)

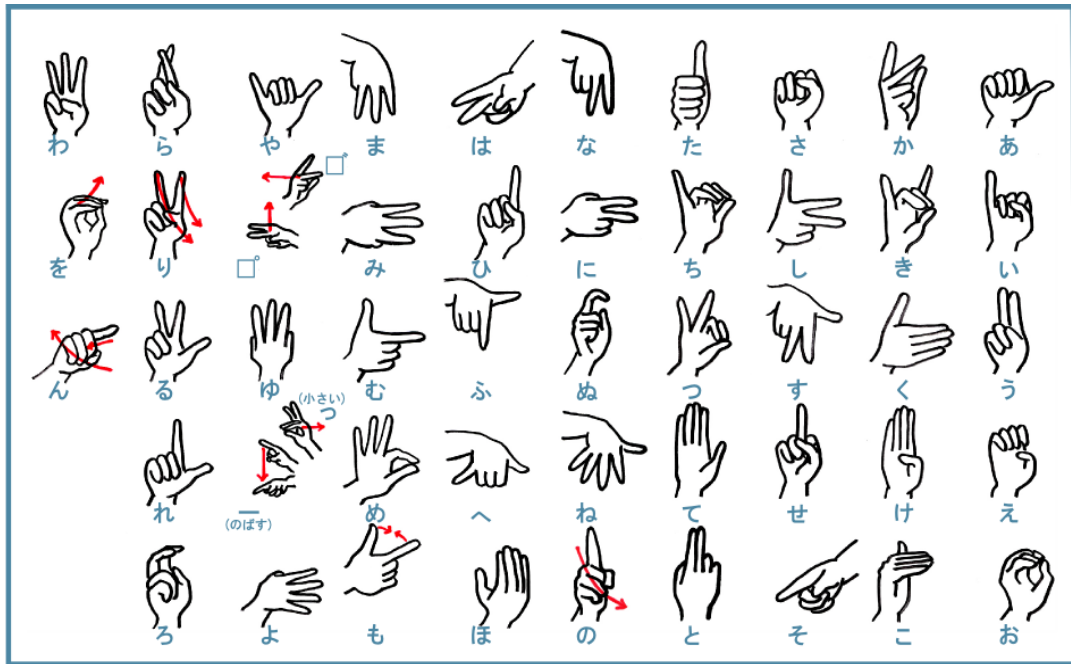


図 5.2 指文字 [13]

5.2 作成したデータセット

手話翻訳を行う上で、手話の特徴を学習させるためには実際に手話の動作を行っている画像が大量に必要である。しかし、我々が調べたところ手話を行っている画像はごく一部の少数でしか見つけることができなかった。翻訳を行うためには我々がデータセットを作成する必要があると考えた。

データセットの作成方法について説明する。グループメンバーの PC 内蔵の web カメラを使い、15 秒程度の手話の動画を撮影した。個々のぼらつきを出すために、手の位置などを微妙に変えながら行った。これらの画像を 60 フレームごとに切り分けた。今回は本プロジェクトメンバー 12 名それぞれが行った行の指文字を行った。図 5.3 は切り出した画像である。切り出した画像には、背景や手話話者の服など、手話の認識に必要な部分以外の情報が多量に含まれている。指文字の認識に関して、これらの情報が認識に必要なないと判断したため、肌色以外の部分を除去し、黒く塗りつぶすことによって、肌色部分のみを残す方法によりデータセットの品質向上を図った。さらにデータセットを増やすため、Keras ライブラリに含まれる、画像をランダムに変形・加工を行うメソッドを使用した。このメソッドで画像を上下左右にランダムに変形し、写る位置の違いによって認識精度の差を無くすためのデータセットのデータ拡張を行った。

データ拡張を行ったデータには、ずれて手が写っていない画像が出てしまうため、画像の厳選を行い品質の向上を行った。最終的な画像枚数は各行につき、21000 枚とした。その内訳は、行の 5 文字の合計枚数を 6000 枚、学習させる行とは異なる行の文字を 15000 枚とした。この 21000 枚を学習の訓練に使う画像が 20100 枚、正しく認識が行われているかを確認するためのテスト用画像を 900 枚として分けた。今回は各行～た行までの学習・認識を行うので、総枚数は 84000 枚となる。(図 5.4 参照)

(※文責：岩田聖)

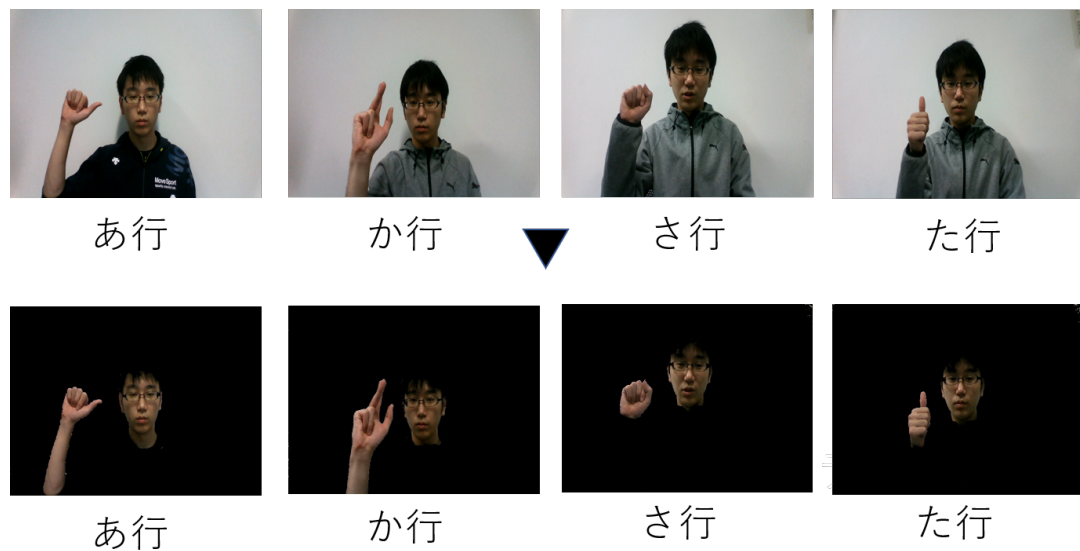
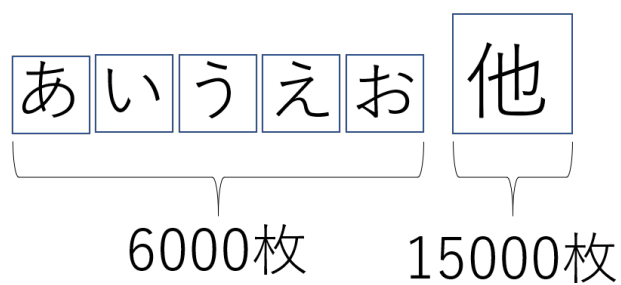


図 5.3 切り出した画像および肌色抽出処理

データセット

行のデータ総数：21000枚
 }
 訓練用データ数：20100枚
 テスト用データ数：900枚



21000×4

 ▼

 あ行～た行総枚数: 84000枚

図 5.4 データセットの詳細

5.3 実験環境

まず, 実験を行った環境について説明する. 実験機器の環境構成は Ubuntu14.04 の OS を使用し, メモリ 32GB, 500GB の SSD と 1TB の HDD, NVIDIA 製 GPU(GEFORCE GTX1080), Intel 製 CPU(Core i7) を搭載した PC を使用し, 学習実験を行った. 次に学習の方法について説明する. 5.2 で用意したデータセットを使用し学習を行った. 学習方法は 5.1 で説明した CNN を使用し, 入力層, 4 層の中間層, 出力層の計 6 層のネットワークを構築した. また, Minibatch のサイズを 20 とすることにより, 一度の学習に 20 枚を使用し, 効率的に学習を行った. また epoch 数は 50 とし, 各行ごとにそれぞれ 5 文字と行以外の文字を判別する 6 値分類を行う学習モデルの構築を行った.

(※文責: 岩田聖)

第 6 章 結果

5.3 で説明した実験を行った結果を説明する (図 5.4 参照). それぞれの行に対する学習の訓練精度はそれぞれ, あ行が 99.49%, か行が 99.61%, さ行が 99.25%, た行が 98.42% という結果になった. これらの学習が正しく行われているかを確かめるテスト精度はそれぞれ, あ行が 71.33%, か行が 78.33%, さ行が 77.99%, た行が 75.30% という結果になった. 今回は, あ行～た行の 4 行だけの判別になったが, それぞれ, 学習精度が 99% 以上, テスト精度が 71～78% と良い結果が出た. この結果を踏まえると, 今回できなかったな行以降も同様の手法を用いて学習が可能と考えられる.

学習済みモデルの精度

訓練データに対する精度 : 99%以上

テストデータに対する精度 : 71～78%

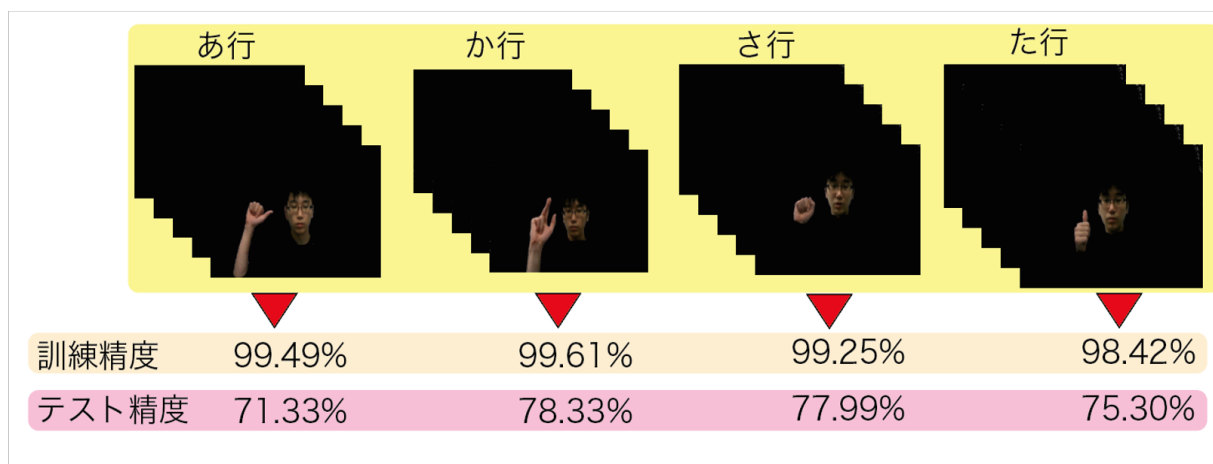


図 6.1 実験結果

(※文責 : 岩田聖)

第 7 章 考察

実験を行った結果から、訓練精度に比べテスト精度の値が低めの結果となってしまったことに対しては、学習に用いたデータセットの種類が不足していたということが原因の一つであると考えられる。学習に用いたデータセットは、本プロジェクトメンバーよりあ～た行それぞれの指文字を収集したが、12 人分のみに留まってしまったため、データを収集する人数を増やすことによって、学習に使用するデータの種類が増加するのでテスト精度の向上が見込めるのではないかと推測される。

また、あ～た行のテスト精度の結果において、か行およびさ行のテスト精度が他の二つの行と比べ比較的高い精度での識別を行うことができているという点においては、これらの 2 つの行をあらわす指文字の特徴量が簡単な手の形、例えば握りこぶしを作った状態から親指だけを立たせた形などの比較的抽出しやすい形であらわされているためであると考えられる。しかし、あ行では「あ」や「え」を表す指文字図 7.1, 図 7.2 やた行では「ち」や「つ」を表す指文字図 7.3, 図 7.4 において、使用される指文字に丸めた形が含まれていたため特徴量の抽出が難しくなってしまうと推測される。これは、 200×200 のサイズの画像を学習に用いたため識別に必要な指の特徴の部分がつぶれてしまったり、ぼやけてしまったりしてしまったために指文字を識別しづらくなってしまったためであると考えられる。この問題を解決するためには学習に用いる画像のサイズを大きくすることによって、より鮮明に特徴量を取得することができるようになり識別精度が向上すると考えられるので、問題の解決が見込めるのではないかとと思われる。

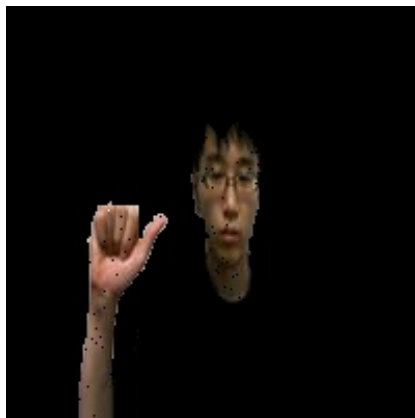


図 7.1 「あ」を表す指文字



図 7.2 「え」を表す指文字



図 7.3 「ち」を表す指文字

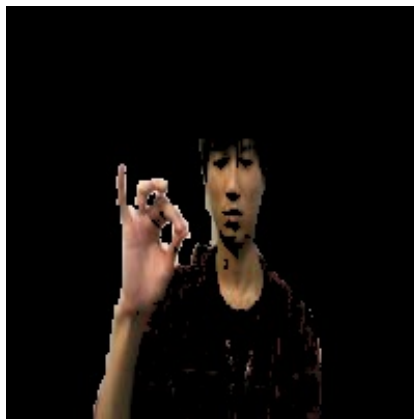


図 7.4 「つ」を表す指文字

今回の結果から、あ～た行においての学習済みモデルの正答率は比較的良好な結果となったことを受けて、な行より後の行に対しても同様のデータセットを作成し学習を行わせることによって 50 音すべての行の指文字に対応した学習済みモデルを作成することが可能であると考えられる。

(※文責：好本大地)

第 8 章 今後の展望

今回のグループ活動の成果物から今後の展望として図 8.1 のような、入力した画像から指文字を翻訳するシステムの開発があげられる。このシステムの機能としては、連続した指文字の画像を入力として受け取ることによって、それらの受け取った指文字がどのような意味なのかを出力するというシステムとなる。今回の成果物では、開発環境を整えることに時間がかかってしまったことやデータセットを作成することに時間がかかってしまったために、あ〜た行のそれぞれの行ごとに対応したモデルを作成し、合計 20 個の指文字の識別をするにとどまったが、これらの作成したモデルを 50 音すべてに対応させることによって指文字の翻訳システムとして実用化できるのではないかと考えられる。

また、今回作成したシステムでは web カメラで撮影した動画をコマ割りにしそれぞれの画像に対して加工を行ったものをデータセットとした。しかし、この方法では一つの指文字や手話のデータセットを作成するのに時間がかかりすぎてしまうなどの問題があった。この問題を解決するために、web カメラで撮影した動画から直接データセットの作成が行えるようなプログラムを作成し、システムに組み込むことによって問題の解決が見込めるのではないかと考えられる。

さらに、ディープラーニングでの手話翻訳における二つの課題点のうちの一つである動きのある動作の検出については、手の動きを抽出することができるアルゴリズムを応用することにより、手話の識別に必要である手や指の動きの特徴を動画から直接手話を認識することが可能となるのではないかと考えられる。また、もう一つの課題点である手話の意味識別に必要となる顔の特徴量を抽出するという課題に関しては、OpenCV を使用することによって対象者の顔の表情を認識することによって顔の特徴を抽出することが可能となると考えられる。

これらの技術を用いてシステムを開発することによって、手や指の動きだけでなく指文字や手話を行っている対象者の顔の表情をリアルタイムで認識することが可能となるので、当初のグループ到達目標であった特殊機器を必要としない手話認識を行うことが可能となり NHK 手話ニュースを翻訳することが可能となるのではなかと考えられる。

また、日本の手話には日本語対応手話と日本手話がありそのどちらの表現にも対応させることや、英語での表現に対応させることによって、より本格的な手話翻訳システムの開発が可能になるのではないかと考えられる。

(※文責：好本大地)

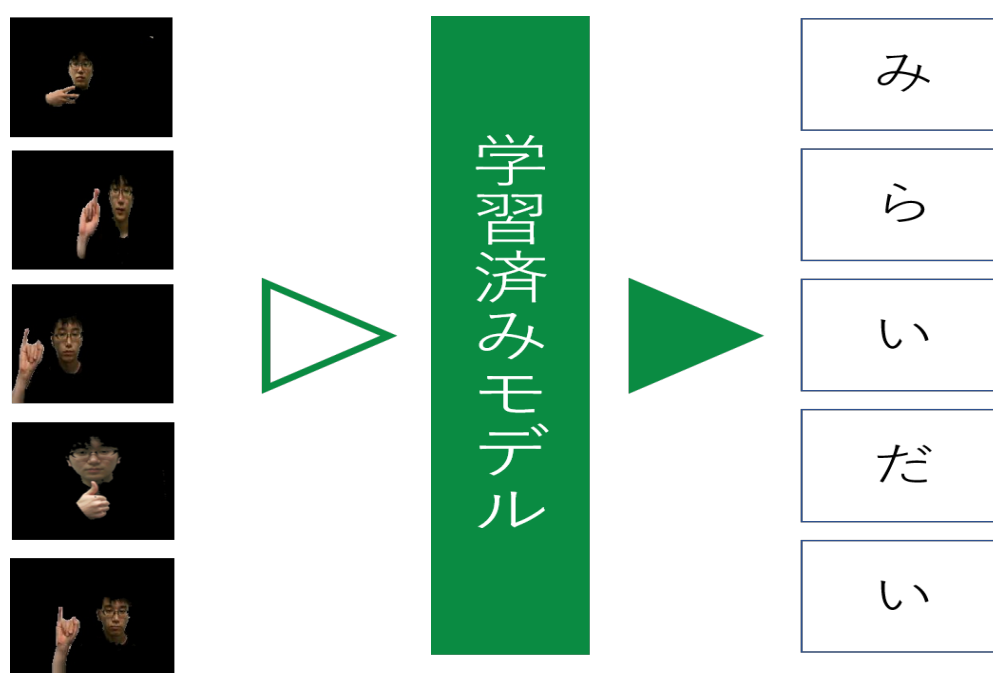


図 8.1 指文字翻訳システムの例

第 9 章 まとめ

本グループでは、動画中から直接手話を翻訳するシステムを開発することを目標に活動を行ってきた。前期の活動では、ディープラーニングについての知識を取得することやグループ活動の最終目標やシステムの実装方法などについての話し合いなどを行った。ディープラーニングについての知識を取得することに関しては、参考書などを用いることによって必要最低限の知識を得ることができた。また、システムの実装方法についての話し合いに関しては、Kinect や Leap Motion といった機器などについての使用方法を模索、データセットの作成方法、作成したデータセットの拡張をどのように行うのかなどの話し合いを行った。最終的には、データセット作成に使用するコンピュータの機能が不足していたため、これらの機器の使用を断念し web カメラを用いることによりデータセットを作成することとなった。手話を識別するためのプログラムの作成については、ケン玉の技を識別するという論文と web サイトを参考にすることにより開発を行っていくこととなった。同時に、Python, OpenCV, Chainer などの導入や Ubuntu14.04 の環境設定などを行いシステムの開発を行うための環境を整えた。

後期の活動では、前期の活動で話し合いを基に実際に学習を行うためのデータセットの作成や入力した画像がどの手話を表すのかを識別するシステムの開発などを行った。データセットの具体的な作成方法は、web カメラで撮影した手話や指文字の動画を一秒間に 60 コマずつに分解し、それらの画像にぼかしなどの加工を加えることにより画像枚数を増やした。さらに、それらの画像に対し肌色抽出処理を実行することによって学習に用いるデータセットとした。システムの開発に関しては、参考にした web サイトに掲載されているプログラムを基に Ubuntu14.04 の環境に合うように修正を加えた。また、参考にしたプログラムでは GPU で実行することができなかったため GPU に対応させることによってより素早く学習を行えるようにも改良を加えた。時間的な問題や、技術的な問題により当初予定していた手話翻訳ではなくその前段階であるあ～た行までの指文字の翻訳にとどまってしまったが、結果としてそれぞれの訓練精度は 99% 以上、テスト精度は 71%～78% の精度で認識させることに成功した。今後は、これらの結果を応用することによって手話の認識までを行えるようなシステムの開発を行っていく必要がある。

前期、後期の活動を通して当初予定していたリアルタイムに手話を翻訳するという目標を達成することができなかったが、プログラム班、データ収集班ともに通年の活動を通してディープラーニングに対する様々な知識や、画像処理に対する様々な知識を習得することができた。

(※文責：好本大地)

参考文献

- [1] Leap Motion, <https://www.leapmotion.com/>
- [2] 小林実来, 武田敦志, 「Leap Motion Controller を用いた手話動作認識手法の実装と評価」 (情報処理学会東北支部研究報告 Vol.2016, 2017 年)
- [3] SignAloud, <http://lemelson.mit.edu/winners/thomas-pryor-and-navid-azodi>
- [4] Kinect for Windows, <https://developer.microsoft.com/en-us/windows/kinect>
- [5] 下元正義, 「Kinect for Windows による手話認識」 (MVP Community Camp 2014 みずほ情報総研講演資料), <https://www.mizuho-ir.co.jp/seminar/lecture/pdf/lecture140315.pdf>
- [6] Hidden Markov Model Toolkit, <http://htk.eng.cam.ac.uk/>
- [7] ゼロから作る Deep Learning, 斎藤康毅, オライリージャパン
- [8] Chainer による実践深層学習, 新納浩幸, オーム社
- [9] DeepLearning 用フレームワーク Chainer を試してみたその_2,
<http://www.casleyconsulting.co.jp/blog-engineer/python/deeplearning>
- [10] <https://prog-8.com/>
- [11] OpenCV による画像処理入門, 小林正直, 中村悦子, 中村恭之, 講談社
- [12] <http://www.dmprof.com/jp/news/2016/6595/>
- [13] <http://手話辞典.com/index.php?>