

手書き文字認識と対話型訂正機構による採点システムの構築

菅 文哉¹ 山本 晃平¹ 村尾 和哉¹ 望月 祐洋² 西尾 信彦¹

概要：人間の活動を計算機により認識して管理・活用することで、業務の効率化や生活の質の向上が期待されている。これらのシステムは、センサデータをもとに人間が現実世界で行った動作を認識し、結果を計算機上（仮想世界）に記録して、結果に応じた情報を提示する。ここで、動作を認識する過程において誤認識は必ず発生すると考えられるため、認識結果を訂正する機構が必要となる。また、同じ作業を再度行うことが許されない状況もあるため、現実世界に影響を与えない訂正機構が必要である。本研究では、計算機が人間の動作を認識し作業を支援するシステムの一例として、手書きによる答案の採点作業の支援システムを構築する。提案システムでは、採点作業中に、ペン型マウスによる「○（丸）」、「△（三角）」、「/（バツ）」三種類の図形の描画を画像認識により認識する。「△」は部分点の記述も考慮しており、また、付属のボタンをクリックすることで音声フィードバックを用いた訂正機構が起動し、ユーザの行動を大きく変えずに対話性に長けたシステムを構築できる。採点後は、得点の計算、表計算ソフトへのデータの入力を自動で行うことで作業を効率化している。評価として、10問からなる答案を10人分用意し提案システムを利用した場合と、利用せずに手書きで採点と集計作業を行った場合で、所要時間を計測した。提案システムでは採点作業自体の時間は若干長いですが、採点が終了すると自動的に得点が集計されるため、提案システムの方が全体の作業時間を大幅に削減できることを確認した。

1. 序論

人間の活動を計算機により認識して管理・活用することで、業務の効率化や生活の質の向上が期待されている。例として、看護師の活動を認識して看護日誌を自動作成するシステム [1] や、車などの組立作業を認識して手順書をモニタに表示するシステム [2] などが存在する。

これらのシステムは、センサデータをもとに人間が現実世界で行った動作を認識して、結果を計算機上（仮想世界）に記録し、結果に応じた情報を提示する。ここで、動作を認識する過程において誤認識は必ず発生すると考えられるため、認識結果を訂正する機構が必要となる。しかしながら、従来システムの多くはその機構を有していない。人間が現実空間で既に行った動作を取り消したり訂正することはできず、また、同じ作業を再度行うことが許されない状況もあるため、現実世界に影響を与えない訂正機構が必要である。

本研究では、計算機が人間の動作を認識して作業を支援するシステムの一例として、手書きによる答案の採点作業の支援システムを構築する。図1に従来システムのフローと提案システムのフローを比較して示す。提案システムで

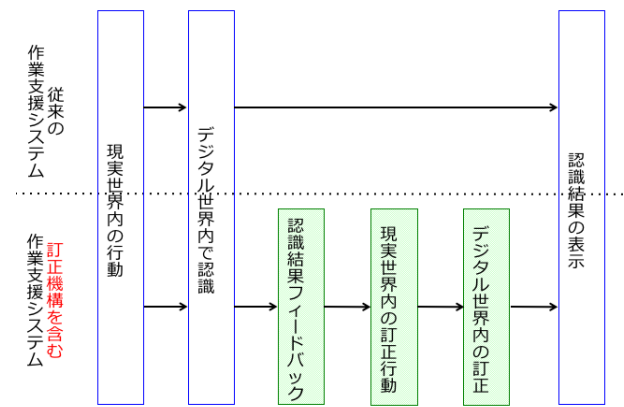


図1 従来型システムおよび本研究におけるシステムフロー

は、採点作業中において、ペン型マウスによる「○（丸）」、「△（三角）」、「/（バツ）」三種類の図形の描画を画像認識により認識する。「△」は直後の部分点の記述も考慮している。ペンマウスに付属するボタンをクリックすることで音声フィードバックによる訂正機構が起動し、図形および部分点の訂正を行うことができる。これにより、認識ミスが発生した場合や、採点作業を行っていないにも関わらず何らかの認識が行われた場合にユーザに伝えることができ、誤って認識された結果の訂正が可能になる。また、音声によるフィードバックを用いることで、コンピュータの画面を見ることなく採点作業に集中できるため、ユーザ

¹ 立命館大学情報理工学部

² 立命館大学総合科学技術研究機構

の行動を大きく変えずに対話性に長けたシステムを構築できる。採点作業後には、得点の計算を行い、表計算ソフトへのデータの入力を自動で行うことで、ユーザの負担を減らしつつ作業の効率を向上させる。

2. 関連研究

採点システムに関する研究は多く行われており、越智ら [3] はタブレット端末を用いて選択肢認識機能を実装した演習システムを開発した。このシステムでは、JPEG で作成された選択肢型の問題への回答を学習者がタブレット上で手書きによって行うと同時に、画像認識により自動的に採点を行う。コンピュータベースの試験は増えてきているが、まだ紙ベースの試験は多く、受験者全員にタブレット端末を用意することは管理やコストの面からも現実的には難しい。また、選択肢型の問題しか扱えないため、記述式の問題には対応できない。

市村ら [4] は、大学内における大規模講義を支援することを目的として、小テストを用いた自動採点を行う講義支援システムの開発を行った。このシステムでは、紙の答案に書かれた学籍番号、選択肢式の解答、および、数字をゼロックス社が開発したグリフシートを利用して認識し、結果をコンピュータに自動入力する。そして、各学生に電子メールでフィードバックを自動送信する機能をもつ。しかし、手書きで書かれた文章やグラフの採点には不向きであるという問題点がある。

手書きによる採点方式は、選択式、記述式、論述などあらゆる回答方式に対応できるが、採点後に手作業での点数の計算や記録が必要となる。手書きによる採点作業をシステムが認識し、認識結果を表計算ソフトと連携することで点数の計算や記録を自動化できるが、筆者らの知る限りそのようなシステムは存在しない。

3. 提案システム

本章では、手書き文字認識と対話的な訂正機構をもつ手書きによる採点の作業システムの要件および提案システムの構成を述べる。

3.1 要件

現状の採点システムは、限定された環境における採点作業を対象としたものである。しかし、紙媒体に直接手書き解答を行う解答方式が多くの場面や環境で主流となっているため、それらを補助、支援するシステムが存在しないことは問題である。加えて、採点作業には、ただ丸付け作業が行われるだけでなく、その結果を分析するための、得点計算、データ化する作業も含まれる。また、システムとして対話的な訂正機構を有していることは重要である。本システムでは、想定環境として教育現場における紙の答案の採点業務を対象としている。

提案システムを利用するにあたって、採点者であるユーザは、実際に答案に「○」、「/」、「△」などの図形や数字を描くことが可能なペン型マウスを用いて答案の採点作業を行う。また、採点を行うユーザが描画した図形や数字の認識を行い、問題ごとの正誤結果のデータ化および得点計算を行う。その際に認識結果をユーザに返す必要があるが、そのフィードバックを返す機構や認識結果の誤りがあった場合の訂正機構は採点作業への負担とならないようなものでなければならない。提案システムでは採点結果を記録するコンピュータとペン型マウスを無線で接続し利用することを想定している。

3.2 提案システムの概要

提案システムのフローチャートを図 2 に示す。はじめに、システムは採点に必要な情報（学生数、問題数や配点など）が入力されたエクセルファイルを読み込み、採点情報として登録する。採点開始後は、ユーザが描画した図形を画像認識を用いた特徴点抽出によって 3 種の図形（○、△、/）に分類する。システムは認識結果を音声でユーザにフィードバックし、認識結果が誤っていれば、ユーザはペン型マウスのボタンをクリックして訂正する。訂正後の結果は再度音声でフィードバックされる。認識結果が○または/の場合は、登録されている配点に従って、採点結果として登録される。△の場合は、ユーザは続けて部分点の描画を行い、システムは数字を Convolution Neural Network (CNN) で認識する。システムは認識結果を音声でユーザにフィードバックし、認識結果が誤っていれば、図形と同様にユーザはボタンをクリックして訂正する。認識結果の数字が部分点として登録される。ここで、図形認識および数字認識において認識結果が正しい場合、ユーザはボタンをクリックせずに次の描画を行うことで認識結果を確定させる。

最終的にエクセルファイルで設定していた問題数と人数分の採点が終了すると、各人のトータルスコアを計算して自動的に集計されエクセルに採点結果が書き出される。以降では、ユーザの描画データの取得、画像認識、音声フィードバックとボタンクリックによる対話型訂正機構を説明する。

3.3 描画データの取得

提案システムは、機械学習を用いるため、計算の自動化ライブラリが豊富なプログラミング言語 python (ver.3.6.1) をベースにして macOS Sierra (ver10.12.4) で動作するアプリケーションとして実装した。このアプリケーションを起動すると、図 3 に示すように画面上にウィンドウが表示され、PC 上のマウスカーソルは自動的にウィンドウの中央に移動する。描画を行なっている間にカーソルの座標を取得し続け、描画が終了した時点でその軌跡を含むウィン

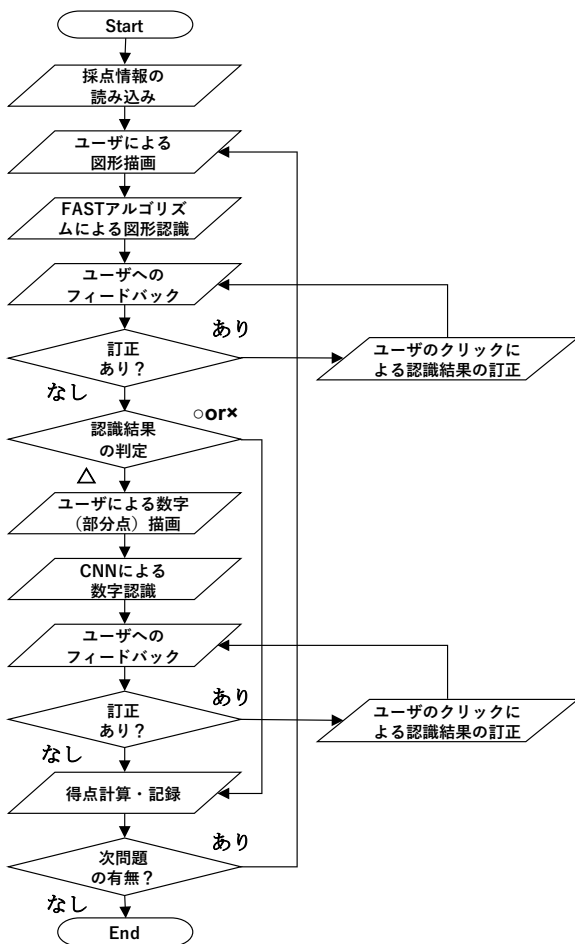


図 2 提案システムのフローチャート

ドウ全体を1つの画像として保存する。そのウィンドウの画像に含まれる軌跡を画像認識することでユーザのペンマウスによる図形や数字の描画を認識する。ユーザは描画する際にクリックする必要はなく、ユーザのマウスが動き出した時から軌跡を記録していくようになっており、描画の終点判定はある程度カーソルの動きが止まった時点で認識に移るようにしている。また、描画を繰り返すうちにカーソルが画面外に出ることを防ぐために、一回の描画が終わると自動的にウィンドウの中心にカーソルが戻るようになっている。

3.4 画像認識

提案システムでは、マウスの軌跡を画像に変換した後、画像認識によって図形および数字認識を行う。本節では、その認識方法について説明する。

3.4.1 図形および数字の切り出し

画像認識を行う際は、ウィンドウ全体の画像に対して処理を行う。図形認識の場合はその中に含まれる軌跡の特徴点を抽出するだけでなので、縮尺を修正する必要はない。したがって、ウィンドウ全体の画像をそのまま画像認識に使うことができる。しかし、数字認識の場合は、機械学習の事前学習において使用する数字のデータセットの縮尺に

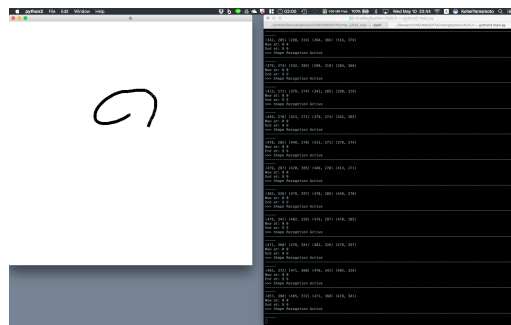


図 3 ウィンドウへの描画時のスクリーンキャプチャ

合わせる必要があります。縮尺が合わなければ数字認識を行うことができない。今回使用した数字のデータセットが正方形なので、その縮尺に合わせるために正方形に切り出す必要がある。したがって、数字認識の際はウィンドウの全体画像から認識の対象物となる数字を切り出す処理を行う必要がある。図 4 に数字認識の際に数字を切り出すアルゴリズムを示す。

はじめに、数字を切り出すために、画像を一旦グレースケールに変換して、構成しているピクセルの値を一つずつ走査していく。そして、画像の左上を0として、数字が描かれた画像の最大の x 座標 max_x および最小の x 座標 min_x 、それから最大の y 座標 max_y および最小の y 座標 min_y を求める。その後、描かれた数字の中心点 C を求め、数字が y 軸方向に比べて x 軸方向に長い場合、 $max_x - min_x$ の辺長で正方形に数字の切りだしを行う。逆に数字が x 軸方向に比べて y 軸方向に長い場合、 $max_y - min_y$ の辺長で正方形に数字の切りだしを行う。

3.4.2 図形認識

図形認識において分類される図形の種類としては、「○ (丸)」、「△ (三角)」、「/ (バツ)」の3種類がある。したがって、図形の認識により出力される値はこの3つに限定される。図形の分類は、FAST アルゴリズム [5] にしたがった特徴点抽出によって行う。FAST アルゴリズムは、任意の画像のあるピクセル p が周囲の点の暗い明るいという情報が連続的であるかどうかということによってコーナ (特徴点) の検出を行う。任意のピクセル p が特徴点となる条件は、ピクセル p を中心とする円上の16ピクセルにおいて、 p の画素値+任意の域値 t よりもすべて明るい、または p の画素値-任意の域値 t よりも暗い n 個の連続ピクセルが存在することである。ここで n の値は高速試験のため12が選ばれている。また、連続的であるかどうかのみの情報だけでなく機械学習によって得られた決定木によって任意のピクセルのみを比較することで高速化が可能となる。このアルゴリズムのより詳細な説明はここでは省略するが、本研究ではFAST アルゴリズムによって抽出された特徴点の総数によって、画像を分類する。

FAST アルゴリズムによる特徴点抽出は OpenCV3.0

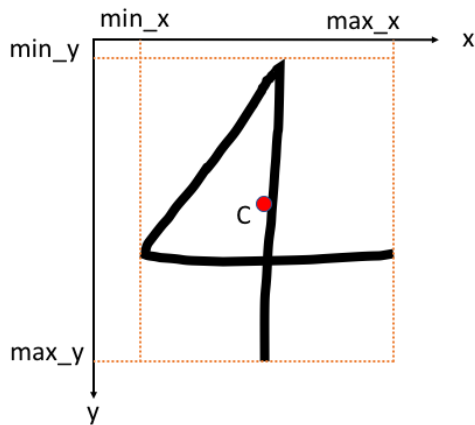


図 4 数字認識の際に数字を切り出すアルゴリズム

(ver.3.2.0) のライブラリを用いて実装した。また、特徴点の総数が 8 未満のときは「/」、9 以上 23 未満のときは「△」、それ以上のときは「○」として図形を描画した場合が事前実験において最も精度が良かったので、そのように閾値を設定した。また、特徴点が 2 以下の場合、意図しない描画であると判断し、画像認識を行わないようにすることで冗長性を高めた。

3.4.3 数字認識

提案システムを使用するにあたって、実際の採点作業において部分点が 2 桁になるという状況は少ないと考えられ、また、「/」は 0 点として扱われるという状況が多いため、数字認識において分類される数字の種類は 1 から 9 までの 9 つの数字とした。したがって、数字認識により出力される値はこの 9 つに限定される。数字の分類は、あらかじめ 1 から 9 の数字のデータセットを多層構造の畳み込みニューラルネットワーク (CNN) を構築し、学習し、作成したモデルでテストして分類する。図 5 に今回使用する CNN の処理を示す。CNN において、畳み込み層とサブサンプリング層は交互に 2 組と 1 つの全結合層をもつ構成とする。活性化関数には、画像認識の分野においてよく利用される ReLU を用いる。出力層の活性化関数にはソフトマックス関数を用いて、各数字である確率を求める。学習は、出力層から入力層に向かって誤差を逆伝播させる誤差逆伝播法を用いて各層の勾配を求めていきながら、Adam によって重みの調整を行う。予測と正解の差を求める損失関数には、交差エントロピーを用いる。

提案システムでは、CNN の機械学習による数字の認識を実現するために、Google が開発しオープンソースとして公開した、機械学習のソフトウェアライブラリ TensorFlow^{*3}

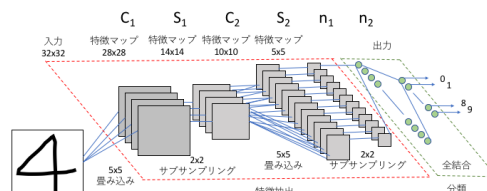


図 5 畳み込みニューラルネットワーク (CNN) のイメージ^{*2}

を用いて、手書き数字のデータセット MNIST^{*4} を 20000 回事前学習させることで実装した。また、ユーザは、ボタンをクリックせず、マウスの移動のみにより図形や数字を描画するため、図 6 に示すように数字の「4」「5」「7」は一筆書きで描く仕様とした。CNN を用いた数字認識においても、特徴点が 3 以下の場合、意図しない描画であると判断し、画像認識を行わないようなことで冗長性のある仕様にした。

3.5 対話型訂正機構

図形認識および数字認識を行う際、現在何問目の採点であるかを示した上で、認識結果を「○」であれば「まる」、「1」であれば、「いち」というように全て音声でユーザにフィードバックを返すようにした。音声で認識結果の正誤を確かめられることで、ユーザは PC の画面を見る必要がなくなり、採点作業に要する労力を少なくできると考えられる。

音声フィードバックにより、ユーザは認識の正誤を確かめることができるが、認識結果が間違っていた場合には訂正が必要となる。もし、図形の認識結果が誤っていた場合、ペン型マウスに付属しているボタンをクリックすることで採点結果の入れ替えによる訂正を行う。もし、数字の認識結果が誤っていた場合、ペン型マウスに付属しているボタンを訂正したい点数分クリックすることで採点結果の訂正を行う。例えば、「3」と描画を行なって誤認識された場合、訂正ボタンを 3 回クリックすることで訂正する。これにより、音声フィードバックと連携したより対話的な訂正機構を実現できる。

4. 評価

本章では、提案システムの有効性を評価するために実施した、実際に採点作業を行う実験およびその結果について述べる。

4.1 評価環境

評価実験では、被験者 5 人に対して、簡単なトレーニング

^{*1} OpenCV:FAST Algorithm for Corner Detection, OpenCV, 入手先 <http://docs.opencv.org/3.1.0/df/d0c/tutorial_py_fast.html> (参照 2017-05-02)

^{*3} Google : TensorFlow, An open-source software library for Machine Intelligence, Google, 入手先 <<https://www.tensorflow.org/>> (参照 2017-05-02)

^{*4} THE MNIST DATABASE of handwritten digits : THE MNIST DATABASE of handwritten digits, Yann LeCun <<http://yann.lecun.com/exdb/mnist/>> (参照 2017-05-02)

^{*2} OpenBook:deep mnist for experts(CNN), OpenBook, 入手先 <<https://openbook4.me/projects/238/sections/1557>> (参照 2017-04-30).



図 6 一筆書きによって描いた「4」「5」「7」

グを行なった後、3種の図形および9種の数字をそれぞれ10回ずつ描画してもらうことで図形認識率および数字認識率を評価した。

その後、提案システムを用いて、実際に採点作業において有用であるかどうかの評価を行なった。評価実験では、図 7 に示すように、10問からなる100点が満点となる答案を10人分用意し、ユーザに解答と答案を見比べてもらい採点作業およびエクセルへの集計作業を行ってもらい、その所要時間を計測した。各問いは全て4つの英単語から成っており、単語が全て同じであれば採点用の枠に「○」を、1つ以上異なっていれば「/」を描画してもらう実験と、単語が全て同じであれば採点用の枠に「○」を、1つだけ異なっていれば「△」を、2つ異なっていれば「/」を記述してもらう実験の2パターン行なった。△を記述した後の部分点については、ユーザに1から9までの任意の数字を描画してもらった。

提案システムを利用して採点作業を行なった場合と、利用せずに手書きで採点と集計作業を行った際の所要時間の比較をするが、公平性を保つため、システムを利用した採点を行なってから手書きの採点を行う被験者と、手書きの採点を行なってからシステムを用いた採点を行う被験者にわけた。

4.2 手書き図形認識の精度の評価

各被験者の図形認識の精度を表 1 に示す。表の縦軸は描画した図形を、横軸は認識された結果を表している。表 1 からわかるように、被験者5人全てにおいて、「/」の認識率は100%で、「○」と「△」の間で誤認識が発生していることがわかる。「○」が「△」として誤認識される原因として、図 8 に示すように、「○」の描画が小さすぎると、特徴点の総数が23未満となり、「△」として誤認識されてしまうということが考えられる。逆に、「△」が「○」として誤認識される原因として、図 9 に示すように、「△」の描画の際に角を緩やかに書いてしまった場合や辺が丸みを帯びたものになっている場合、つまり、「○」に近い形になってしまった場合において、特徴点の総数が23以上となり、「○」として誤認識されてしまうということが考えられる。

1. 私はペンを持っています I have a pen	1. 私はペンを持っています I have an pen	<input type="checkbox"/>
2. 私はサッカーをすることが好きです I like playing soccer	2. 私はサッカーをすることが好きです I like playing soccer	<input type="checkbox"/>
3. 私は京都へ行きました I went to Kyoto	3. 私は京都へ行きました I went to Kyoto	<input type="checkbox"/>
4. 私はりんごを食べます I eat an apple	4. 私はりんごを食べます I eat an apples	<input type="checkbox"/>
5. あなたは先生ですか? Are you a teacher?	5. あなたは先生ですか? Are you a teacher?	<input type="checkbox"/>
6. 私は毎日英語を話します I speak English everyday	6. 私は毎日英語を話します I speak English everyday	<input type="checkbox"/>
7. これは私のバイクです This is my bike	7. これは私のバイクです This is my bike	<input type="checkbox"/>
8. 今すぐ手を洗ってください!! Wash your hands now!!	8. 今すぐ手を洗ってください!! Wash your hands now!!	<input type="checkbox"/>
9. 今日は晴れています It is sunny today	9. 今日は晴れています It are sunny today	<input type="checkbox"/>
10. 今日は疲れました I am tired today	10. 今日は疲れました I are tired today	<input type="checkbox"/>

解答の答案

答案

図 7 評価実験で用いる解答と回答

表 1 各被験者の図形の認識精度

O	被験者 1			被験者 2			被験者 3			被験者 4			被験者 5			認識精度
	○	△	/	○	△	/	○	△	/	○	△	/	○	△	/	
○	10			8	2		10			8	2		9	1		90%
△	2	8		1	9			10		1	9		3	7		86%
/			10			10			10			10			10	100%



図 8 「△」と誤認識される「○」の例



図 9 「○」と誤認識される「△」の例

4.3 手書き数字認識の精度の評価

各被験者の数字認識の精度を表 2 に示す。表 2 からわかるように数字の「4」「5」「7」の認識率が他の数字に比べてよくないことがわかる。これはおそらく普段一筆書きで書いていない数字を一筆書きで書くような仕様にあまり慣れなかったためであると考えられる。「4」が誤認識される原因としては、図 10 に示すような赤線で囲った部分が短いと、「9」により近い数字として認識されてしまうのだと考えられる。「5」が誤認識される原因としては、図 11 に示すような赤線で囲った部分が繋がってしまうと、「6」や「3」により近い数字として認識されてしまうと考えられる。「7」が誤認識される原因としては、図 12 に示すように全体が斜めに描画されてしまうと「9」、横幅が極端に短く描画されると「1」により近い数字として認識されてしまうと考えられる。

表 2 各被験者の数字の認識精度

O	被験者 1									被験者 2									被験者 3									被験者 4									被験者 5									認識精度				
	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9					
1																																																		96%
2																																																		94%
3																																																		90%
4																																																		72%
5																																																		78%
6																																																		96%
7																																																		70%
8																																																		96%
9																																																		88%

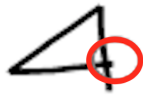


図 10 「9」と誤認識される「4」の例



図 11 「6」や「3」と誤認識される「5」の例



図 12 「9」や「1」と誤認識される「7」の例

表 3 「○」と「/」のみによる採点作業の実験結果

	提案システム非使用時			使用時
	採点作業 (sec)	集計作業 (sec)	総所要時間 (sec)	所要時間 (sec)
被験者 1	258	216	474	309
被験者 2	243	197	440	277
被験者 3	248	148	396	385
被験者 4	183	230	413	284
被験者 5	204	139	343	301

表 4 「○」、「/」および「△」による採点作業の実験結果

	提案システム非使用時			使用時
	採点作業 (sec)	集計作業 (sec)	総所要時間 (sec)	所要時間 (sec)
被験者 1	593	238	831	694
被験者 2	407	353	760	555
被験者 3	395	259	654	651
被験者 4	467	268	735	608
被験者 5	417	212	629	563

集計作業も含めた総所要時間でみると、提案システムを用いた時の方が 107.6 秒早く終わらせることができていたことがわかった。

総じて、提案システムを用いてない場合において、計算が得意で手際の良い被験者は他の被験者よりも採点作業自体を早く終わらせることができ、コンピュータの操作に慣れている被験者は他の被験者よりも集計作業を早く終わらせることができていたことがわかった。

5. 結論

本研究では、人がコンピュータを利用して行う作業を利便化するシステムの一例として、手書きで採点を行う採点を利便化するシステムを、手書き文字認識と音声フィードバックを用いた対話型訂正機構を用いて構築した。マークシート方式や Web 回答方式を対象とした採点システムは存在しているが、手書き回答方式が多くの場面や環境で主流となっていないながら、それを補助、利便化するシステムが存在していないことが大きな問題点としてあげられた。加えて、丸付け動作を行う採点は、ただ丸付け動作を行うだけでなく、採点者による得点計算や、結果を分析するためにデータ化する作業も含まれる。そうした一連の作業に時間と手間がかかり、それらを解決するシステムが存在しなかった。そこで、提案手法として紙の答案の採点業務を対象とした、採点システムを提案した。

提案システムでは、ユーザが行う丸付け作業を丸付けが行われるたびに認識し、ユーザに音声フィードバックを与え、認識結果の変更を行うことができる他、採点と同時に得点計算や正答のデータ化を自動で行うことができる。そ

4.4 実環境を想定したシステムの評価

実環境を想定した状況での提案システムの有効性を示すために評価実験を実施した。以降では、その結果について述べる。

4.4.1 「○」と「/」のみによる採点作業の評価

「○」と「/」のみによる採点作業の評価実験を行った結果を表 3 に示す。表 3 から、被験者 5 人のうち全員が提案システムを用いていない場合の方が採点作業自体は早く終わらせられることができていたことがわかる。しかし、集計作業も含めた総所要時間でみると、提案システムを用いた時の方が平均 102.0 秒早く終わらせることができていたことがわかる。

4.4.2 「○」、「/」および「△」による採点作業の評価

「○」、「/」および「△」による採点作業の評価実験を行った結果を表 4 に示す。表 4 から、被験者 5 人全員が提案システムを用いていない場合の方が採点作業自体は早く終わらせられることができていたことがわかる。しかし、

のシステムを特徴点抽出と CNN による画像認識を用いて python で実装し、その評価を行なった。

評価方法は、認識精度に加え、提案システムを利用した場合と、利用していない通常の手書きで行う丸付け作業にかかる所要時間を計測し、比較を行なった。本研究で実装して認識することができたのは「○」、「△」、「/」の3種と9種の数字で、各図形は高い精度で認識を行うことができた。しかし「○」と「△」およびワンストロークで描画を強制した数字に関しては、ユーザの書き方により認識率にわずかに差がでた。また、実験を行なった5人の被験者全員が、システムを利用した場合の方が採点・集計作業にかかる時間を大幅に短縮できており、ユーザにかける手間を減らしつつ採点業務を終えることができると証明された。

参考文献

- [1] 井上 創造, “実世界に広がる装着型センサを用いた行動センシングとその応用: 9. スマートフォン行動情報収集と看護行動センシング,” 情報処理, Vol. 54, No. 6, pp. 600–604, 2013.
- [2] 熊谷 卓也, “ジェスチャー認識装置を用いた人体位置検出と工程作業動作分析への応用,” コミュニカルテクノロジーレポート, Vol. 11, No. 6, pp. 42–47, 2014.
- [3] 越智洋司, 井手勝也, “選択肢認識機能を実装したタブレット型問題演習システム,” 教育システム情報学会誌, Vol. 32, No. 1, pp. 37–47, 2015.
- [4] 市村哲, 山下亮輔, 松本圭介, 中村亮太, 上林憲行, “紙答案と電子フィードバックを併用した講義支援システム,” 情報処理学会論文誌, Vol. 49, No. 1, pp. 525–533, 2008.
- [5] E. Rosten and T. Drummond, “Machine Learning for High-Speed Corner Detection,” in Proc. of the 9th European Conference on Computer Vision (ECCV’06), Vol. 1, pp. 430–443 (2006).