

デスクトップキャプチャ画像による アプリケーション操作演習の進捗状況解析の検討

占部 弘治 *1・福田 裕也 *1

Email: k.urabe@niihama-nct.ac.jp

*1: 新居浜工業高等専門学校電子制御工学科

◎ Key Words 演習進捗解析 ラーニングアナリティクス ディープラーニング

1 はじめに

ワードプロセッサや表計算ソフト・プレゼンテーションソフトなどを用いた文書・資料・発表スライドの作成は日常の業務に欠かせないものとなっている。

そのため、ICT 活用技術を学ぶ上でアプリケーション操作方法を習得する演習は重要なカリキュラムの一つとなっている。新居浜工業高等専門学校においても全学科の1年生で開設されている科目「情報リテラシー」において、セキュリティや情報モラルを学ぶとともに、ワードプロセッサや表計算ソフトの操作習得を目的とした演習を実施している。

この演習実施時において、受講者個別の進捗を把握できれば、進捗の遅れている受講者を発見し、これに対して適切な指導を行うことができると考える。また、受講者全体の進捗の傾向からは、演習の内容やテキストの改善の助けになると考える。

そこで本研究では演習時のPCのデスクトップ画像を収集し、これを解析することで、進捗の段階を判定するシステムを構築した。演習時のデスクトップ画像に表示される演習の状況は、画像のマッチングなどの画像処理による判定は難しいと考え、ディープラーニングを用いることにした。

以前の行ったスライド提示型授業における学生の受講態度を解析する研究⁽¹⁾でデスクトップ画像の解析を導入し、その報告を行っている。この解析においても画像処理の手法だけでなく、ディープラーニングを用いることでその精度が上がることを示しており、今回もディープラーニングの導入の効果が期待できると考える。

今回は、構築したシステムについて検証実験を行い、判定に用いるディープラーニングの比較検討を行った結果を報告する。

2 システムの構成

本研究で作成したシステムは図1に示す通り画面キャプチャアプリケーション・画像受信サーバ・進捗判定システムによって構成する。

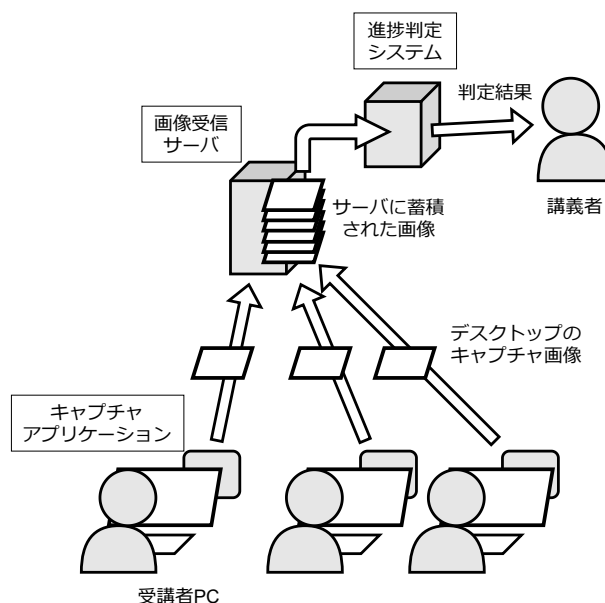


図1 システム構成図

画面キャプチャアプリケーションは Visual C# で作成した Windows アプリケーションである。これは受講者が使用するPCにインストールされ、一定の時間ごとにデスクトップ画像をキャプチャし、PCのIPアドレスとともに画像受信サーバへアップロードを行う。

画像受信サーバはキャプチャアプリケーションからアップロードされるデスクトップ画像を受信し、保存を行い、WebサーバApacheとその上で動作するPHPスクリプトで構成されている。画面キャプチャアプリケーションからアップロードされた画像にアップロードされた時刻とPCのIPアドレスを含んだファイル名を付け、サーバ内のストレージに保存する。これに

より、画像のファイル名からデスクトップ画像をキャプチャした PC と時刻の情報を得ることができるようにした。

進捗判定システムは得られた画像から進捗を判定する。判定には MATLAB の ディープネットワークデザイナー を用いて、事前学習済みネットワークに学習データを与えることで再学習し、これを用いる。事前学習済みネットワークには ResNet 101 (2), AlexNet(3), SqueezeNet(4) を採用し、比較検討を行った。

3 システムの検証実験

実際の演習課題を用いて、システムの検証を行った結果を示す。

3.1 演習課題

このシステムを実際に新居浜工業高等専門学校電子制御工学科 1 年の科目「情報リテラシー」で行っている演習課題について行った。

この課題は、この科目の教科書に掲載されている表とグラフを含んだ文書の見本「実験報告書の書き方」(5) を元に文書を作成するというものである。図 2 に作成例を示す。

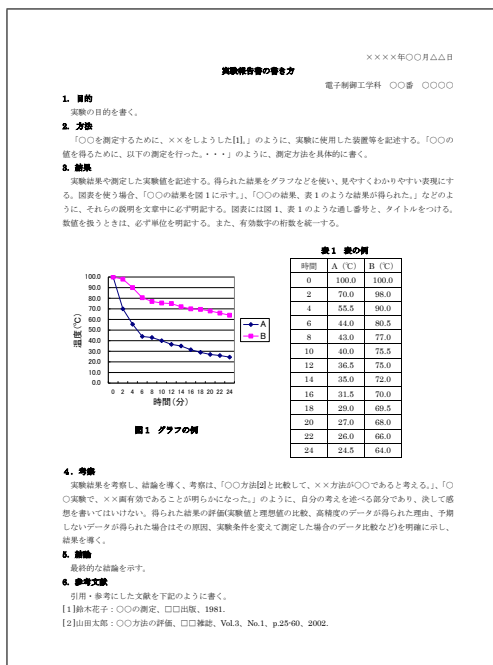


図 2 表とグラフを含んだ文書の作成演習の完成例

この文書の作成の進捗を次の 6 段階に分け、デスクトップ画像からどの段階かを判定させる。

1. 全てのセクションを入力

2. 「結果」の文書まで入力
3. 表計算ソフトで表を作成
4. グラフの作成が完了
5. 「結論」までの文章の入力
6. 参考文献までが入力 (完成)

なお、セクションの入力には見出し機能を使うため最初に入力するよう指示したが、それ以外の手順は特に指示は行わなかった。

3.2 キャプチャ画像の収集

今回の実験に用いるデスクトップ画像は実際の授業で収集したものではなく、学生 4 名に同じ演習を実施させ、これを収集した。このときのキャプチャの時間間隔は 2 秒とした。なお、収集されたデスクトップ画像はネットワークに与えるデータにする際にネットワークに合うように縮小している。

これらを学習データおよび評価データとするために得られた画像について目視の判定で 6 段階のいずれかに分類し、段階ごとに無作為に抽出した。学習及び評価に用いた画像の枚数を次の表に示す。

表 1 画像データ

演習の 段階	学習データ			合計
	訓練 データ	検証 データ	評価 データ	
1	480	120	200	800
2	480	120	200	800
3	480	120	200	800
4	480	120	200	800
5	480	120	200	800
6	480	120	200	800
計	2880	720	1200	4800

3.3 学習の結果

この画像データを今回の実験の判定に用いた事前学習済みネットワーク ResNet 101, AlexNet, SqueezeNet に訓練データを与え、再学習を行った。再学習後のネットワークを用いて学習データの判定結果からの正解数を表 2 に、正解率を表 3 に示す。

また、評価データの判定結果からの正解数を表 4、正解率を表 5 に示す。

表 2 学習データの正解数

段階	ResNet-101	AlexNet	SqueezeNet
1	567	572	572
2	585	555	535
3	560	556	553
4	552	554	553
5	594	591	586
6	600	585	600
合計	3458	3413	3399

(各段階のデータ数は 600, 合計で 3600)

表 3 学習データの正解率 (%)

段階	ResNet-101	AlexNet	SqueezeNet
1	94.5	95.3	95.3
2	97.5	92.5	89.2
3	93.3	92.7	92.2
4	92.0	92.3	92.2
5	99.0	98.5	97.7
6	100.	97.5	100.
全体	96.1	94.8	94.4

表 4 評価データの正解数

段階	ResNet-101	AlexNet	SqueezeNet
1	200	194	200
2	191	121	184
3	195	172	163
4	184	180	165
5	128	65	114
6	152	144	165
合計	1050	876	991

(各段階のデータ数は 200, 合計で 1200)

表 5 評価データの正解率 (%)

段階	ResNet-101	AlexNet	SqueezeNet
1	100.	97.0	100.
2	95.5	60.5	92.0
3	97.5	86.0	81.5
4	92.0	90.0	82.5
5	64.0	32.5	57.0
6	76.0	72.0	82.5
全体	87.5	73.0	82.6

以上の結果から、学習データおよび評価データに対する正解率が最も高かったモデルは ResNet-101 であった。また、評価データに対して 5 段階と 6 段階における正解率が低くなっていた。

3.4 進捗判定の結果

再学習後のシステムに対して、時系列でデスクトップ画像データを与え、演習開始からの時刻に対する進捗の段階を判定した例を図 3 に示す (図中の青色のグラフ)。ネットワークは再学習後の正解率が最も高かった ResNet-101 を採用した。また、時系列の画像データは学習用に採取した学生のデータである。

段階が変わる前後において、判定の切り替わりが多く発生している。これは段階が変わる前後に判定が難しい画像が多く、そのため誤判定が生じていると考えた。

これは雑音に似た現象と考え、解消するために移動平均を用いることで進捗状況の表現を行った。これも図 3 に合わせて示している (図中の赤色のグラフ)。この移動平均の区間数は 20 である。

4 まとめ

本研究では、学生のアプリケーション操作演習における進捗状況を判定するために、演習中のデスクトップ画像を収集するシステムについて提案した。また、収集されたデスクトップ画像より進捗状況を判定するために学習済みネットワークの再学習を行い、これを用いることを試みた。学習済みネットワークには ResNet-101, AlexNet, SqueezeNet を用いて再学習を行い、これを用いて学習データおよび評価データにおける正解率を調べたところ、ResNet-101 によってもっとも良い結果が得られた。

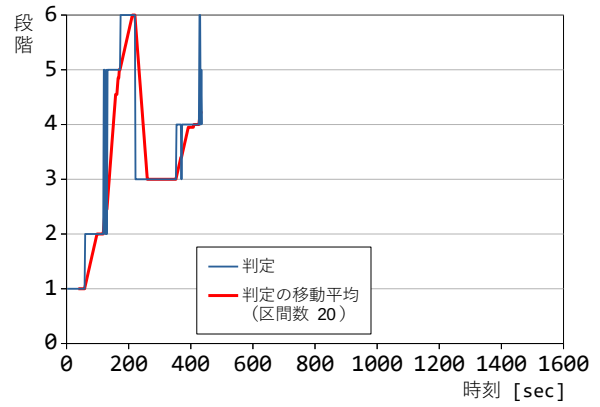
しかしながら、演習を行っている画像データに対して判定を行い、進捗状況の解析を行ったところ、段階が移り変わる前後での誤判定が多く現れた。これに対しては、移動平均を用いることで進捗を表現する方法を提案した。

学習に用いた学習データおよび評価データの数を増やすことでより精度の良い結果が得られると予想する。また、デスクトップ画像全体でなく、対象とするアプリケーションの画像のみを取得し、これを用いて学習や判別に用いることで改善できるのではないかとない

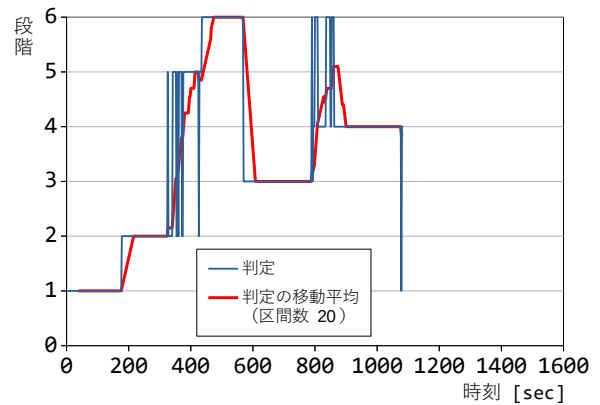
かと予想している。これらについて、今後の研究を進めていきたい。また、実際の授業での演習で用いた場合についても調査したいと考えている。

参考文献

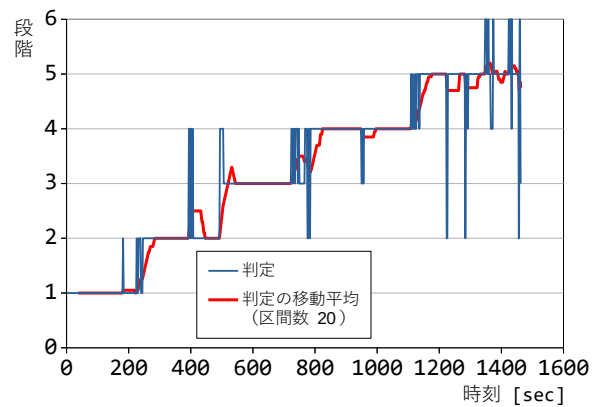
- (1) 山内拓磨, 占部弘治, “スライド提示型授業受講状況収集システムにおけるデスクトップ画像解析”, PCカンファレンス 2017 論文集, p.237-239, 2017.
- (2) K.He et al., “Deep residual learning for image recognition”, IEEE CVPR 2016, p.770-778.
- (3) Alex Krizhevsky et al., “ImageNet Classification with Deep Convolutinal Neural Networks”, Neural Information Processing System, p.1087-1105, 2012.
- (4) F.N. Iandola, et al., “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5 MB model size”, arXiv:1602.07360, 2016.
- (5) 岡田正, 高橋参吉, 藤原正敏, “ネットワーク社会における情報の活用と技術 三訂版”, p.36, 実教出版, 2010.



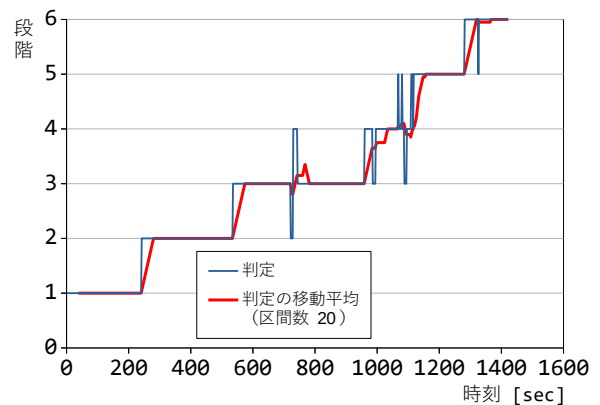
(a) 学生 1 の進捗状況



(b) 学生 2 の進捗状況



(c) 学生 3 の進捗状況



(d) 学生 4 の進捗状況

図 3 ResNet-101 による進捗状況判定