

書道自動添削スマートフォンアプリケーションの開発

奈良工業高等専門学校 電子制御工学科 ○西岡亜美 ◎西田茂生

要旨

書道を習う人は世界中で増加している。近くに指導者がいない人に便利な添削システムはあるが、現状ではほぼオンラインである。本研究ではスマートフォンを用いて、既存の添削システムとは異なる、指導者が朱書きで添削するようなオフライン添削システムの開発を行う。本稿では Android 上で動作するアプリ開発を試みた。開発したシステムをスマートフォン上で動作させ、あたかも指導者が添削するようなアプリとして動作することを確認した。

1. はじめに

現在、書道に取り組む人は世界中で増加している。しかし、海外で書道に取り組もうと思っても、書道を教える指導者がいなければ上達は困難である。既存の添削システムは、手本を臨書したものを指導者に送り添削を受けるか、書いた作品をオンラインで指導者に送り添削を受ける手法が一般的である¹⁾。いずれにしても海外では金銭的、もしくは時間的余裕が足りず、添削を受けることは厳しい。

これらを解消するため、本研究では時間と場所を選ばない、スマートフォンを利用した簡易添削システムの開発を行う。入力した画像から半紙の輪郭を抽出し、それをもとに台形補正を行う。補正した入力画像は、お手本画像のみを赤色に色変換処理を行う。書道作品をお手本に重ね合わせるにより、作品を添削するスマートフォンアプリを製作する。これにより、あたかも先生が生徒の作品を朱書きで添削するようなアプリとなる。さらに開発した添削システムの検証を実機を用いて行う。

先行研究にて、python のシミュレータ上で台形補正から重ね合わせまでの一連の流れを行い、入力画像とお手本画像を重ね合わせることが可能であると検証されている。また、Android Studio エミュレータ上で1枚の画像を入力し、手動で座標を設定することで台形補正を行い出力できることを確認している。

そこで本稿では以下の3点を追加し改良を行った。まず、様々な種類のスマートフォンでアプリケーションが使用できるよう、座標を自動で検出できるようにプログラムの改良を行った。次に、お手本画像のみを赤色に色変換処理を行い、重ね合わせるプログラムを製作した。最後に実機を用いて開発した添削システムの検証を行った。

2. 研究の方法

2.1 添削システムの条件

添削対象となる書道作品について以下の条件を指定する。

- (1) 撮影背景は一般的な書道用下敷きを想定し、黒とする。
- (2) 一般的な半紙サイズとする (243×333mm)。しかし、手本作品は実際の半紙サイズと同じ縦横比であれば大小を問わない。

2.2 添削手法

Table1 に添削システムの開発環境を示す。この開発環境の下で Android 用の書道添削アプリを開発する。

Table1 添削システムの開発環境

開発環境	Android Studio 4.0.1
対応するバージョン	Android 8.0~10.0+
エミュレータ	Nexus 5X API 29
	Pixel 3 XL API 29
	Pixel 2 API 30
実機	LENOVO Lenovo TB-7305F
画像処理ソフトウェア	OpenCV 3.4.11

2.3 添削の手順

添削の手順を Fig.1 に示す。実際の添削と同様に作品の上にお手本を朱書きで重ねる手法を採用する。入力した画像のうち、添削対象となる画像をお手本画像と同じサイズになるよう

にサイズを変更し、半紙の輪郭を抽出した後、それをもとに台形補正を行う。補正した画像は半紙のしわや汚れといったノイズを除去し、お手本画像と重ね合わせ、作品としての文字のバランスを添削する。

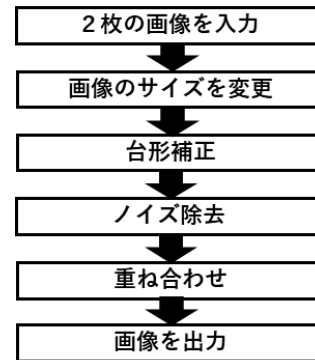


Fig.1 修正手順

2.3 台形補正

書道作品をカメラで撮影する際、カメラと撮影対象の間の角度から遠近による歪が発生し、対象が台形に見えてしまう。これを解消するために、撮影した画像から半紙のみを切り出し、抽出した半紙の頂点座標を長方形になるように座標変換させる射影変換を行う。

台形補正の処理手順を Fig.2 に示す。台形補正では、まず入力した画像にグレースケール変換、二値化を行い OpenCV で定義されるメソッドである findContours 関数を使用できる状態にする。次に findContours 関数を用いて輪郭抽出を行い、輪郭を単純な直線に近似するため面積計算、輪郭近似を行う。最後に変換後の座標を検出し射影変換を行うことで、台形補正された画像を取得できる。なお、この処理は先行研究で開発されていたプログラムにおいて手動で座標指定していたものを自動で検出できるプログラムに改良し、添削システムへの組み込みを行った。

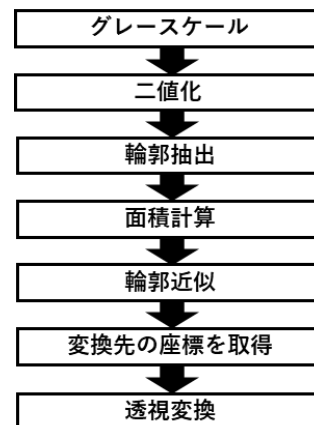


Fig.2 台形補正の手順

2.4 ノイズ除去

ノイズ除去の処理手順を Fig.3 に示す。ノイズ除去では台形補正までの処理を行った元画像と輪郭抽出、輪郭描画で得られ

たノイズのみの画像を作成し、それらの重ね合わせを OpenCV によって定義される copyTo 関数を用いて行う。また、台形補正までの処理を行った元画像のみを赤色へ色変換処理を行っておくことで、重ね合わせたときに文字部分のみが赤色になるように調節する。そして、OpenCV によって定義される inRange 関数を用い赤色部分以外をマスクングすることによって、ノイズ除去を行ったマスク画像を作成する。

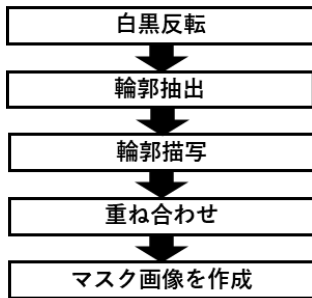


Fig.3 ノイズ除去の手順

3. 研究の結果とその意味

3.1 様々なエミュレータおよび実機による検証

世界中で扱える書道添削スマートフォンアプリケーションとしては、本来は iOS なども含めたクロスプラットフォーム環境での開発が望ましい。しかし開発難度の観点から、今回は Android Studio を使用した Android 上でのみのアプリ開発を行う。

また、世界中の Android の普及率は非常に高く、開発を行っているメーカーは多岐にわたり種類も多い、機器の種類によって画面サイズ等も大きく変わり、1つのエミュレータに合わせて開発したアプリケーションが他のエミュレータでは正常に作用しないこともある。そのため、本節では3種類のエミュレータと1種類の実機を用いて、開発したスマートフォンアプリケーションの検証を行う。

作成した添削アプリの出力画像を Fig.4, Fig.5 に示す。また、実機画面のスクリーンショットはエミュレータの画像と見分けがしづらいため、実機本体ごと撮った写真を Fig.6 に示す。

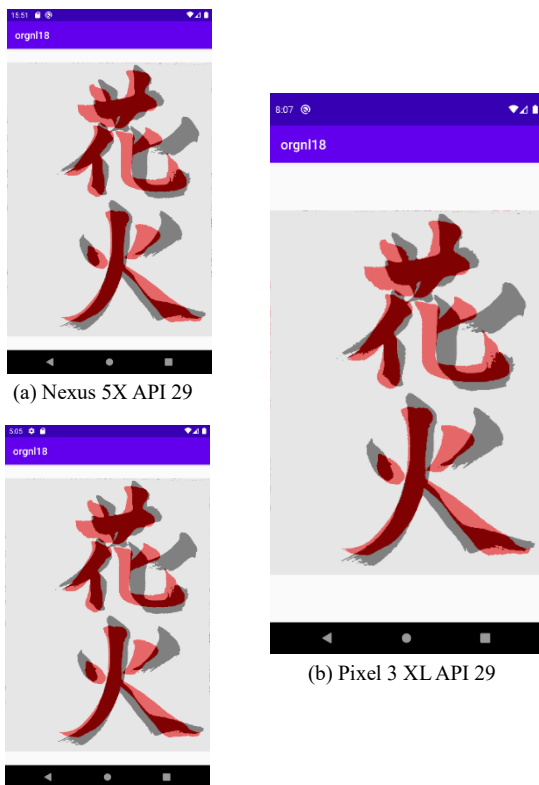


Fig.6 Pixel 2 API 30

Fig.4 エミュレータの出力画像



Fig.5 スクリーンショット



Fig.6 実機本体

図より機器の種類や画面サイズ等に関わらず正しい添削結果が出力されていることがわかる。これにより、Android スマートフォンまたはタブレットを所持していれば、国内外問わず書道自動添削スマートフォンアプリケーションを使用できることがわかる。

3.2 アプリケーションの改善点

作成した添削アプリの出力画像とその一部の拡大図を Fig.7 に示す。

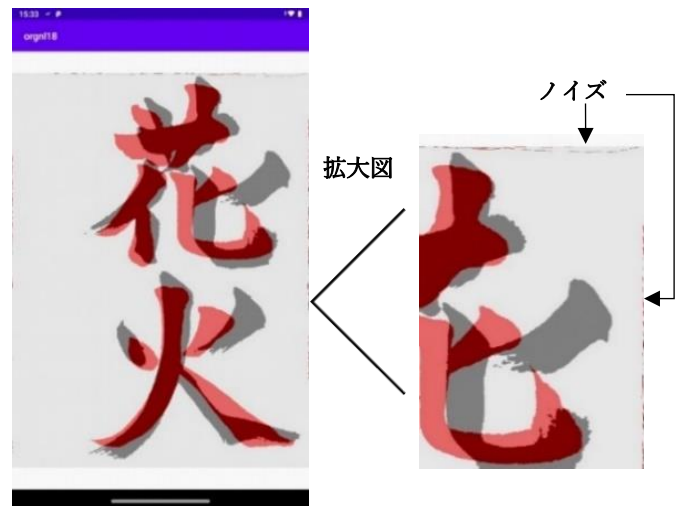


Fig.7 評価結果

図より入力した画像に対して正しく台形補正、重ね合わせができていることが確認できる。しかし、半紙の縁に生じるノイズは完全には除去できていない。

4. おわりに

Android 上で動作する書道自動添削スマートフォンアプリケーションを開発し、実機を用いた検証を行った。その結果、画像を台形補正し重ね合わせることであたかも指導者が朱書きで生徒の作品を添削するようなアプリケーションであることが示された。また、ノイズ除去の精度を上げることでアプリとしての完成度が高まると考えられる。

参考文献

- 1) 書道 (習字) を愉しむお手本販売サイト <https://www.o-tehon.com/wp/syoho-guide>, 2021/1/19