

デザイン修正案の現実環境における リアルタイムプレビューを可能にする AR インタフェース

岩谷 明^{1,a)} 高橋 伸² 田中 二郎²

概要: ポスターなど印刷物のデザイン制作では、コンピュータ上に動作する DTP ソフトウェア及び液晶ディスプレイを用いて作業を行う。そのため、印刷した際の見え方とディスプレイ上での見え方が異なって見える、“印象の不一致問題”が発生する。本研究では拡張現実 (AR) の技術を用いてこの問題の軽減を試みる。すなわち、実際に印刷し掲示した状態のポスターに対してその場でデザイン要素の修正のプレビューを行うことを目的に、デザイン修正のための AR インタフェースの考案・設計を行い、現実を上書き修正をプレビューできるシステムを開発する。本稿ではフォントの修正に特化したプロトタイプシステムについて詳述する。

An AR Interface to Enable Real-time Preview Design Variations in Actual Environment.

IWAYA AKIRA^{1,a)} TAKAHASHI SHIN² TANAKA JIRO²

Abstract: To create graphics designs of printed materials such as posters, nowadays we use DTP softwares on computers and liquid displays. Because of that, an problem called "impression inconsistency" which is a difference between an impression on a liquid displays and one on a printed materials occurs. In this study, we attempt to resolve this problem by using technologies of augmented reality. Namely, we devise AR interfaces to modify designs to make a system to preview modifications of printed posters put up in an actual environment by overwriting real world and devise AR interfaces to modify designs. In this paper, we describe a prototype system to modify design elements, especially fonts in detail.

1. はじめに

Desktop Publishing(DTP) が、コンピュータの進化に伴って普及した結果、今ではポスターやフライヤーなどの印刷物のデザインには Adobe Illustrator などの DTP ソフトウェアを用いることが標準となった。

DTP ソフトウェアでは、ディスプレイに表示される内容を、印刷結果と一致するようにユーザーに提示する WYSIWYG(What You See Is What You Get; ウィジウィグ) と

いうユーザーインタフェースが用いられる。これにより、ユーザーは円などの図形や、写真、フォントを設定した文字などを直観的に配置することができ、またその見たとおりのものを印刷結果として得ることができる。

しかし、A4〜ポスターサイズの印刷物をデザインする際は、1つの問題が発生する。ディスプレイサイズで良く見えるデザインと、ポスターサイズで良く見えるデザインとは必ずしも一致しないという、“印象の不一致”である。

印象の不一致にはいくつかの理由が考えられる。まず第1に、素材の違い、解像度の違い、大きさの違いなどの物理的な理由である。物理的な問題の根本的な解決は難しい。しかし、ディスプレイ上での色味と印刷した色味を一致させるカラーキャリブレーション技術はプロユースのディスプレイに搭載されており、最近では 72dpi 以上の高解像度ディスプレイも登場し始めるなど、ハードウェアの進化に

¹ 筑波大学 大学院システム情報工学研究科 コンピュータサイエンス専攻

Department of Computer Science, Graduate School of Systems and Information Engineering, University of Tsukuba

² 筑波大学 システム情報系

Faculty of Engineering, Information and Systems, University of Tsukuba

a) iwaya@iplab.cs.tsukuba.ac.jp

よって対処できる部分も少なくない。

第2の理由として、印刷物を掲示する環境に影響されることが挙げられる。具体的には、屋内か屋外か、照明や環境光はどのようなものか、ポスターを貼る壁などの周辺の物体はどのようなものか、などである。

光の色が異なれば、DTPソフトウェア上でデザインした時と見かけの色は異なってくるし、また背景が何色なのか、どのような素材なのかでも印象が異なって見える。

これらの問題の結果、ディスプレイ上ではよく見えたフォントが印刷して掲示するとあまりインパクトなく見えたり、良いと思った配色が実際の環境の中で目立たなくなってしまうりする。これが印象の不一致である。この第2の問題は、実際の環境に設置してみないと雰囲気がわかりにくい、という点でハードウェアの進化では吸収しにくく、異なったアプローチが必要と考えられる。

このように WYSIWYG にも限界はあり、現状ではデザイナーは印刷・確認・修正のプロセスを何度も繰り返して、この問題に対処している。

2. 目的と手法

本研究の最終的な目的は、この印象の不一致問題を軽減することである。実際の掲示場所の大きさ・環境でデザイン制作を行うことができれば、不一致の差を大幅に少なくすることができると考えられるが、コンピュータに向かい DTP ソフトウェアを用いて作業する以上、現実には難しい。そこで、デザインを DTP ソフトウェアで制作・印刷し、実際の環境に掲示した後にその場でデザインの修正をプレビューできるようにすることを目的とする。そのために本研究では手法として AR（拡張現実）を採用した。AR（Augmented Reality; 拡張現実）とは、人間が知覚する現実世界及びそこから得られる情報をコンピュータにより拡張する技術、またはその拡張された環境を示す言葉である。近年では、スマートフォンなどのカメラを通して得られる動画像にリアルタイムで処理を施して、付加的な情報をオーバーレイし表示することを総称して AR と呼ぶケースが多い。現実世界の物体に対し、リアルタイムで情報を付加あるいは既存の情報を変化させるという AR の手法は、ユーザーの思うままに、すでに印刷され掲示されたデザインを、周りの環境を体感しながら改変することを可能にすると考えられる。これによりユーザーは、今までの印刷・確認・修正の繰り返しを大幅に減じることが可能になると想定される。本研究では AR を用いて実際の環境に掲示された印刷物のデザイン修正を行うシステムを提案、また設計、実装、評価を行う。すなわち、現実の環境に掲示されたポスター上のフォントや配置、色などのデザイン要素を上書きし、修正結果をプレビューできるシステムである。図1に現実の環境に提示されたポスターの例を、図2に AR により修正した様子を示す。情報の付加のみなら



図1 現実の環境に掲示されたポスター

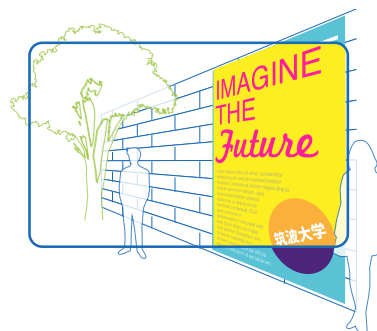


図2 AR によりデザインの修正を加えプレビューした画面
ず現実世界の情報を、ユーザーが選択した別の情報にて上書きして修正するための AR 技術及びインタフェースを考察・設計することは、AR の応用範囲を広げ、現実世界とコンピュータのよりシームレスな連携に寄与できると考えられる。

2.1 AR に用いるデバイス

AR に用いるデバイスとして、iPhone などのスマートフォンと、iPad などのタブレット端末、Oculus Rift などのヘッドマウントディスプレイ (HMD)、Google Glass などの Glass 型端末を考えている。スマートフォン／タブレットの利点としては大画面であること、複数人で見られることが、欠点としては、アプリケーションを起動し、対象物に向けるという手間が挙げられる。HMD の利点としては、没入感が得られる点が、欠点としては、その重量と外が全く見えない点が挙げられる。グラス型端末の利点としては、常に装着している想定なので、対象物を見るだけで AR が表示される点が、欠点としては、タッチインタフェースの領域が小さい、音声入力では複雑な操作をしづらい点が挙げられる。本研究では、これらのデバイスで実装を行い、どちらの使い勝手が良いか、デザイン編集という観点で各々に最適なインタフェースはどのようなものか、を合わせて検討する予定である。

3. デザイン修正 AR インタフェース

本研究ではまず、ポスター等の印刷物において最も頻出する文章のデザインを対象に研究を行った。すなわち、印刷された文章の字体を色やサイズを保ったまま異なるフォ

ントにて上書きし修正する AR システムを開発した。

3.1 AR インタフェースの利用例

AR のデバイスとしてスマートフォンを用いた場合の、AR システムの利用の流れを以下に示す。ターゲットユーザーは印刷物のデザインを行うデザイナーであり、想定する利用シーンは、デザインのプロセスにおける印刷物を確認する部分である。通常は前述のとおり、印刷した結果が意図した通りの仕上がりでなかった場合、再度コンピュータに向き合い、変更を加え、再度印刷し確認する。しかし本研究の AR システムを用いることにより、印刷結果の編集を AR により即座に行うことができる。まずユーザーはポスターを DTP ソフトウェアで制作し、印刷、実際の環境に掲示して確認する。ここで「想定したよりインパクトに欠ける」などの印象の不一致が発生し、ユーザーはフォントを変更してみてもどうか、と考える。そこでユーザーは本研究の AR システムを起動し、スマートフォンをポスターにかざして、変更した部分をタップにより選択する（図 3 左）。元々の文字が新しいフォントで上書きされるので、スワイプにより選択する（図 3 右）。



図 3 AR によりデザインの修正を加えプレビューした画面

これにより実際の見え方でデザイン修正のプレビューを行うことができる（図 4）。したがって印象の不一致を解消し、印刷・確認・修正のサイクルを削減することができる。ユーザーはプレビューで確認後、DTP ソフトウェアで修正・印刷・掲示する（図 5）。また、自分で作成したデザインに限らず、日常で見かけたポスターなどの印刷物に対して「ここを変更したらどうなるのか」を即座に試せるため、デザインの学習においても役立つと考えられる。



図 4 修正のプレビュー

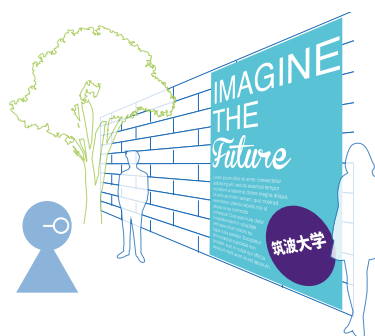


図 5 完成図

4. プロトタイプシステム

スマートフォン (iPhone 6, 4.7inch, 1334x750, 326dpi, iOS 8.1) 上で動作するプロトタイプの AR システムを、Mac OS X 10.10 の Xcode 6.1, iOS SDK 8 で開発している。また、画像処理には OpenCV を用いている。実際の動作の様子を図 6, 図 7 に示す。

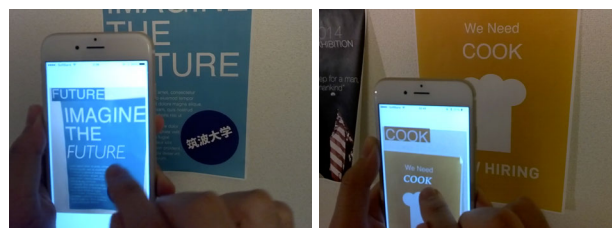


図 6 実際の動作の様子



図 7 スワイプによるフォントの選択

図 6 左では、“FUTURE”という単語を、図 6 右では、“COOK”という単語を選択し、フォントを変更したプレビューを確認している。また図 7 ではそれぞれにおいて、フォントをスワイプ操作により変更している。

4.1 処理の流れ

前述した利用シーンに必要な機能を達成するための処理の流れを以下に示す。まず、カメラの映像を取得し、文章領域の検出処理を行う。あるいは、ユーザーがドラッグなどの操作により対象とする文章領域を指定する。文章領域の検出処理を行った場合は、その中からどの文章を対象とするのかを指定する。そして得られた文章領域に対して、文字認識を行い、認識結果の文字列を保存する。その後、文字周辺の背景領域の色やテクスチャにより文字領域に本

来存在するはずの背景を再構築し、文字領域を除去したものを背景画像として保存する。先ほどの文章領域をトラッキングし、背景画像及び、認識した文字を様々なフォントでレンダリングしたものを重ねて表示する（図 8）。

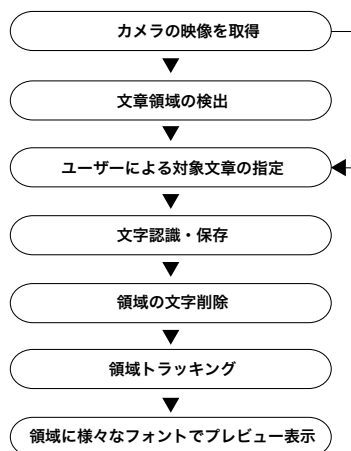


図 8 プロトタイプシステムの処理の流れ

カメラ入力

iPhone に搭載されているカメラからリアルタイムに動画を取得する。iOS SDK に含まれるフレームワークである AVFoundation を用いて実装を行った。

ユーザーのタッチ入力による領域選択

ユーザーが指のドラッグ操作により修正の対象とする文章領域を矩形領域（修正対象領域）として指定する。修正対象領域は半透明の白色でユーザーの指の移動に合わせて塗りつぶされるため、ユーザーはどの領域を指定しているかを確認することができる。ユーザーが指を離すと修正対象領域は確定する。

文字認識エンジン Tesseract[1] による文字認識

修正対象領域を文字認識エンジンの Tesseract に入力し、文字認識を行う。本システムは iOS 上で動作するため、Tesseract-OCR-iOS[2] を用いて実装を行った。

Inpaint[3] による元々の領域の文字の除去

修正対象領域を Inpaint により塗りつぶし、文字領域を本来存在するはずの背景にて上書きし除去する。Inpaint は傷のついた画像を修復するために用いられるアルゴリズムである。傷領域をマスク画像として入力することにより、傷領域周辺の情報を用いて傷領域を埋める。本システムではマスク画像として修正対象領域を Inpaint に入力することにより、文字領域が背景で塗りつぶされた画像（背景復元画像）を得ている。

領域中の背景部分を削除しての文字色の抽出

元々の領域の文字を除去する一方で、元々の文字色の抽出を行う。修正対象領域の四隅に存在する色で、かつ出現頻度が最も高い色を背景色とし、その色を修正対象領域から消去し残った色を元々の文字色として

いる。

Template Matching による領域トラッキング

文字除去前の修正対象領域を Template Matching により領域トラッキングし、その位置に背景復元画像と後述するフォントプレビューを上書きすることにより、ユーザーがカメラを動かした場合にも追従するようにしている。

様々なフォントでのプレビュー表示

文字認識により得られた文字列を、iOS にインストールされているフォント及び修正対象領域の高さを元に計算した大きさに描画し、修正対象領域にオーバーレイして表示する。ユーザーは修正対象領域をスワイプすることにより、様々なフォントに切り替えてプレビューを確認することができる。

5. 関連研究

Quest Visual, Inc. による “Word Lens”^{*1} は、看板などの文章をスマートフォンのカメラにかざすと、機械翻訳し、元々の文章を翻訳済みの文章に置き換えて表示する AR アプリケーションである。Word Lens では、元の文章の色、傾き・歪みはそのままに、あたかも、最初から翻訳済みの文章が描かれていたように、元の文章を消して上書きし表示する。Word Lens が自動でカメラに写った文章をすべて翻訳するのにに対し、本研究ではユーザーはどの部分の文章を変更するか、どのような字体に変更するかを選択できる、すなわちユーザーの介在がある AR アプリケーションである。デザイン修正というタスクを遂行できる AR アプリケーションのインタフェース・インタラクションの考案・設計は本研究におけるメインテーマであり、追究したい。

ViewAR GmbH による “ViewAR”^{*2} は、部屋に飾る絵画を選択し AR でシミュレーションを行うアプリケーションである。ユーザーは実際の絵画を置く位置に、マーカー（特徴的なパターンの画像）を貼り、どの位置に絵画をオーバーレイするかを指定し、複数の絵画からどの絵画を貼るかを選択する。本研究はマーカーを用いないマーカーレス AR であり、また現実すでに存在しているポスターを上書きする点が、ViewAR とは異なる。

Agusanto らは、AR で 3 次元モデルを現実の空間に配置する際に、環境光を用いて写実的にレンダリングする手法 [4] を研究している。本研究でも、印象の不一致を軽減するには修正したデザインを元の環境と違和感ないように合成する必要があるため、関連している。

Lee らは、プロダクトデザインを対象とし CNC により作成したラフなモックに手などの遮蔽を考慮した上で AR により色付けを行い、現実においてデザインした製品をプレビューする手法を研究している [5]。本研究では対象が

^{*1} Word Lens, 入手先 (<http://questvisual.com>) (2014.11.24).

^{*2} ViewAR, 入手先 (<http://www.viewar.com>) (2014.11.24).

印刷物であり、プレビューのみならず修正を行うため、その点においては異なるが AR を用い実際の環境でデザインを確認するという点において一致している。

Kawai らは、ユーザーが指定した領域に含まれる物体をリアルタイムで除去（周辺の背景領域を用いて上書きする）隠消現実感 (Diminished Reality) について研究している [6]。本研究においても、ユーザーの指定した文字領域を背景を考慮して削除する処理を Inpaint にて行っている。

6. まとめと今後の予定

本研究では、印刷物のデザイン制作における、ディスプレイで編集時の見え方と印刷し掲示したものの見え方から生じる、デザイン印象の不一致問題を軽減するため、実際の環境で修正をプレビューできる AR アプリケーションを提案する。そして、その過程で、ユーザーによる修正というインタラクションが存在する AR アプリケーションではどのようなインタフェースが良いのかを追究する。今後は、スマートフォン版プロトタイプの実装を完成させる。現状では Template Matching にて領域のトラッキングを行っているが、3次元空間での領域トラッキングを行い、どのような角度からも適切に見えるよう修正する。また、ユーザーの入力領域をそのまま Inpaint に入力しているため、文字部分以外の、領域内の背景も塗りつぶされてしまう。そこで、文字領域のみの検出を行うことでこの問題を改善する。そして、他の端末においても実環境でのデザイン修正という観点からインタフェースを設計・実装し、各端末で評価実験を行う。

参考文献

- [1] Tesseract, 入手先 (<https://code.google.com/p/tesseract-ocr/>) (2014.11.24).
- [2] Tesseract-OCR-iOS, 入手先 (<https://github.com/gali8/Tesseract-OCR-iOS>) (2014.12.1)
- [3] M. Bertalmio, A. L. Bertozzi, G. Sapiro, “ Navier-stokes, Fluid Dynamics, and Image and Video Inpainting ”, Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol.I, pp.I355-I362, 2001.
- [4] K. Agusanto, L. Li, Z. Chuangui, N. Wan Sing : “ Photo-realistic rendering for augmented reality using environment illumination ”, Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality, pp. 208-216.
- [5] W. Lee, J. Park : “ Augmented Foam: A Tangible Augmented Reality for Product Design ”, Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality, pp.106-109.
- [6] N. Kawai, T. Sato, and N. Yokoya : “ Diminished Reality Considering Background Structures ”, Proceedings of the 2013 IEEE International Symposium on Mixed and Augmented Reality, pp. 259-260, Oct. 2013.