

配置と時間による可読性を維持した画像内文字提示手法

小川 直記 岡部 誠 尾内 理紀夫 益子 宗 平野 廣美*

概要. 広告画像等, 文字情報の入った画像が表示媒体の違いや, ブラウザの表示サイズの変更等で縮小されて使われ, 画像内に書かれた文字が読めなくなってしまうことがある. この問題を解決するため, 現在 Web 上に多数存在する文字情報と画像情報が結合された画像を対象として, 画像内の文字情報の可読性を維持するための, 新しい画像表示手法を提案する. 我々の画像表示手法では, 画像製作者があらかじめ画像中の文字を切り出し, それぞれの文字に重要度を付ける. 画像が縮小された時, 重要度の低い文字を消去し, 重要度の高い文字を拡大して表示することで, 可読性を維持する. この時, 拡大された文字は画像枠内からはみ出すことを許し, 拡大された文字が互いに重ならないよう配置するための手法も提案する. ユーザスタディを行い, 提案手法が広告画像中の文字の可読性の維持に有効である事, また, 文字が拡大されたり多少はみ出したりしても, 広告画像の分かりやすさを損なわない事を確認した.

1 はじめに

現在 Web ページにおける広告画像では, 文字情報と画像情報が結合されているものが多数存在している. Yahoo ショッピング¹や楽天市場², Amazon³等の電子商取引サイトでは, 多くの広告画像を素人が制作している状況があり, 今後も文字情報と画像情報が結合している状態が続くと考えられる.

文字情報を分離するには, 広告画像を制作する店舗の人が HTML における alt 属性を用いて文字情報と画像情報を分離して保存する方法もあるが, 素人であるため直接文字情報を打ち込むことは殆どない. また, alt 属性を設定するとしても, 画像の代わりになる適切な文字情報を新たに設定しなくてはならないといった画像製作者への負担や, alt 属性は画像が表示されない時や文字が読めない人へ対しての読み上げ用の代替文字情報であるため, フォントのデザインや画像そのものに写っている商品の画像情報がなくなってしまう. そのため, Web ページにおける広告画像では, 見た目を重視した画像の上に文字が挿入されている広告画像 (図 1(a) に例を示す) が多く使用され (図 2), 文字情報と画像情報の分離がなされていない. また, 近年では, レスポンシブデザインといった複数の CSS を用いて, 様々な画面サイズに対応することが普及してきているが, 画像が縮小されると文字が読めなくなる問題は残り, 多様な画面サイズに対応するために, CSS の知識を修得

Copyright is held by the author(s).

* Naoki Ogawa, 電気通信大学, Makoto Okabe, 電気通信大学/JST CREST, Rikio Onai, 電気通信大学, Soh Masuko, 楽天株式会社 楽天技術研究所, Hiromi Hirano, 楽天株式会社 楽天技術研究所/電気通信大学

¹ <http://shopping.yahoo.co.jp/>

² <http://www.rakuten.co.jp/>

³ <http://www.amazon.co.jp/>



図 1. 提案手法のイメージ: (a) 元画像, (b) そのまま縮小, (c) 提案手法



図 2. 実際に商品画像が並べられている例

する必要があり, CSS を複数用意する手間もかかるため, 素人が容易に扱うには敷居が高い.

そこで本論文では, 文字情報と画像情報が分離されていない画像を対象とした新しい画像提示手法を提案する. 従来の表示では, 図 2 のように広告画像を並べ, 1 画面に複数表示した場合は, スマートフォンなどの小さい画面で表示するときは画像を縮小して表示することにより, 画像中の重要な文字が読めなくなる (図 1(b)). そこで本提案では文字が画像の枠をはみ出して表示することを許し, 文字を大きく見やすく表示する (1(c)).

我々は画像内の文字の可読性を維持するために, 1

つの試みとして画像を縮小した際の文字の大きさに着目し、文字が読みやすい表示への変換を簡単なマウス入力で行うシステムと、作成した画像を視聴者に対して文字が読みやすい大きさに保ち、文字同士が重ならないようにする自動配置表示、及び、一定以上小さくなった際には文字を順番に拡大表示する時間拡張表示を行うことで、読みやすい文字情報の提示をする。また、重要でない文字は消すことで必要な情報を視聴者に見せるための領域を作る。

実際に広告画像の文字にシステムを用いて重要度を付けて表示を行い、可読性が維持されたことと、拡大した文字が画像内の商品の認識の妨げとならず、文字情報が画像情報を補うことで、画像情報、文字情報の双方を活用した表示手法であることを確認した。

2 関連研究

サムネイルを拡張して効果的に情報を提示する研究は存在する [9]。Woodruff らは、Web サイトを検索した際に、<H1> タグ内の文字サイズを指定の大きさにしたり、検索ワードを飛び出させ、元の画像に特定の色をオーバーレイすることで、文字を読みやすくし、人間の画像認識の速さと文字の情報を融合した拡張サムネイルを提案し、有意性を示した。他にも画像を表示する際に文字を見やすくする手法は存在する [12]。しかし、本研究のように文字を画像の矩形領域からはみ出すことを許したり、重要でない文字を消すことは行っていない。

文字が読める大きさの基準として最小可読文字サイズがある [5]。ここでは年齢、輝度、視距離や書体によって 80% 以上人間が読めるサイズを最小可読文字サイズとしている。印刷物を対象としているため、ソフトウェアやウェブサイトは対象外であるが、本研究では 1 つの基準として参考にする。また、最小可読文字サイズを基準として、書体の太さと視距離が見やすさにどう影響するか調べた研究もある [11]。

また、重要な部分を残し画像サイズを拡大、または縮小する研究は数多く存在する [1, 8, 6, 7]。Wang らは画像の重要度マップを作り、重要度が低い部分をメッシュ変形させることで、重要な部分を歪ませずに拡大縮小をした [10]。しかし、商品画像に適用すると、商品物体の領域が変形し、どのような商品が映っているのかわかりにくくなり、人間の画像認識が速いという利点が活かせなくなる。いずれにしても、静的な画像では表示可能な情報には限界がある。

そこで本研究では、画像を縮小した際に、画像中の文字が読みやすくなる表示方法と、簡単な操作で表示用のデータを作成可能とするシステムを目指す。

3 システム概要

本論文で提案するシステムは、画像製作者が使う文字指定部と視聴者が使う表示部がある (図 3)。



図 3. 画像製作から視聴者が見るまでの流れ

文字指定部では画像を入力とし、文字列候補の自動抽出を行い (図 3(b))、視聴者に見せたい重要な文字と消えてもいい文字をユーザがマウス入力で指定する。

表示部では、表示倍率に応じて文字を読みやすい大きさに保つため、自動配置表示と時間拡張表示の 2 種類を用いて表示する。自動配置表示では画像を縮小した際に、ユーザが指定した重要な文字を一定以上の大きさに保ち、極力重ならないように配置し、そうでない文字を消すことで必要な文字情報を視聴者に伝える表示である。時間拡張表示では、重要な文字列を順番に表示することで、より小さな画像になった時でも視聴者に文字情報を伝える表示である。さらに時間拡張表示では文字列の一部をより見せたい部分文字列として指定し、シンプルに情報を伝えることを可能とした。

4 ユーザインタフェース

ここでは画像製作者が使う文字指定部と視聴者側が使う表示部のユーザインタフェースについて述べる。

4.1 文字指定部

画像中の表示させたい重要な文字の決定は画像を製作するユーザが行う。ユーザが画像をシステムに入力すると、自動で文字列候補を切り出し、ユーザは各文字列候補に対して重要度を入力する。

文字列候補抽出

文字を挿入するとき、画像製作者は画像の上に文字を入れるため、画像情報と文字情報が分離されている場合も想定できる。この場合は抽出をする必要性はなく、画像製作の保存の際に画像情報と文字情報を切り分けることで文字列候補の抽出が出来る。しかし、画像製作に不慣れな人もおり、画像情報と文字情報が結合された画像が Web 上には多数存在している。そこで現状では手軽に使えるシステムとするため、製作の支援を行う必要があり、文字列候補の抽出を行う。また、文字列を矩形ではなく文字の輪郭に沿った領域を抽出することで、拡大しても下の

配置と時間による可読性を維持した画像内文字提示手法



図 4. 文字列候補抽出の流れ

画像情報を極力隠さないことを目指す。

文字列の抽出では商品に元から書かれている文字列ではなく、後から挿入された文字列を抽出するため、81% の精度で後から挿入された文字列を抽出した Hirano らの手法 [4] を基にして行う。本手法の文字領域抽出の精度は未検証であるため、既存手法との精度の比較は行っていないが、81% の精度で後から挿入された文字列を抽出している手法を基にすることで、文字領域を精度よく抽出できると考えられる。

1つの文字は1色で書かれていることが多く(図4(a))、同じ色の隣接しているピクセルをまとめることで、文字領域の輪郭を抽出出来ると期待できる(図4(c))。そこで Felzenszwalb らが提案した手法 [3] を用いて、色が似ているピクセルをまとめ、文字の輪郭を抽出する。

まとめた領域に対して、二値分類等に用いられる手法の一つであるグラフィカット [2] を用いて、文字領域か背景領域かに分類し、輪郭に沿った文字領域を抽出する(図4(d))。グラフィカットでは文字領域らしさと背景領域らしさを与える必要があるため、Hirano らの手法で文字列らしさ、まとめた領域のピクセル数を画像のピクセル数で割った値と隣接するまとめた領域の数を背景らしさとして与える。

得られた文字領域をまとめ、文字列候補とすることでユーザがシステムを扱いやすくする。文字領域をまとめる手法として、得られた文字領域を文字領域同士の外接方形の面積や重心のずれ等から文字列としてまとめていく平山らの手法 [13] によって文字列としてまとめ、文字列候補とする(図4(e))。

文字列指定 UI

ユーザは抽出した文字列候補に対し、クリックで視聴者に見せたい重要な文字列、消えてもいい文字列、文字列でない領域の3種類から重要度を指定す



図 5. 文字列の種類と部分文字列の指定インターフェース



図 6. (a) 抽出された文字列が分かれている例,(b) 分かれている文字列をまとめた例

る(図5)。また、時間拡張表示の際に「ダイエット成分が他社の30倍」という文字列があった時、「ダイエット成分」と「30倍」の部分文字列だけを見せればシンプルに視聴者に情報を伝えることが出来る。そこで、部分文字列をより見せたい文字列としてドラッグで指定可能とする。

また、抽出された文字列候補の中には図6(a)のように、1つの文字列が複数に分れてしまうことがある。この時、まとめた2つの領域を右クリックとドラッグで容易にまとめることが出来る(図6(b))。

4.2 表示部

本提案では表示倍率によって自動配置表示と時間軸を用いた時間拡張表示の2つに分かれる。最小可読文字サイズを用いるため、年齢、輝度値、視距離の入力部を備え、計算は読みやすいサイズに合わせるため、全て明朝体での最小可読文字サイズを元に算出した値を用いる。

4.3 自動配置表示

画像を縮小し、文字の大きさを一定以上に保つため、そのまま文字列だけを拡大すると、文字列同士が重なり、読みにくくなる(図7(a))。そこで、文字指定部で指定された消えてもいい文字を最小可読文字サイズ未満となった時に消去し、重要な文字を自動で再配置して極力衝突を回避する(図7(b))。

自動配置では次の4つの条件に基づいて文字列を繰り返し上下左右に動かして配置する。

1. 文字列領域同士の衝突を回避(図8(a))
2. 消えた文字列領域を活用(図8(b))

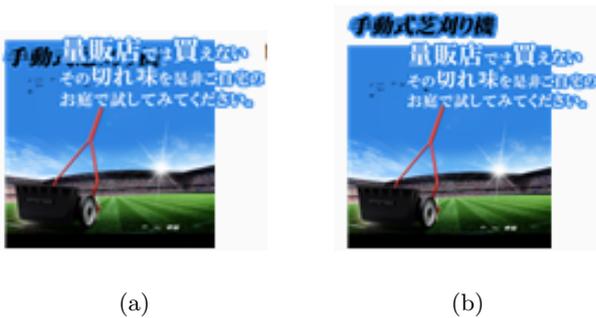


図 7. (a) その場で拡大すると文字列同士が重なり読みにくくなる, (b) 自動配置で文字列が重ならないようにし読みやすくする

3. そのまま縮小した際の位置とのずれを低減 (図 8(c))
4. 文字列同士の距離, 角度はそのまま縮小した時と同様の相対位置に近づける (図 8(d))

1 番目の条件は文字列同士が重なると読みにくいためであり, 2 番目の条件は文字列を消したことで空いた領域を活用し, 視聴者に見せるためである. 3 番目は文字列同士が衝突を回避することや, 消えた領域を活用することだけでは, 文字列が本来の画像と全く関係のない位置への配置が行われることがあるため, どの画像の文字列であるか視聴者がわからなくならないようにする条件である. 4 番目は例えば文字列 A が上, 文字列 B が下にあった時, 縮小した際に文字列 A が下, 文字列 B が上に逆転して配置されないようにするための条件である.

以上の 4 つの条件に基づき, 各文字列領域を繰り返し上下左右に移動することで, 出来るだけ条件を満たす配置を行う. また, 本研究では文字列領域を文字の輪郭に沿って抽出しているが, 計算時間を減らすためにこれらの条件の計算には文字列領域を囲む矩形で計算する.

文字領域を拡大, 移動すると消した文字領域を背景として表示しなくてはならないことが起きる. そこで, OpenCV⁴で領域の拡張を行えるモルフォロジー変換の一つである Dilate と, 周囲の情報から領域の色情報を補完する手法のインペイントで背景領域の情報を補完する. 抽出した文字領域は, 本来の文字領域より小さくなる事があり (図 9(a)), そのままインペイントを行うと, 消したはずの文字領域の情報を補完することになる (図 9(b)). そこでモルフォロジー変換で抽出した文字領域を拡張し (図 9(c)), インペイントを行うことで背景領域の情報のみで補完する (図 9(d)).

⁴ <http://opencv.org/>

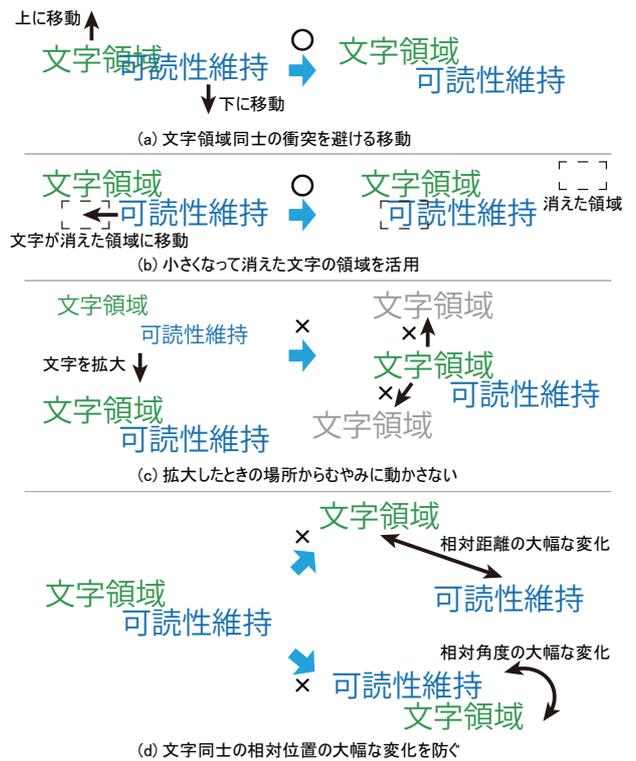


図 8. 自動配置条件のイメージ

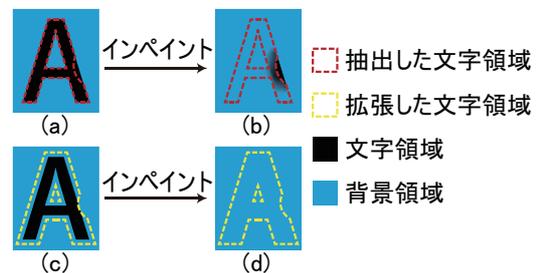


図 9. 文字領域の消去: (a) 実際の文字領域の一部を抽出失敗した場合, (b) 文字領域の一部が背景領域として残るとインペイントに文字領域の情報をうってしまう, (c) (a) の抽出した文字領域を拡張した場合, (d) 拡張した文字領域により背景領域の情報のみでインペイントを行う

4.4 時間拡張表示

画像を縮小していった時, 文字の配置を最適化しても必ず表示領域の限界がある. 本提案では, 全体の表示倍率がいずれかの文字の最小可読文字サイズの倍率の半分未満となったとき時間軸に拡張し, 表示させる文字を時間で切り替え, 読みやすい文字情報を視聴者が見ることの出来る範囲を拡大する (図 10).

各文字列を表示させる時間の決定には, 人間の日本語文章の読む速度を用いる [14]. 人間が文字を読むと

配置と時間による可読性を維持した画像内文字提示手法



図 10. 時間拡張表示の例:(a) 元画像,(b) 時間拡張表示開始,(c)「ダイエット成分」が0.0~1.0倍に拡大アニメーションをしている途中,(d)「ダイエット成分」のアニメーション終了,(e)「ダイエット成分」の次の文字表示開始,(f)「30倍」の拡大アニメーション途中,(g)「30倍」のアニメーション終了

きは連続して文字を追わず、文章の一点に止まる停留と、次の注視点にすばやく移動するサッカーカードに分けられる。それぞれ平均すると停留時間は300ms、サッカーカードは15msで3.5文字程度を跳躍する。以上を用いて文字列の切り替わり表示時間を算出する。例えば10文字であれば $15 + 10 / 3.5 \times (300 \times 15) = 915\text{ms}$ と算出する。但し、315msに満たない場合、1文字でも停留時間が必ず存在するため315msとする。

算出された表示時間を用いて、文字列を順番に最小可読サイズの2倍で表示する。最小可読文字サイズは80%読める文字サイズであり時間拡張表示の時は自動配置表示に比べて文字を表示する領域が広く取れると考え、より読みやすい表示とするため文字サイズを2倍で表示することとした。表示位置は文字列の中心座標を変更せずに拡大して表示する。この際、単純に切り替えて表示すると、拡大された文字列がどこから出たのかわかりにくいため、切り替わりの前に、文字列の拡大率0.0から1.0へのアニメーションをすることでどの画像の情報であるか、視聴者にわかるように提示する。

5 結果と考察

本システムを用いて画像を縮小したものとそのまま縮小したものの比較を学生8人に対して行った。被験者に対し、16枚の画像の中から8枚の画像を

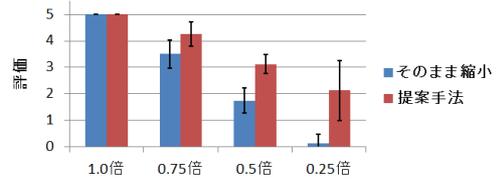


図 11. Q1の結果. 読みやすさの評価. 色棒は平均点, 誤差棒は標準偏差.

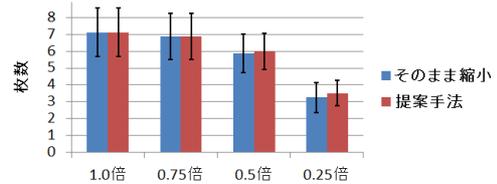


図 12. Q2.1の結果. 文字がないとした場合. 色棒は平均枚数, 誤差棒は標準偏差.

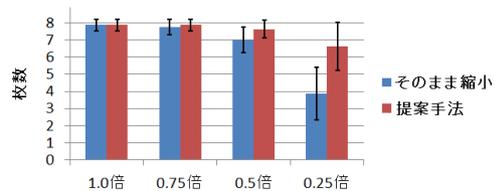


図 13. Q2.2の結果. 文字があるとした場合. 色棒は平均枚数, 誤差棒は標準偏差.

ランダムに2行4列に並び、1.0, 0.75, 0.5, 0.25倍で表示した際の見え方に関する評価を行った。画像は全て200×200を1.0倍とし、1.0倍でそのまま表示した際に見やすい位置に移動してもらい、その地点での読みやすさを非常に読みやすいとして、評価した。この際、頭の位置は動かさないようにした。

ユーザースタディでは次の2項目、及び自由記述欄を設け、気付いたことや気になった事を調査した。

Q1 文字は読みやすかったか

Q2 画像中の物体が何か認識出来た枚数は何枚か。

1. 文字がないとして考えた場合、
2. 文字があるとして見た場合についてそれぞれ記述してください

Q1では読みやすさの調査を目的とし、6段階の評価(5.非常に読みやすい, 4.読みやすい, 3.普通, 2.読みにくい, 1.非常に読みにくい, 0.読めない)を行った。Q2.1では、拡大された文字列によって商品の画像が見えず、認識出来なくなることへの懸念から調査をし、Q2.2では拡大された文字列が情報として役に立つか調査をした。Q2はそれぞれ認識出来た0~8の枚数をそれぞれ記述して貰った。

ユーザースタディの結果を図11,12,13に示す。

ユーザースタディの結果, 提案手法での表示がそのまま表示するよりも, 見えない文字がなく, また読みやすいという結果が得られた (図 11). 縮小していった結果, そのまま縮小すると 0.25 倍でほぼ見えないという結果に対して, 提案手法は読みやすいとは言えないが, 文字情報が無駄な情報にならず, 視聴者に伝わったと言える. 被験者のコメントでは「拡大されると情報が伝わりやすい」, 「文字が大きくなるアニメーション効果がよい」といった意見が得られ, 自動配置表示, 時間拡張表示共に効果的に文字情報を伝えることが出来たことや, 文字の拡大によって他の画像に文字が被ることを問題視する意見は得られなかった. 一方で, 文字領域抽出の不完全さから「潰れて見難い部分があった」という意見や 8 枚が順番にアニメーションされるため, 「後の画像の文字が出てくるのが遅い」という意見があり, 文字領域抽出の改善や時間拡張表示の改良がで, より効果的に文字情報を提示できることが期待される.

画像中の物体はそのまま縮小した場合と, 提案手法で縮小した場合とで物体の認識にほぼ差は見られなかった (図 12). よって画像の利点をなくすことができなく, 文字情報を視聴者に与えることが出来たと考えられる. また, Q1 と Q2.1 の 0.25 倍の部分より, 文字が見えないほど縮小した時でも, 何が画像内に写っているのかある程度わかることを示している. さらに, 文字があるとして見た場合, 文字の情報がそのまま縮小するより見えるため, 広告画像としての分かりやすさに文字列が役に立っている (図 13). しかし, Q2.1 では文字領域が拡大され, 画像だけで考えると文字で隠れる部分が多く, 認識率は下がるはずであるが, 0.5 倍と 0.25 倍で提案手法がわずかに高かったため, Q2 に関してはユーザースタディの方法を再考する必要があると考えられる.

6 まとめと今後の課題

本研究では, 簡単なユーザ入力で画像中の文字を指定するシステムの提案と, 画像を縮小した際に読みやすい画像中の文字の表示手法の有効性を, ユーザースタディで示した. そのまま画像を縮小するだけの表示方法ではなく, 自動配置表示や時間拡張表示といった表示方法はより効果的に情報を提示できる可能性をユーザースタディの結果は示唆している.

今後の課題としては, 時間拡張表示での待ち時間の改善や, 文字の大きさだけでなく文字と背景の色コントラストや, 書体, ウェイト等も取り入れることで, より効果的な情報提示手法を提案をしていきたいと考えている. また, 文字領域抽出の精度が見た目に関わるため, 精度を検証し改善することでより違和感のない表示とし, ユーザーの負担を減らすシステムとしていきたい.

参考文献

- [1] S. Avidan and A. Shamir. Seam Carving for Content-aware Image Resizing. In *ACM SIGGRAPH 2007 Papers*, SIGGRAPH '07. ACM, 2007.
- [2] Y. Boykov, O. Veksler, and R. Zabih. Fast Approximate Energy Minimization via Graph Cuts. *IEEE Trans. Pattern Anal. Mach. Intell.*, 23(11):1222–1239, 2001.
- [3] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient Graph-Based Image Segmentation. *Int. J. Comput. Vision*, 59(2):167–181, Sept. 2004.
- [4] H. Hiromi, O. Makoto, and O. Rikio. Detection of inserted text in images. In *In Symposium MEIS2013: Mathematical Progress in Expressive Image Synthesis*, 2013.
- [5] JIS S 0032:2003. 高齢者・障害者配慮設計指針-視覚表示物-日本語文字の最小可読文字サイズ推定方法.
- [6] P. Kaufmann, O. Wang, A. Sorkine-Hornung, O. Sorkine-Hornung, A. Smolic, and M. Gross. Finite Element Image Warping. *Computer Graphics Forum*, 32(2pt1):31–39, 2013.
- [7] F. Liu and M. Gleicher. Automatic Image Retargeting with Fisheye-view Warping. In *Proceedings of the 18th Annual ACM Symposium on User Interface Software and Technology*, UIST '05, pp. 153–162. ACM, 2005.
- [8] Y. Niu, F. Liu, X. Li, and M. Gleicher. Image Resizing via Non-homogeneous Warping. *Multimedia Tools Appl.*, 56(3):485–508, Feb. 2012.
- [9] A. Woodruff, A. Faulring, R. Rosenholtz, J. Morrisson, and P. Pirolli. Using Thumbnails to Search the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01, pp. 198–205. ACM, 2001.
- [10] O. S. Yu-Shuen Wang, Chiew-Lan Tai and T.-Y. Lee. Optimized Scale-and-Stretch for Image Resizing. *ACM Trans. Graph. (Proceedings of ACM SIGGRAPH ASIA)*, 27(5), 2008.
- [11] 宮下 佳子, 椎名 健. 書体の太さと視距離の関係における文字の見やすさ評価: 最小可読文字サイズから算出した最適文字サイズにおいて. 電子情報通信学会技術研究報告. HIP, ヒューマン情報処理, 107(369):109–112, 2007.
- [12] パナソニック株式会社. 井村 康治. 画像表示装置及び画像表示方法, 2011-6-16. WO2011070793 A1.
- [13] 平山 勝裕, 大町 真一郎, 阿曾 弘具. カラー情報を利用した情景画像中の文字列の高精度抽出. 電子情報通信学会技術研究報告, 104(742):91–96, 2005.
- [14] 神部 尚武. 読みの眼球運動と読みの過程. 国立国語研究所, 第 85 巻, pp. 29–65, 1986.