

# レスポンス Web デザインのための画像形式及び制作用インターフェース

藤井 陽介\* 福里 司\* 五十嵐 健夫\*

**概要.** 本研究では、多様化する端末の画面サイズに合わせて、Web ページ上の画像をレスポンスに変化させるための画像形式及び、画像編集システムを提案する。従来の方法では、デザイナーとエンジニア間のコミュニケーションの手間が大きく、表示結果に対する画像の編集や修正に非常に時間がかかってしまう。そこで本研究では、デザイナーが予め編集した情報を一つのファイルに直接格納することで、コミュニケーションを省きつつ、Web ページ上で端末ごとの編集結果を再現することを可能にした。

## 1 はじめに

近年、PC やタブレット、スマホ端末などの画面サイズに合わせて Web ページを見やすいおデザインで表示する「レスポンス Web デザイン」[3] が注目されている。レスポンス Web デザインは、端末ごとに Web ページ自体を制作しなおす必要が無い、「Web ページ上の情報の更新が容易」「アクセス解析がしやすい」などの長所を持つ。実際の現場では、(1) 事前に複数の画像を用意し、端末の画面サイズに合わせて画像を切り替える方法と、(2) 画像の一部をトリミングする方法、もしくは(3) リサイズ処理が用いられている。しかし、最初の手法は、複数の画像を用意しなければならない上、その閾値を設定しなければならないためデザイナーとエンジニア間で密なコミュニケーションが必要になる。その他の手法も、画像サイズによって情報量や視認性が大幅に低下する可能性がある。

そこで本研究では、デザイナーが各端末の画面サイズに合わせて画像を編集した際のパラメータをメタ情報として含む新しい画像形式 RVG (Responsive Variable Graphics) と、RVG 画像の制作用インターフェースを提案する。本インターフェースは、従来の画像編集 (トリミング、拡大縮小) に加え、画像中の不要な部分を削りながら縮める手法「シームカービング法」[2] における事前計算に基づくリサイズ機能を有している。また、制作された RVG 画像は各端末用に編集したパラメータ間を自動補間することで、任意の画面サイズに対応する機能を有している。本システムを用いることで、レスポンスな Web デザイン制作におけるデザイナーとエンジニア間でのコミュニケーションの手間を削減することができる (図 1)。

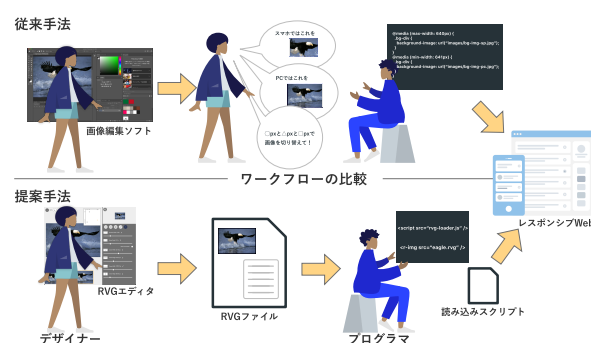


図 1. ワークフローの比較



図 2. 制作用インターフェースの全体像

## 2 提案手法

### 2.1 RVG 画像の制作用インターフェース

本インターフェースはデザイナーの普段使い慣れている Adobe Photoshop[1] などの画像編集ソフトウェアを参考に構成した (図 2)。

デザイナーはこのインターフェースを用いて、任意の画像を元に、各サイズでの (1) 拡大縮小、(2) シームカービング法の事前計算に基づく拡大縮小、(3) トリミング位置の 3 つのパラメータを編集する。補間結果や、編集途中における各サイズでの表

- ① CSSでr-imgのサイズをレスポンシブに設定する。

```
r-img {
  display: block;
  width: 100%;
  height: 100%;
}
```

- ② 読み込みスクリプトをロードし、r-imgタグを記述する。

```
<r-img src="image.rvg"></r-img>
<script src="rvg-loader.js"></script>
```

図 3. 実際に必要なコード

示結果はリアルタイムに確認することができる。そのためデザイナーは実際に表示される様子を確認しながら編集することができ、これによって実際に組み込まれた後の手戻りの量が減ることが期待される。

## 2.2 Web ページ上での RVG 画像の読み込み

出力した RVG 画像は、専用のスクリプトを用いて HTML 上で読み込みを行う。このスクリプトは、HTML タグである「*r-img*」タグを使用可能にするものである。このタグは *src* 属性に RVG 画像のファイルパスを渡すことで、RVG の画像を描画するものになっている。実際に使用する際は、まず *r-img* タグの *display* プロパティを *block* などに指定しておく。そして読み込みスクリプトをロードし、*r-img* タグを記述する(図 3)。

以上の手順で、ユーザは端末の画面サイズに合わせてレスポンシブな画像サイズの調整が可能となる。

## 3 議論

本論文における画像形式 RVG のプロトタイプは、現在のレスポンシブ Web デザインの支援になることに加え、Web ページ上の画像データに対する新たなユーザ体験を提案する。実際の Web ページ上の画像では、レスポンシブ対応の制作過程の煩雑さから、PC 向けとスマートフォン向けの 2 種類のみが用いられていることが多い。つまり、デザイナーが想定していない端末の画面サイズを用いて Web ページを閲覧した際、Web ページ全体のレイアウトが大きく崩れてしまう可能性がある。しかし、本システムを用いることで、任意の画面サイズに合わせて表示する画像を変化させることができるため、全体のレイアウトの段階でデザイナーの意図を反映させたユーザ体験を提供することが可能になる。ただし、従来の画像リサイズ技術と同様に、表示される画像が直感とは異なる可能性がある。今後の課題として、

画像形式に「どのパラメータを導入すべきか」「どの補間アルゴリズムを使用すべきか」などの選別と検証が挙げられる。

また、*background-image* を用いる画像のレスポンシブ対応(トリミング処理)の場合、画面サイズに合わせて複数の画像を容易する必要がないものの、領域指定などのパラメータを CSS 上で逐次設定しなければならず、デザイナーとエンジニア双方の負担が大きい。その一方、本プロトタイプはトリミング処理も考慮しつつ、コード上での手間を削減出来る点で有用であると考えられる。

## 4 おわりに

本研究では、Web ページ上の画像をレスポンシブに変化させるための画像形式及び、その画像編集システムを提案した。提案する画像形式によって、Web ページ上でデザイナーの意図を反映しつつ各端末向けの画像編集結果を効率的に表示することができる。編集できるパラメータとして従来のトリミング位置と拡大縮小率に加え、シームカービング法に基づくパラメータ(事前計算部)を導入することで、画像内のひとまとまりの図形や物体の形状を保持しながら画像のサイズの変更ができる。また、補間アルゴリズムを導入することで、少数の編集結果から任意の画面サイズにも自動対応することも可能にしている。本システムは、デザイナーとエンジニア間のコミュニケーションを省くことで、デザイナーは「見た目の調整」、エンジニアは「その他の機能の実装」に集中することを可能とし、レスポンシブ Web デザインの制作の大きな改善が期待できる。

## 参考文献

- [1] Adobe. Adobe Photoshop CC. <https://www.adobe.com/jp/products/photoshop.html>.
- [2] S. Avidan and A. Shamir. Seam carving for content-aware image resizing. *ACM Transactions on Graphics*, 26(3):10, 2007.
- [3] E. Marcotte. *RESPONSIVE WEB DESIGN*. A BOOK APART, 2011.