「Web ページ評価システム」

02H129 渡部 玲子

目次

1 はじめに 1

2 概要 2
2.1 現行のシステム 2
2.2 作成したシステム 2

3 説明 3
3.1 アクセシビリティ 3
3.2 Web コンテンツ JIS 3
3.3 色覚の障害 4
3.4 色彩と明度 4

4 システムの詳細 5
4.1 実行手順 6
4.2 実行結果 7

5 考察 8
5.1 利点 8
5.2 問題点 8

6 まとめ 8

付録 A 付録 1 10
A.1 付録 1.1 コントラストを求めて判定するプログラム 10

1　はじめに

インターネット上には多種多様な Web コンテンツが存在する。また、Web コンテンツを簡単に作成するツールによって個人の Web ページも多く存在し、今後も増加していくことが予想される。しかし、その中には高齢者や障害のある人に配慮されていない Web ページも存在する。例えばフォントが読みにくいであるとか、色に頼った表現などが挙げられる。これでは高齢者や障害をもつ人に正確な情報を提供できない恐れがある。利用者に等しく情報を提供できるよう、 Web ページ作成におけるガイドラインとして Web コンテンツ JIS(詳しくは 3.2 に後述) が存在するが、個人の Web ページでは守られていないところも多い。

そこで、GUI 環境における Web ページをブラウザに表示された状態で誰がみても情報が正確に読み取れるかをチェックするシステムを提案する。このシステムは Web ページのレイアウトを制限するものではない。

また、本当のバリアフリーというのは、お互いが気をつかいあってお互いの負担になることではなく、お互いが対等の位置にあることだと、私は考える。誰もが自由なレイアウトで Web ページを作成し、ほんのちょっとの配慮で誰でも自由に閲覧できる Web ページにすることができる、というのが理想的ではないだろうか。

2　概要

Web ページを画像として扱い、プログラムによってコントラスト値をカウントする。そのコントラスト値から、アクセシビリティに配慮されている Web ページかそうでない Web ページかを判別してメッセージを返す。

2.1　現行のシステム

W3C[*1] がコードチェックを提供している。

また、WAT-C(Web アクセシビリティ－ ・ツール・コンソーシアム[*2]) という団体が、Web コンテンツ JIS に沿った Web ページ作りのためのツールを無償で提供している。

http://www.wat-c.org/jp/

特に色に関してのチェックツールにはカラー・コントラスト・アナライザー[*3](CCA) という前景色と背景色を選択し、色覚の障害の種類別の見え方と明度差 (コントラスト) を表示するツールが存在する。

http://html.idena.jp/program/cca.html

2.2　作成したシステム

CCA では単色同士のコントラストについてチェックするツールだが、本システムは Web ページ全体を画像として評価する。これにより画像、背景画像やロゴも含めた全体の評価をすることができる。また、画像なので特定の部分のみを切り出して評価することもできる。

画像として扱って評価するシステムは他にも存在するが、それは色の見え方を変化させてチェックするものである。このシステムでは 1 ピクセル毎にコントラストを数値で算出し、その割合も数値で算出し、判定する。よって判定の線引きがはっきりとしている。

また、色の見え方を変化させてチェックするものは使用しているディスプレイによって感じ方が変化してしまうが、このシステムでは数値による判定なので左右されない。

[*1] www で利用される技術の標準化をすすめる団体。www 技術に関わりの深い企業、大学、研究所、個人などが集まって 1994 年 10 月に発足した。 http://www.w3.org/

[*2] 特定の目的を達成するために形成された複数の企業などによる集まりのこと

[*3] 解析装置

3 説明

3.1 アクセシビリティ

Web アクセシビリティー (Accessibility) とは、ハードウェア、ソフトウェア、サービスなどの情報技術を、複数の支援技術 (アシスティブ・テクノロジー) を併用、または単独で使用することによって、ユーザの閲覧環境を問わず、Web を利用しての情報の送受信ができること。あるいは、その使いやすさを意味する。[*4]

アクセシビリティというのは、障害のある人でも高齢者でも、誰でも利用できるかどうか、利用はしやすいか、アクセスしやすいか、ということを指す。Web アクセシビリティというのは Web 上におけるアクセシビリティであるので、Web コンテンツが誰でも利用できるかどうか、ということである。

Web アクセシビリティに関しては以下で説明する「Web コンテンツ JIS」や、W3C によるガイドラインなどが存在している。

3.2 Web コンテンツ JIS

Web コンテンツ JIS とは、「高齢者・障害者等配慮設計指針− 情報通信における機器、ソフトウェア及びサ− ビス第 3 部:ウェブコンテンツ JIS X8341-3:2004」[*5]のことをさす。2004 年 6 月に公示・出版されたウェブコンテンツのアクセシビリティのガイドラインで、高齢者や障害者にも配慮された Web コンテンツの作成に関する項目について決められている。

今回はその中でも特に色覚に障害をもつユーザーに配慮した Web コンテンツの作成に関する項目に着目した。色覚に関する項目では以下のような点について定められている。

- 色のみに依存する情報提示
- 形・位置のみに依存する情報提示
- 背景色と前景色のコントラストと配色

[*4] http://www0.ise.osaka-sandai.ac.jp/fken2003/tori/

[*5] Web コンテンツ JIS は Web 上からダウンロードすることができる (有料)。閲覧だけであれば以下の URL からデータベース検索 > JIS 検索 > JIS 規格番号 X8341-3 から内容を読むことができる。http://www.jisc.go.jp/

3.3 色覚の障害

色覚に障害があるというのはどういうことなのかというと、色が全く判別できないというよりは判別認識できる色が少ない、という表現のほうが適切であるといえる。

眼の網膜には 3 種類の錐体が含まれていて、それぞれ赤、緑、青を感じる視物質を持っている。色覚の障害は、1 つ以上の種類の錐体がすべて正常なレベル以下で機能している、またはまったく機能していない場合に発生する。

色覚の障害の種類には以下のようなものがある。

- 第一色盲
  - 赤の視感度がない。赤への視感度を提供する L 錐体の機能が失われていることが原因
- 第一色弱
  - L 錐体は機能しているが、本来のレベルに達していない
- 第二色盲
  - 緑の視感度がない。緑への視感度を提供する M 錐体の機能が失われていることが原因
- 第二色弱
  - M 錐体は機能しているが、本来のレベルに達していない
- 第三色盲
  - 青の視感度がない。青への視感度を提供する S 錐体の機能が失われていることが原因

上記のような色覚の障害の種類によってはグレースケールではっきり見えても、実際には障害をもつ人からは見えないことがある。よって、グレースケールに変換することで十分に確認できるわけではない。

重要なのは明度によるコントラストである。コントラストの値が高ければ、色覚に障害があっても見ることができるといわれている。

3.4 色彩と明度

発光や反射光を波長分解すると色彩の違いがわかる。しかし波長成分のグラフによって表色するのは人間向きとはいえない。そこで色を数値で表わすために色の三属性が考案された。明度と彩度と色相である。ここでは色の見やすさに関係する明度のみにしぼって説明する。

明度は色の明るさを表す。最も明るい色は白で、最も黒い色は黒となる。明度は文字の読みやすさ (可読性) や図版の見やすさ (視認性) などに影響を与える。明度の差が低ければユーザーにスムーズに伝えることができない。スムーズに伝えるためには一定の明度差を確保する必要がある。ppm 形式の画像ファイルにおいて、色は 10 進数 0 から 255 までの RGB 値で表されている。その R(Red)G(Green)B(Blue) の 3 つの値から、以下の式により各ピクセルの明度を算出することができる。

Y(明度)=R*0.299+G*0.587+B*0.114

4　システムの詳細

Web ページをキャプチャツールを使い画像としてローカルに保存する。保存した画像は ppm 形式に変換し、C 言語で組んだプログラムに渡す。プログラムは ppm ファイルに含まれる RGB の数値から各ピクセルの明度を算出する。画像の周囲 1 ピクセルを除く全ピクセルを、各ピクセルの周囲 8 ピクセルと比較し、コントラストが低い箇所があればカウントする。そのカウントが一定値を越える、つまりコントラストが低い箇所が支障がある程に多ければ「見えにくいです (割合)」と返す。一定値に至らなければ「問題ありません (割合)」というメッセージを返す。

見やすいコントラストは 125 以上とした。これは W3C でコントラストの値は 125 以上と定められているからである。

また、低コントラストのカウントの一定値は以下の Web ページの一部を評価し、その低コントラストの値と実際の見やすさを参考にして設定した。ブラウザは Mozilla を使用した。

- Yahoo!Japan[http://www.yahoo.co.jp/]
- Google[http://www.google.co.jp/]
- Vine Linux[http://vinelinux.org/]
- Slash.Japan[http://slashdot.jp/]
- 大阪産業大学 [http://www.osaka-sandai.ac.jp/]
- 大阪産業大学 工学部 [http://www.osaka-sandai.ac.jp/kougakubu.html]
- 大阪産業大学 工学部 情報システム工学科 [http://www.ise.osaka-sandai.ac.jp/]
- 大阪産業大学 工学部 情報システム工学科 情報安全工学研究室 [http://www0.ise.osaka-sandai.ac.jp/]
- 大垣 斉 講師 [http://www.ise.osaka-sandai.ac.jp/%7Eandrew/]
- e-word[http://e-words.jp/]
- W3C[http://www.w3.org/]
- WAT-C[http://www.wat-c.org/jp/]
- goo[http://www.goo.ne.jp/]
- その他個人で運営されている Web ページ

テストの結果から以下のようなことがわかった。

- コントラストに配慮している Web ページは文字数、情報量が多くても低コントラストのカウントが 10% 程度に収まっている。
- 逆に、コントラストに配慮していないページでは情報量が多くない場合でも 50% 程度まで達してしまう。しかし、情報量が極端に少ない、またはブラウザに対して文字が大きいときは正常に動作しないことがある。
- 写真やロゴマークの有無だが、写真があっても配慮されているページであれば 30% を越えることはない。これはカラー写真とモノクロ写真の両方が掲載されている Web ページで確認した。

以上のことから、低コントラストになっている箇所が 25% 以上あればエラーメッセージを返すことにした。
また、このプログラムは情報量、写真の有無などに左右されにくいことが証明された。

4.1　実行手順

Web ページ評価システムを実装した手順は以下の通りである。

1.Web ページをキャプチャツールを使い画像としてローカルに保存する。

2.convert ファイル名. 拡張子 iro.pnm

pnmnoraw iro.pnm > iro.ppm

で ppm ファイルに変換する

3.gcc sotsu.c　でコンパイルする

4../a.out　で実行する

プログラムのソースは末尾の付録に記載する。

4.2 実行結果

```
[reiko@lindow01 sotsu]$ convert testw3c.png iro.pnm
[reiko@lindow01 sotsu]$ pnmnoraw iro.pnm > iro.ppm
[reiko@lindow01 sotsu]$ gcc con.c
[reiko@lindow01 sotsu]$ ./a.out
問題ありません(11.7%)
```

図 1　低コントラストの箇所が少なく問題がない場合

```
[reiko@lindow01 sotsu]$ convert testaoiro.png iro.pnm
[reiko@lindow01 sotsu]$ pnmnoraw iro.pnm > iro.ppm
[reiko@lindow01 sotsu]$ gcc con.c
[reiko@lindow01 sotsu]$ ./a.out
見えにくいです(39.3%)
```

図 2　低コントラストの箇所が問題となるくらい多かった場合

5　考察

5.1　利点

- 色だけでコントラストをチェックするのではなく、作成したものを画像として評価するので、ページ全体としての評価ができる。
- 既存の Web ページでもそのまま評価することができる。
- 普通の画像として扱うことができるので、評価したい箇所だけを評価することができる。
- 情報の量や写真の有無によらず評価できる。
- プログラムは c 言語で組んだので、どの OS でも利用できる。

5.2　問題点

- キャプション機能はついていないので、自分で画像として保存しなくてはいけない
- 淡くて細かすぎる模様に対応していない。
- ppm ファイルはサイズが大きい。
- 1 ページずつしか評価できない。
- 文字数が極端に少ない、または文字がブラウザに対して大きいために、低コントラストのカウントが一定値に達さずクリアしてしまうことがある。

6　まとめ

問題がないのにエラーメッセージを出したケースは殆どなかったことから、Web 上にある Web コンテンツは大体の場合評価できると考えられる。

ただし、大きな問題点がいくつか存在するので、それを改善しなくては実用化は難しい。特に模様によっては問題がないのにエラーメッセージを表示してしまう点を改善すべきだった。これでは Web ページのレイアウトの自由度を下げてしまう恐れがあり、本来の目的である自由なレイアウトで Web ページを作成する、ということから離れてしまう。

謝辞

本研究を進めていく上で御指導及び御協力を頂いた大垣 斉講師・藤井 信夫教授、fken.a4w メイルリストのメンバーの方々、そして情報システム工学科の卒研生、院生、居候の方々に深く感謝の意を表します。

参考文献

[1] 2003 年度卒業生 鳥坂武志 [http://www0.ise.osaka-sandai.ac.jp/fken2003/tori/]
[2] RFC792, Internet Control Message Protocol
[3] IT 用語辞典 e-Words[http://e-words.jp/]
[4] W3C CSS 検証サービス [http://jigsaw.w3.org/css-validator/]
[5] WAT-C[http://www.wat-c.org/jp/]
[6] eye's Inc. ウェブコンテンツ JIS 規格 [http://xixyoxu.dip.jp/eyes/JIS/]
[7] JISC 日本工業標準調査会 [http://www.jisc.go.jp/]
[8] JIS X 8341-3:2004「高齢者・障害者等配慮設計指針-情報通信における機器、ソフトウェア及びサービス-第 3 部:ウェブコンテンツ」技術解説第 1.1 版委員会ワーキングドラフト (7 月 22 日版) [http://www.jsa.or.jp/domestic/instac/commitee-acc/web-tech-repo/technical-report.html]
[9] 色覚に障害を持っていたとしたらあなたのサイトは見えるでしょうか [http://www.microsoft.com/japan/msdn/columns/hess/hess10092000.asp]
[10] The Web KANZAKI[http://www.kanzaki.com/]
[11] 創作プログラミングの街 [http://www.sm.rim.or.jp/ shishido/]
[12] 碧色工房 [http://www.mm2d.net/]
[13] 進歩し続ける Web デザイナーの考え方 (ISBN4-88337-432-7)
[14] Web Site expert #4(ISBN4-7741-2517-2)
[15] Web creators 2005 年 8 月号 vol.44
[16] コンピュータ図形処理の基礎 (ISBN4-320-02601-2)

※ URL はすべて 2005 年 1 月 16 日時点のもの

付録 A　付録 1

A.1　付録 1.1 コントラストを求めて判定するプログラム

```
#include<stdio.h>
#include<stdlib.h>

int main(void)
{
/*宣言*/
double cont=0; /*コントラストの一時的な入れ物*/
double **meido; /*明度を格納*/
double wari=0; /*割合*/
double zen=0;   /*チェックしたピクセルの総量*/

char str[50];

int tei=0; /*低コントラストのカウンタ*/
int heigh,width; /*画像の縦と横*/
int **r,**g,**b; /*RGB 値を格納*/
int i,j,k; /*カウンタ*/
int *img; /*元の画像*/
int d;     /*階調*/

FILE *fp; /*ファイル読み出しの宣言*/

/*ファイルの読み出し*/
 if((fp=fopen("iro.ppm","r"))==NULL)
   {
     fprintf(stderr,"ファイルがオープンできません\n");
     exit(1);
   }
 else{
  fscanf(fp,"%s",&str);     /*ファイル形式 */
  fscanf(fp,"%d %d",&width,&heigh);  /* 画像の大きさの読み込み */
  fscanf(fp,"%d",&d);       /* 画像の階調の読み込み */

/*メモリ確保*/
  img=(int *)malloc(sizeof(int)*width*heigh*3);
  if(img==NULL){
    printf("メモリが確保できません\n");
    exit(1);
```

```
  }

/*ファイルの中身を img に格納*/
for(k=0;k<=width*heigh*3;k++){
 fscanf(fp,"%d",&img[k]);
   }
 }

/*ファイルを閉じる*/
fclose(fp);

/*R メモリ確保*/
  r=(int **)malloc(sizeof(int)*width);
  for(i=0;i<width;i++)
    r[i]=(int *)malloc(sizeof(int)*heigh);
  if(r==NULL){
    printf("メモリが確保できません\n");
    exit(1);
  }

/*G メモリ確保*/
  g=(int **)malloc(sizeof(int)*width);
  for(i=0;i<width;i++)
    g[i]=(int *)malloc(sizeof(int)*heigh);
  if(g==NULL){
    printf("メモリが確保できません\n");
    exit(1);
  }

/*B メモリ確保*/
  b=(int **)malloc(sizeof(int)*width);
  for(i=0;i<width;i++)
    b[i]=(int *)malloc(sizeof(int)*heigh);
  if(b==NULL){
    printf("メモリが確保できません\n");
    exit(1);
  }

/*img から数値読み出して rgb に格納*/
k=0;
for(j=0;j<heigh;j++){
   for(i=0;i<width-2;i++){ /*はみ出さないように-2 する*/
    r[i][j]=img[k];
```

```
    g[i][j]=img[k+1];
    b[i][j]=img[k+2];
    k=k+3;
   }
}

/*メモリの開放*/
 free(img);

/*明度のメモリ確保*/
  meido=(double **)malloc(sizeof(double)*width);
  for(i=0;i<width;i++)
    meido[i]=(double *)malloc(sizeof(double)*heigh);
  if(meido==NULL){
    printf("メモリが確保できません\n");
    exit(1);
  }

/*明度へ変換*/
for(j=0;j<heigh;j++){
   for(i=0;i<width;i++){
    meido[i][j]=r[i][j]*0.299+g[i][j]*0.587+b[i][j]*0.114;
   }
}

 free(r);
 free(g);
 free(b);

/*コントラスト（中央）*/
for(j=1;j<heigh-1;j++){
   for(i=1;i<width-1;i++){
    cont=meido[i][j]-meido[i][j+1];
    if(cont<0){
      cont=cont*(-1);
    }
    if(cont!=0 && cont<125){
      tei++;
    }

    cont=meido[i][j]-meido[i][j-1];
    if(cont<0){
      cont=cont*(-1);
```

```
}
if(cont!=0 && cont<125){
  tei++;
}

cont=meido[i][j]-meido[i+1][j];
if(cont<0){
  cont=cont*(-1);
}
if(cont!=0 && cont<125){
  tei++;
}

cont=meido[i][j]-meido[i-1][j];
if(cont<0){
  cont=cont*(-1);
}
if(cont!=0 && cont<125){
  tei++;
}

cont=meido[i][j]-meido[i+1][j+1];
if(cont<0){
  cont=cont*(-1);
}
if(cont!=0 && cont<125){
  tei++;
}

cont=meido[i][j]-meido[i+1][j-1];
if(cont<0){
  cont=cont*(-1);
}
if(cont!=0 && cont<125){
  tei++;
}

cont=meido[i][j]-meido[i-1][j+1];
if(cont<0){
  cont=cont*(-1);
}
if(cont!=0 && cont<125){
  tei++;
```

```
    }

   cont=meido[i][j]-meido[i-1][j-1];
    if(cont<0){
      cont=cont*(-1);
    }
    if(cont!=0 && cont<125){
      tei++;
    }
   }
}

 free(meido);

/*結果を表示*/
zen=(width-2)*(heigh-2)*8;
wari=tei/zen*100;
if(tei!=0 && wari>25){
 printf("見えにくいです (%.1f%%)\n",wari);
}
 else
   printf("問題ありません (%.1f%%)\n",wari);

 return 0;
}
```