USING COLOR AND COMPOSITION TO CLASSIFY WEB PAGES

Thomas van den Berg

<thomas.g.vandenberg@gmail.com>

5789346

Bachelor Thesis
Credits: 15 EC

Bachelor Opleiding Kunstmatige Intelligentie

Faculty of Science
University of Amsterdam
Science Park 904
1098 XH Amsterda

Supervised by:

dr. M.W. van Someren
Informatics Institute
Faculty of Science
Universiteit van Amsterdam
Science Park 107
1098 XG Amsterdam

V. de Boer, MSc.
Department of Computer Science
Faculty of Sciences
Vrije Universiteit Amsterdam
De Boelelaan 1081a
1081 HV Amsterdam

June 27th, 2010

Abstract

This article describes a method of classifying webpages using features extracted from the design. Design features are language independent and correspond well to a user's first impression of a page. A number of new feature extractors were developed, capable of extracting human-readable features. These features are based on guidelines for use of color and composition in web design. The composition feature extractors detect photos and columns on a page. The color feature extractors consist of a method for extracting a color palette from a web page and analyzing this palette. A new dataset was created and labelled. The new features lead to better classification of web pages when classes are based on visual characteristics. They can also give insight into what causes a page to look good, modern or commercial.

1 INTRODUCTION

The first impression of a website can be the most important one. When a user visits a page on the internet, he will immediately recognize certain characteristics. Even before a user reads the first word on the page, he probably has an idea about the *kind* of page in front of him. Earlier research has shown that it is possible to classify web pages based on this *Look and Feel* [de Boer and van Someren, 2008]. Even though the extracted features were effective for classifying the pages, they leave a lot of guessing about what exactly makes a page look beautiful, ugly, new, or old. In order to establish a better understanding of the relationship between the extracted features and classification, this research is focused on extracting features that are similar to those a human observer would notice. Hopefully this method will also improve results when classifying pages. My approach is to extract features that are directly related to the color and composition of the page's design. As an example, in earlier research the number of photos was shown to be correlated to subjects' perception of the *quality* of a page [Amento et al., 2000]. And it was mentioned in [de Boer and van Someren, 2008] that "[m]ore specific color features such as the adherence to 'good' design colors schemes can produce better results".

There are three advantages to this approach. First – and most obviously – it could lead to better classification of web pages. Secondly, it will make it possible to give specific *feedback* about how a page was classified, because the features are related to design choices. Finally, knowing which features correspond to beautiful pages might give us *new insights* into what constitutes good web page design.

Four new feature extractors were developed to extract the number of photos, the size and position of columns, the color palette that was used, and to analyze this color palette. These are described in Section 2. The predictive value of these features was then tested in a classification experiment, where websites were classified according to aesthetics, recency, topic and whether they were commercial or informative (Section 3).

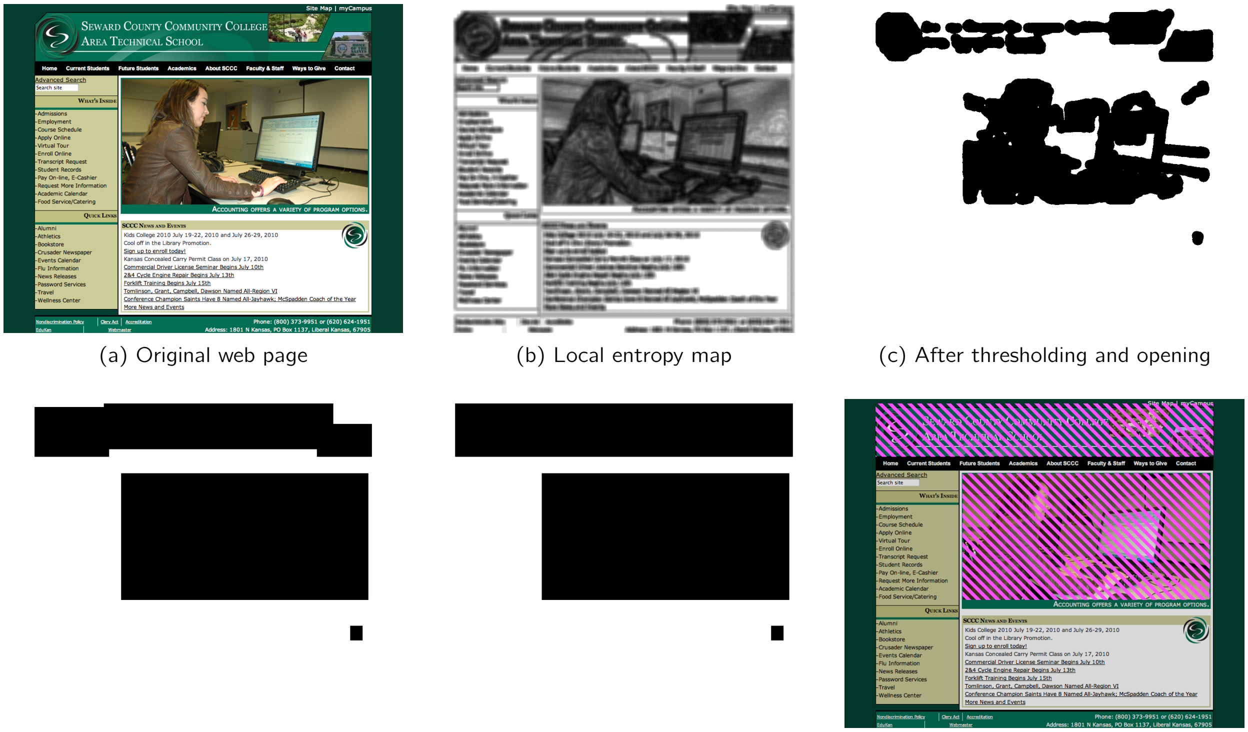
2 COLOR AND COMPOSITION FEATURES

In this section, the four new feature extraction algorithms are described. Two of these characterize the composition of the page, and two other characterize the colors. All of them work on an image, the HTML of a web page is not used. Each web page is rendered in a browser, and then a screenshot of 1200 pixels wide and 1000 pixels high is taken of the upper portion of the page. Most pages fit entirely within this 1000 × 1200 pixel window, but for pages with more content the clipping corresponds to the part that the visitor sees before scrolling down. The feature extractors were written in MATLAB.

2.1 Composition: Detecting Photos

An important factor when people make up their minds about a web page is whether it has photos. A few photos can make a page more salient, and a lot of photos make it look cheap or confusing. Knowing what areas of a page consist of photos can also make it easier to extract a color palette, because photos usually do not employ the same colors, more about this in Section 2.3. A new algorithm for detecting photos on a webpage is described in this section.

Web pages are usually designed using mostly flat colors or subtle textures. Photos are much "noisier" than most parts of a webpage. Text also adds noise to the page, but it does not form contiguous areas. Because of this, images and photos will stand out as large, contiguous noisy areas. Noise can be characterized by the entropy of an area. Rectangular areas with high local entropy can thus be marked as images. Not every image has a high entropy everywhere (e.g. a blue sky might cause smooth parts in a photo). However, it is possible to exploit the fact that images on web pages are usually rectangular and aligned to the page. If we assume that each high-variance area is contained in a rectangular image region, we'll discover that some of these rectangular areas overlap, making them part of the same image. By repeatedly merging image areas whose *axis-aligned bounding boxes* overlap, we'll end up with a very precise estimation of which pixels belong to images or photographs on the page These steps are explained below and illustrated in Figure 1.


Seward County Community College
Area Technical School
Accounting offers a variety of program options.

(a) Original web page

(b) Local entropy map

(c) After thresholding and opening

(d) Filled up bounding boxes

(e) Filled up bounding boxes again

(f) Striped areas indicate photos

Figure 1: Detecting photo areas

(1) Convert the image to a grayscale image.

(2) Use a local ($w \times w$) pixel filter to create a map of the entropy of each pixel's neighbourhood. (The function `entropyfilt()` in MATLAB).

(3) Threshold the entropy map at a minimum (t).

(4) Perform an *opening* of the binary image, with a $w \times w$ square structuring element.

(5) Repeat until image doesn't change:

   (a) For each contiguous area, set every pixel in its *axis-aligned bounding box* to 1.

(6) Remove contiguous areas that are smaller than b pixels in either dimension.

In this research, parameters were chosen as follows: $t = 3.4$, $w = 11$, $b = 60$.

The areas in which photos were detected are then used to create the following features.

Number of photos The number of isolated areas that have been marked as photos.

Average photo area The average area in pixels per photo.

Total photo area The total area in pixels that is covered in photos.

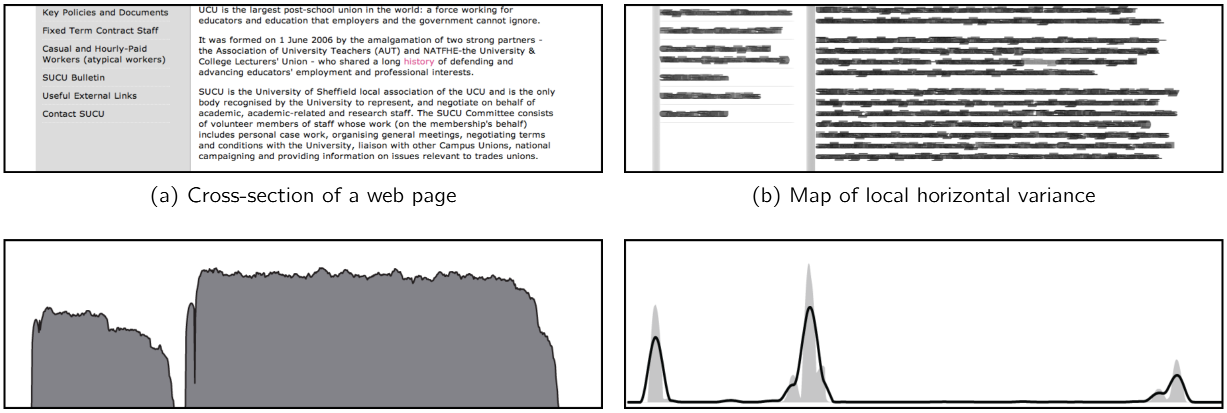
Evaluation of Photo Detection Algorithm

To evaluate the precision of the photo finding algorithm, it was used to find image areas on a random 50 page sample of the Discovery Challenge dataset (Section 3.2). Afterwards, the results were inspected and the number of correct classifications, false positives, and false negatives were counted. True negatives (non-image areas that were classified as such) were not counted, because it is not clear what counts as *a single* non-image area. In the non-photo areas a distinction was made between text and *graphics*. It seems reasonable that areas with large graphic elements such as logos and illustrations would also be marked as photos, since they have much the same features. Of the 101 photos on the pages, 93 were detected, yielding a sensitivity of 92%. Regarding false positives, 18 pieces of text were classified as photos, yielding a positive predictive value of 84%. If the 49 graphics marked as photos are counted as false positives, the positive predictive value drops to 58%.

2.2 Composition: Detecting Gutters

Most webpages are divided in columns. These columns can be used for navigation menus, sidebars, widgets, etc. The text of a site can be divided into columns as well. The number and size of these columns is an indicator for different types of websites. Many equal-sized text columns indicate a magazine style layout, often used on news sites, while shopping sites tend to have a small menu column on the left side, and a large center column displaying products. The whitespace between these columns is usually referred to as a "gutter". By detecting these gutters, we can get an indication of the number and size of the columns used on the page.

To detect gutters, we use the fact that the *local variance* in an image along horizontal lines is more or less constant inside of a column. In a text column, this variance will be high, as each horizontal section will contain text pixels as well as non-text pixels. In empty columns – or outside the boundaries of the page – the variance will be close to zero. After summing this local variance along columns, sections will appear where the variance is more or less constant, indicating columns. The positions where this variance changes from one constant value to another indicate gutters. The algorithm is shown in detail below, and illustrated in Figure 2.



UCU is the largest post-school union in the world: a force working for educators and education that employers and the government cannot ignore.

It was formed on 1 June 2006 by the amalgamation of two strong partners - the Association of University Teachers (AUT) and NATFHE-the University & College Lecturers' Union - who shared a long history of defending and advancing educators' employment and professional interests.

SUCU is the University of Sheffield local association of the UCU and is the only body recognised by the University to represent, and negotiate on behalf of academic, academic-related and research staff. The SUCU Committee consists of volunteer members of staff whose work (on the membership's behalf) includes personal case work, organising general meetings, negotiating terms and conditions with the University, liaison with other Campus Unions, national campaigning and providing information on issues relevant to trades unions.

(a) Cross-section of a web page

(b) Map of local horizontal variance

(c) Vertically summed local variance

(d) Moving variance of (c) shown as grey area, smoothed shown as black line. Peaks indicate gutters.

Figure 2: Detecting Gutters

(1) Convert the image to a grayscale image (*im*).

(2) Use a local $(1 \times s)$ pixel filter to create a map of the standard deviation of the values in each pixel's horizontal neighborhood.

$$stdmap_{ij} = \text{stdev}([im_{i,(j-\frac{s-1}{2})} \cdots im_{i,(j+\frac{s-1}{2})}])$$

(3) Sum the standard deviation map along columns, yielding a $(1 \times n)$ vector.

$$sums_j = \sqrt{\sum_{i=1}^{m} \left[stdmap_{ij}\right]}$$

(4) Compute the *moving variance* with span w.

$$vars_j = \text{var}([sums_{j-\frac{w-1}{2}} \cdots sums_{j+\frac{w-1}{2}}])$$

(5) Smooth the values using a *moving average* filter with span w to merge neighboring peaks.

(6) Find local maxima (*peaks*). A peak is an element whose value is larger than both its neighbors. Only peaks larger than threshold t and further than d pixels apart are returned. The threshold depends on the mean of the local variance vector. The locations of these peaks indicate gutters in the page.

$$t = f \cdot \overline{vars}$$

(7) Finally, a vector is constructed containing the *width* of each column bounded by two gutters.

$$colsizes_p = peak_{(p+1)} - peak_p$$

In this research, parameters were chosen as follows: $s = 13$, $w = 25$, $f = 0.6$, $d = 50$.

The positions of the detected gutters are used to generate the following features. The same feature extractor was also used on the transposed image, so that rows instead of columns are analyzed. The gutter detection algorithm was not separately evaluated.

Left margin The area on the left side of the page that is not used.

Page width The width of the area used by the web page.

Number of columns The amount of gutters, minus one.

Average column width Mean of column sizes.

Maximum column width Width of the largest column in pixels.

Minimum column width Width of the smallest column in pixels.

First column width Size of the first column.

Irregularity The standard deviation of the column sizes.

2.3 Color: Extracting Palettes

When designing a website or other publication, a designer is aiming for a certain Look and Feel, trying to appeal to a specific audience or trying to evoke a specific response from a viewer. Color is the most important tool for this, so the intended purpose of a website usually shines through in its use of color. Besides just extracting the colors used in the image, it is useful to analyze the palette itself. A *bold* or *subdued* palette is defined by the *relation* of the colors, not by single colors.

A color palette usually consists of thee main colors [Boulton, 2007]. The most used color is the *base color*, this is usually a neutral tint, used as the background color on most web pages. The most notable color, though, is the *dominant color*, this is often the color of graphical elements such as menus and page headers. Then there is *highlight color*, this color can either be a stronger tint of the dominant color or a contrasting color, and is used as an accent. The highlight color is used for elements that require the user's attention, such as navigation buttons and section headers. In web design, a fourth color is employed for practical reasons: the color of the text is picked so that it has a high contrast with the background [W3C, 2008]. The text color usually has a low saturation, regardless of the other colors in the palette.

Requirements

There are existing techniques to extract color palettes for use in reducing the number of colors in a photo (e.g. [Kruger, 1994]). These algorithms focus on minimizing the difference between the original photo and the color-reduced photo. However, when extracting a color palette from a website these algorithms are not satisfactory. A lot of colors appear in a photograph, while on web pages there are usually a small number of flat colors. The goal is to extract a small number of colors that are representative of the colors that the designer used to design the page. From this, three requirements from the extraction algorithm emerge.

A. The algorithm should return the actual colors in the image, not necessarily colors that yield the least error when reproducing the image.

B. Visually proximate tints of the same color should be grouped together.

C. The palette should have a representation of the prevalence of each color, so that highlights are distinct from the background and main colors.

When using a standard clustering algorithm on the pixels of an image, the result is close to satisfactory, but the results have to be manually tweaked.

Clustering Colors

The algorithm used is essentially k-means clustering in the $L^*a^*b^*$ color space [MacEvoy, 2005]. The $L^*a^*b^*$ color space has three dimensions: L^* is the perceived *lightness* of a color, a^* indicates whether the color is green or red, and b^* indicates whether it is blue or yellow. The dimensions are nonlinear functions of RGB values, approximating human color perception. If two pairs of $L^*a^*b^*$ color vectors have the same Euclidian distance, they will appear equally different to a human observer. Additionally, if two colors have the same L^* value, they will appear equally bright. Because the Euclidian distance in the $L^*a^*b^*$ color space is a measure of the perceived difference between colors, Requirement B is satisfied. In the RGB color space, colors can have a large Euclidian distance, even though they appear very similar to human observers. Clustering in RGB has the effect that a lot of grays will be selected, because their values can vary greatly, while still appearing similar.

Clustering works on *every* pixel in the image, not just a set of the unique colors. In a default implementation, k-means clustering returns the *centroids* for each cluster, these do not necessarily coincide with actual points in the clustered data. Instead, this algorithm returns the colors that occur most within each cluster (the mode), taking care of Requirement A. The relative size ($\frac{pixels\_in\_cluster}{total\_pixels}$) of each cluster is returned, satisfying Requirement C. Colors in a palette are ordered by their relative size. Any number of colors can be extracted, but 6 colors were chosen, because those include the 4 colors of a palette and some room for noise. Some examples of the extracted palettes can be seen in Figure 3 and the exact procedure is described below.

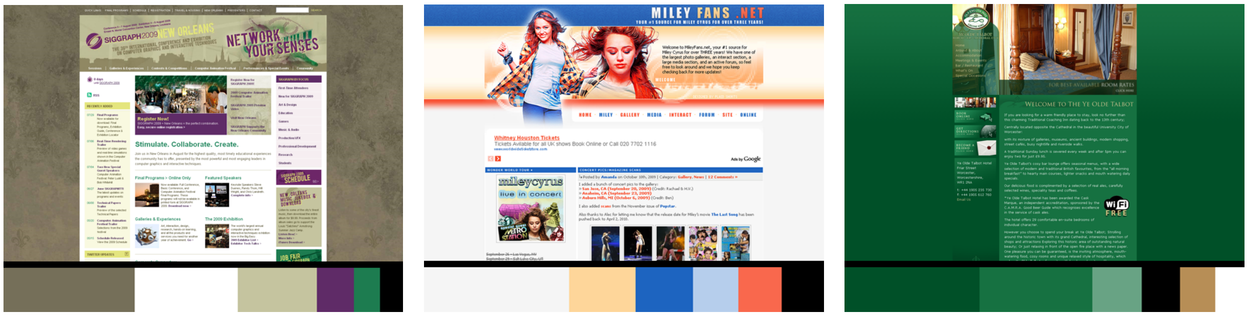


Figure 3: Examples of extracted palettes

(1) Subsample the image with factor s.

(2) Convert the image to the $L^*a^*b^*$ color space.

(3) Reshape the $m \times n \times 3$ image matrix to a $(m \cdot n) \times 3$ list of pixels.

(4) Use k-means clustering to obtain 6 clusters.

(5) From each cluster, retrieve the most occurring color and relative size.

(6) Return *palette* $P_1 \cdots P_6$.

In this research, parameters were chosen as follows: $s = 0.4$.

2.4 Color: Analyzing Palettes

In order to classify images based on the color palettes, we have to extract features. Of course the $L^*a^*b^*$ values of each color serve as a starting point. Because the colors are ordered by size, the first color is usually the background, and the first L^* coordinate thus indicates the lightness of the background. Whether a page uses a dark or light background is a good indicator of its topic, so raw color values are useful. However, there are more decisions a designer makes when creating a color palette. By looking at relations between colors in the palette, more features can be extracted. These features are explained in the next section. For some of these it is useful to adopt the *hue, chroma, luma* color space, which is identical to the *hue, chroma, intensity* colorspace described in [Joblove and Greenberg, 1978]. The difference is that *luma* was used because it is more perceptually relevant than *intensity*. *Luma* is calculated as: $luma = 0.2989 * R + 0.5870 * G + 0.1140 * B$.

Using the "Müller Formula" to predict color preferences

Though most of the features extracted from a palette are quite straightforward, the "Müllerness" requires some additional explanation. There are two phenomena that explain the relevance of this feature, described in [Müller, 1948]. The first is that colors with different hues, have a different perceptual "brightness". Even though the *saturation* and *value* are the same, people will judge yellow to be "lighter" than blue. This is reflected in the $L^*a^*b^*$ color space, where the L^* component is a *nonlinear* function of the RGB values (Figure 4). This perceptual brightness of the fully saturated color is referred to as the "natural brightness" of a hue. The second phenomenon is that people *prefer* colors when their brightness is distributed according to the natural brightness of the color's hue, meaning that people will prefer color combinations where the yellows are bright and the blues are dark [Asselbergs, I, 2007]. An example of this is shown in Figure 5. To formalize this, the colors in a palette are converted into two sets of grayscale values. The first is obtained by computing the natural brightness of each color's hue. The second is the *actual* L^* value of each color. A Spearman rank correlation is performed on the values, and this is referred to as the "Müllerness". A high value for this feature means that the colors are indeed ordered according to their natural brightness, a low value means the opposite.

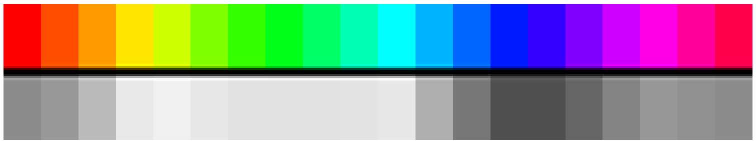


Figure 4: The top shows a series of colors where only the hue differs. The bottom shows the L^* value, indicating perceived brightness.

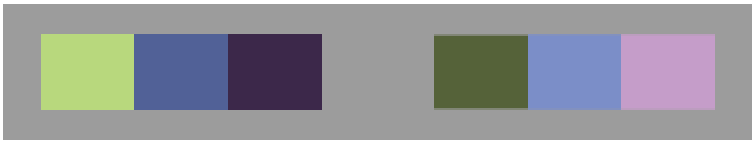


Figure 5: The two sets of colors have the same hues, only the brightness is different. Most people will find the color combination on the left more beautiful, because the purple and blue hues are dark, and the green hue is lighter. When the brightness values are distributed differently, the colors clash.

The following features can be extracted from a color palette. To understand the equations, note that the returned palette is represented as matrix P, with each column of P representing a color and P_n indicating the n^{th} color/column. Each column has 4 values: $[L^*,\ a^*,\ b^*,\ size]^T$, but instead of indexing $(P_{v,n})$, I will use the more convenient notation $L^*(P_n)\ \ldots\ size(P_n)$. I will also use this notation for values that can be derived from the color, e.g. $\text{luma}(P_n)$ and $\text{chroma}(P_n)$ to indicate the *luma* and *chroma* of the n^{th} color, respectively.

Mean Chroma Mean chroma indicates the overall "coloredness" of the palette. Though not identical, it is comparable to the *saturation* of a palette. The chroma of a color decreases as more black or white is added. The mean chroma is calculated by averaging the chroma values of the colors, weighing them by their relative prevalence in the palette.

$$\sum_{color \in P} [\text{chroma}(color) \cdot \text{size}(color)]$$

Mean Luma Similar to mean chroma, but instead uses the luma values. Represents the average lightness of the page.

$$\sum_{color \in P} [\text{luma}(color) \cdot \text{size}(color)]$$

Contrast Some palettes have tints that are similar in lightness, this makes the website appear soft and calm. If stark contrasts are used, the website will appear bold or loud. Contrast is defined as the variance in luma values, weighted by their size.

$$var(\text{luma}(P), \text{size}(P))$$

Balance Balance indicates whether the page uses light text on a dark background or dark text on a white background. It is calculated by subtracting the luma value of the second color from the luma value of the first color. We assume that the background is the most used color and the text color is second, which is true for most websites.

$$\text{luma}(P_1) - \text{luma}(P_2)$$

Weight Weight indicates the portion of the page that is not occupied by background. Heavier pages have less "whitespace". A page with a lot of whitespace appears minimalistic and is easy to read.

$$1 - \text{size}(P_1)$$

Hue Spread A high value for hue spread indicates that many complementary colors have been used. Red and yellow have a large difference in their hue value, creating a bold palette, while quiet palettes are characterized by low differences in hues. For each pair of colors in the palette, the closest circular distance between the hues is calculated. This distance is weighted by the color's chroma and summed for all pairs.

$$\sum_{\text{pairs}(a,b)} \Big| \big((\text{hue}(P_a) - \text{hue}(P_b) + 0.5) \bmod 1 \big) - 0.5 \Big| \cdot \text{chroma}(P_a) \cdot \text{chroma}(P_b)$$

a*b* Spread Similar to the hue spread. This feature is the *Euclidian* distance of each pair's $a^* b^*$-coordinates. Because the $a^* b^*$ coordinates of desaturated colors are close to zero, grey colors will have a low impact on this feature. Highly saturated contrasting colors will cause this feature to have a high value.

$$\sum_{pairs(a,b)} \sqrt{(a^*(P_a) - a^*(P_b))^2 + (b^*(P_a) - b^*(P_b))^2}$$

Müllerness How close the colors are to the Müller formula [Asselbergs, I, 2007]. This is formalized as the Spearman rank correlation between the actual luminescence (L^* value) and the natural brightness of a color with the same *hue* but full *chroma* and *intensity*.

3 EXPERIMENTS

In this report, we present the results of a number of different classification experiments. First, the experiments in [de Boer and van Someren, 2008] were repeated. Because these experiments were performed on a somewhat artificial dataset, two more experiments were performed on datasets with a more realistic distribution. The datasets used in each of the three experiments are described and the results of the classifications are shown in Sections 3.1 – 3.3. Then in Section 3.4 and 3.5, the results are examined further.

For each experiment all 45 features (shown in Table 1) were used to train a Naive Bayes Classifier. Classification accuracy was measured using 10-fold cross validation. A C4.5 tree-based classifier [Quinlan, 1993] was also used, but results are not reported since its performance was worse in every single experiment. In addition to using the Naive Bayes Classifier with all attributes, a second experiment was done using only the 5 best predicting features, as measured by their gain ratio. A large number of non-predictive features will cause "noise" leading to misclassifications, so using only the best features sometimes leads to better results.

```
P1_L
P1_a
P1_b
P2_L
P2_a
P2_b
P3_L
P3_a
P3_b
P4_L
P4_a
P4_b
P5_L
P5_a
P5_b
P6_L
P6_a
P6_b
num_colors
mean_chroma
contrast
balance
weight
hue_spread
ab_spread
mullerness
left_margin
page_width
num_columns
avg_col_size
max_col_size
min_col_size
first_col_size
col_irregularity
top_margin
page_height
num_rows
avg_row_size
max_row_size
min_row_size
banner_size
row_irregularity
photo_area
avg_photo_area
num_photos
```

Table 1: List of used features.

3.1 Experiment 1: Aesthetics, Recency, and Topic Datasets

To compare performance of the new algorithm to an existing one, the new feature extractors were applied to the same three datasets that were used in [de Boer and van Someren, 2008]. These three datasets are hand picked from different sources to ensure a good contrast in their appearance. The three sets are described below and the results of are shown in Table 2.

Aesthetics This datasets consists of 30 websites from “The World’s Ugliest Websites” [Andrade, 2009] and 30 websites from a design web log, listing the most beautiful web pages of 2008 [Design, 2008].

Recency The second dataset consist of 30 recent (2009) and 30 old (1999) websites, respectively taken from the highest ranked pages on Alexa.com[1] in 2009, and the most popular pages in 1999 from the Internet Archive[2].

Topic This set consist of 30 websites from each of the following categories: *newspaper*, *hotel*, *celebrity*, *conference*. Refer to [de Boer and van Someren, 2008] for the sources of each of these categories.

Aesthetics

		Predicted		
		Nice	Ugly	
Class	Nice	27	3	30
	Ugly	1	29	30
		28	32	60

Accuracy:	93%
Top 5 Features:	97%
Original Experiment:	80%

Recency

		Predicted		
		New	Old	
Class	New	25	5	30
	Old	3	27	30
		28	32	60

Accuracy:	87%
Top 5 Features:	93%
Original Experiment:	85%

Topic

		Predicted		
		Correctly	Wrong	
Class	Celebrity	18	12	30
	Conference	16	14	30
	Hotel	19	11	30
	News	24	6	30
		77	43	12

Accuracy:	64%
Top 5 Features:	58%
Original Experiment:	56%

Table 2: Results for the experiment that was repeated from [de Boer and van Someren, 2008]. The confusion matrix for each of the *aesthetics*, *recency* and *topic* test is shown, together with classification accuracy when using all features. Additionally, the accuracy reported in the original experiment is shown, and the accuracy when only the 5 best predicting features were used.

[1] `http://www.alexa.com`

[2] `http://www.archive.org`

3.2 Experiment 2: ECML/PKDD 2010 Discovery Challenge Dataset

In order to test performance in a more real-world scenario, the algorithm was tested on the dataset provided for the Discovery Challenge [Benczur et al., 2010]. This dataset is a crawl of the `.eu` domain. It contains URLs labeled with the categories they belong to, and some other attributes such as neutrality and trustworthiness. Some of the labels in this dataset are unrelated to visual features, and other labels have a very skewed class distribution. The remaining labels are useful for a classification experiment. These labels are "Commercial" and "Educational/Research". Again, all 45 features were used with a Naive Bayes classifier to obtain the results in Table 3.

Commercial versus Non Commercial

		Predicted		
		Commercial	Non Comm.	
Class	Commercial	280	269	549
	Non Comm.	257	438	695
		537	707	1244

Accuracy:	57%
Top 5 Features:	59%

Educational versus Non Educational

		Predicted		
		Educational	Non Edu.	
Class	Educational	202	268	470
	Non Edu.	174	600	774
		376	868	1244

Accuracy:	64%
Top 5 Features:	63%

Table 3: Results for the experiment on the ECML Discovery Challenge dataset. The numbers shown in the confusion matrix apply to the experiment with all features. The accuracy when using the 5 best features is shown separately.

3.3 Experiment 3: ODP Dataset

Another dataset was created with the goal of testing on a set that has more real-world data than de Boer's original data, and more variation in appearance than the Discovery Challenge dataset. 353 URLs were taken from the DMOZ Open Directory Project[3]. These URLs were taken from a number of different categories to ensure some variance in the desired attributes. The attributes were selected so that there was some relation to visual features, they are *aesthetics*, whether the site looks *modern*, and whether the website is *informative or commercial*. The distribution over different categories was as follows.

- Design (131)
  - `Computers/Internet/Web Design and Development/Designers/Freelance` (66)
  - `Computers/Internet/Web Design and Development/Designers/Full Service` (65)
- Education (147)
  - `Reference/Education` (88)
  - `Reference/Education/K through 12/Home Schooling` (59)

[3] http://rdf.dmoz.org/

- Kids (75)
  - `Kids and Teens/People and Society/Personal Homepages/By Teens` (75)

These categories were chosen because they make up a nicely varied dataset. In the "Design" category there are both professional designers with *beautiful* websites and amateur designers with less pretty sites, however, both categories are mostly *commercial*. The "Education" category contains a lot of university websites, usually with a *modern* appearance. Contrary to that, the "Home Schooling" category contains a lot of *informative* websites on how to home-school, most of which look quite *outdated*. The "Kids and Teens" category contains mostly websites that are downright *ugly*. The resulting URLs were rendered in a web browser and four people were asked to label them according to the following questions.

Aesthetics Judge whether the web page looks better than average. For "looks good", the page should look pleasant and polished. Pick "looks very good" if the website stands out.
(1) Looks bad. (2) Looks good. (3) Looks very good.

Recency Judge whether the page looks as if it was made in the last 3 years. Pick "looks old" if the website looks outdated. Pick "looks very modern" if the website stands out in this regard.
(1) Looks old. (2) Looks modern. (3) Looks very modern.

Commercial vs. Informative This question is about whether the website appears as if something is being sold. The opposite is a website that just supplies information. If a website looks like it provides useful information and sells products or services, pick "both".
(1) Commercial. (2) Both. (3) Informative.

The labels were translated into numerical ratings (1,2,3). These were then converted to standard scores for each participant. The standardized scores were averaged between participants, yielding a continuous score for each label. Finally, these scores were discretized into *two* equally-frequent classes per label. The results of the classification experiment are shown in Table 4.

Looks Good versus Looks Bad

		Predicted		
		Looks Bad	Looks Good	
Class	Looks Bad	122	58	180
	Looks Good	40	133	173
		162	191	353

Accuracy:	73%
Top 5 Features:	70%

Old versus Modern

		Predicted		
		Old	Modern	
Class	Old	112	66	178
	Modern	48	127	175
		160	193	353

Accuracy:	67%
Top 5 Features:	71%

Commercial versus Informative

		Predicted		
		Commercial	Informative	
Class	Commercial	101	76	177
	Informative	65	111	176
		166	187	353

Accuracy:	60%
Top 5 Features:	61%

Table 4: Results for the experiment on the new hand-labeled dataset. The confusion matrices for each experiment are shown. The results in the matrices are for the experiments using all features.

3.4 Comparison

A short comparison of results is given in this section. The results are summarized in Figure 6. A "majority" percentage was added as a baseline, it represents the size of the largest class in a dataset, and thus the accuracy if only prior probabilities would be used.

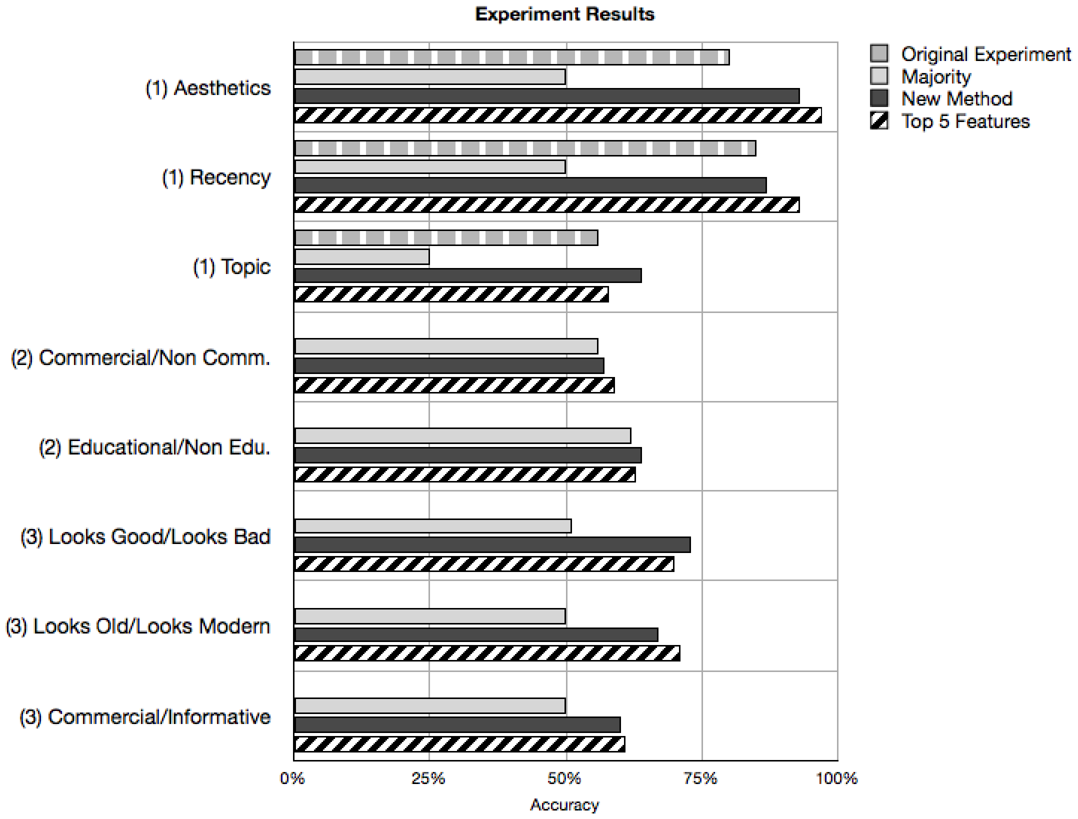


Figure 6: Results for each of the experiments.

In the first experiment, results were better than the previous experiment for each test. Accuracy was highest on the aesthetics task, which is not surprising since the new features described in this report are focused on characterizing web design. Also, accuracy on both the aesthetics and recency test improved when only the 5 best features were used, indicating that these classes are easily distinguished by a few features. The accuracy in the topic classification task dropped when only the best features were used, this might be explained by the fact that this task has four different classes. Each of these classes might be set apart by different features, so performance is better when all available information can be used.

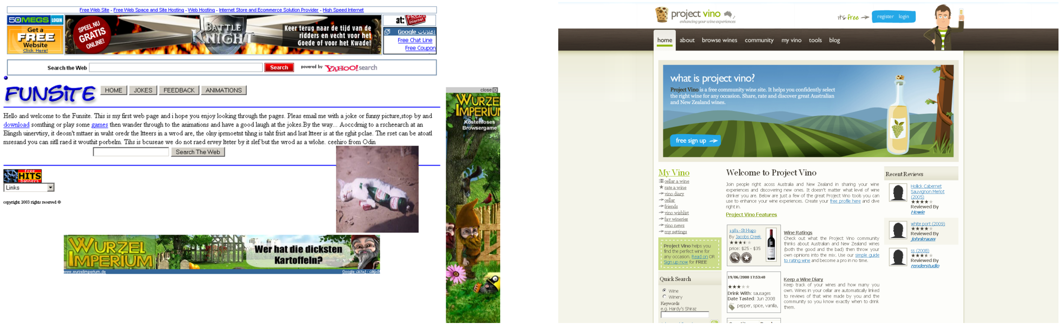
Accuracy was much lower in the second set of experiments, only just above the baseline. There are two possible reasons for this. First, whether a website is commercial and/or educational is less related to the appearance of the site. Second, the pages in this dataset are much more similar to each other, in looks and purpose.

The accuracy on the third experiment was somewhere in between the accuracy achieved in the previous experiments. They were ranked in the order you would expect given that visual feature extractors were used. Aesthetics are most related to visual features, followed by whether the website looks modern; whether a website is commercial is least related to its design, so this final task yielded the lowest accuracy. It is interesting to see that the *Looks Good versus Looks Bad* task has a lower accuracy when only the 5 best features are used, meaning that beauty in this task has more facets than in the first experiment's aesthetics task. The *Commercial versus Informative* test gave better results than the *Commercial versus Non-Commercial* test in Experiment 2. This might be explained by the fact that the ODP dataset has a more varied set of websites, while the appearance of classes was very similar in the Discovery Challenge dataset.

3.5 Relevant Features

In this paragraph I will offer an analysis of the results of the experiments. The new feature space makes it easy to interpret the probabilities assigned to each feature by the Naive Bayes Classifier. I will discuss and give examples of the features that led to good predictions in some of the experiments.

In the aesthetics experiment of the first dataset (Section 3.1), the *hue spread* feature was most significant. This feature indicates how many complementary colors are used. This is in line with our impression; the ugly websites tend to use a lot of different colors, some even literally use "every color of the rainbow". Beautiful websites use a subdued base color and a single main color. Misclassifications in this experiment originated from beautiful websites using a lot of smooth graphics. These graphics are not marked as photos, so their color contributes to the color palette, causing a higher hue spread. On the other end there were ugly websites using only few colors, we dislike the way they look because of their lack of good typography and chaotic placement of images. Examples of these misclassifications can be found in Figure 7.



Hello and welcome to the Funsite. This is my first web page and i hope you enjoy looking through the pages. Pleas email me with a joke or funny picture,stop by and download somthing or play some games then wander through to the animations and have a good laugh at the jokes.By the way... Aoccdrnig to a rscheearch at an Elingsh uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht frist and lsat ltteer is at the rghit pclae. The rset can be atoatl mseesand you can sitll raed it wouthit porbelm. Tihs is bcuseae we do not raed ervey lteter by it slef but the wrod as a wlohe. ceehiro from Odin

copyright 2003 rights reserved ©

Welcome to Project Vino

Join people right across Australia and New Zealand in sharing your wine experiences and discovering new ones. It doesn't matter what level of wine drinker you are. Below are just a few of the great Project Vino tools you can use to enhance your wine experiences. Create your free profile here and dive right in.

Project Vino is a free community wine site. It helps you confidently select the right wine for any occasion. Share, rate and discover great Australian and New Zealand wines.

Wine Ratings
Check out what the Project Vino community thinks about Australian and New Zealand wines (both the good and the bad) then throw your own opinions into the mix. Use our simple guide to rating wine and become a pro in no time.

Keep a Wine Diary
Keep track of your wines and how many you own. Wines in your cellar are automatically linked to reviews of that wine made by you and the community so you know exactly when to drink them.

Project Vino helps you find the perfect wine for any occasion. Read on OR Sign up now for FREE

Figure 7: The page on the left was wrongly classified as beautiful because photos are excluded from the color palette extraction. The page on the right was wrongly classified as ugly because the smooth graphic adds a lot of colors to the extracted palette.

In the recency experiment, the most important features were *photo area*, *left margin*, and *page width*. These can be traced back directly to the technological improvements over the last decade. Old web pages are more narrow because of the size and resolution of the computer screens available a decade ago. Larger screens have become available, up to a point where it is no longer in the benefit of readability to spread the website over the entire width of the screen. Because of this, designers started leaving a margin on the left of the page to center it horizontally on larger screens. Old pages also used less photos and graphics because the available bandwidth was more limited.

In the topic discrimination experiment the most important features varied per class. To distinguish celebrity pages, $L^*(P_1)$ (the background's lightness) is the most relevant feature. This is because most celebrity sites have a black background, making them look "glamorous". News websites stand out because they have a high *contrast*, almost always employing black text on a white background, mimicking an actual newspaper. Hotel and conference websites are not distinguished by a small set of features, that is probably why accuracy on this experiment was worse when using only the 5 best features.

Performance on the Discovery Challenge dataset (Section 3.2) was not good enough to be able to say anything about what features contributed to classification. Also, performance did not improve greatly when using only the 5 best features, this is an indication that none of the features stand out in terms of predictive value.

In the experiments on the ODP dataset (Section 3.3) accuracy was worse and correspondingly it was less clear which features make a big difference. The predictive power of features was generally smaller. In the "Looks Bad" versus "Looks Good" experiment, a^*b^**-spread* came out as the most important feature. This is a measure of the amount of complementary colors used, so it shows the tendency for ugly websites to use “every color of the rainbow” again.

Websites that “Looked Modern” were again set apart by their larger *page width* and *left margin*. It is interesting to note that the $a^*(P_1)$ and $b^*(P_1)$ features indicating the color of the background were significant here. In the first dataset (in the *recency* experiment) almost all of the *old* websites had a white background, indicated by low values for these features. However, in the “Looks Old” versus “Looks Modern” experiment, websites with a white background were labeled as more *modern*. An explanation for this could be that, although web designers have started using more color in their designs, which colors they use is subject to short-term fashion. A clean design with a white or grey background is less prone to looking outdated.

When distinguishing between *Commercial* and *Informative* websites, the *number of photos* was one of the best features. It makes sense that commercial websites display more photos to display products and “happy customers”. Informative websites present text, with a smaller amount of supporting images.

4 CONCLUSION AND DISCUSSION

The experiments have shown that it is effective to extract more high level design-related features from web pages in order to classify. The method can be used to distinguish between a number of classes of websites, and works well for classes that are closely related to the website’s appearance. Experiment 2 and 3 demonstrate that it is harder to classify websites according to broad categories such as ‘commercial’ or ‘education’. High accuracy on aesthetics tasks in Experiment 1 and 3 indicate that the new features are well suited for characterizing *beauty* in web design. Among the most important features were those that characterize the number of colors on the page, the size of the page, and the number of photos used. The method was an improvement over [de Boer and van Someren, 2008], where only low level texture and color features were used. It has the added benefit of producing a human-readable feature space, allowing for easy examination of classification results and insight into what constitutes good web design.

A practical application of the described methods could be a web design evaluation tool. If a classifier was trained on a larger dataset with good and bad design, it could be used to *grade* new pages. The tool could then return an overview of which features contributed to the grade, and how the page should be changed to get a higher grade. However, it remains to be seen if this method is by itself practically useful. Though results for each experiments were significantly better than random, a real-world application might need more than 8 out of 10 correct predictions. In order for performance to improve, the feature extractors have to be fine-tuned and new feature extractors might be necessary.

A continuation of this line of research could be the expansion of the repertoire of available feature extractors. A face-detection algorithm could be used to distinguish faces in photographs and determine whether they are portraits or group pictures. Some design elements could also be explicitly detected in a similar way (e.g. menu bars, page headings, thumbnail galleries). It will then be more important to select features that are appropriate for each task.

A second direction could consist of combining these visual features with features extracted from the HTML and CSS code, for a much more precise approach. For example, the method for color extraction described in this report still returns approximate matches in many cases. By restraining the available colors to those that occur in the CSS code, a palette that matches the page *exactly* is much easier to extract. Similar combinations can be made for estimating the size and position of columns on the page.

References

[Amento et al., 2000] Amento, B., Terveen, L., and Hill, W. (2000). Does “authority” mean quality? Predicting expert quality ratings of Web documents. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 296–303. ACM.

[Andrade, 2009] Andrade, L. (2009). The worlds ugliest websites!!! http://www.nikibrown.com/designoblog/2009/03/03/the-worlds-ugliest-websites/.

[Asselbergs, I, 2007] Asselbergs, I (2007). The Müller Formula. http://www.colourlovers.com/blog/2007/09/02/the-muller-formula-or-predictable-color-preferences/.

[Benczur et al., 2010] Benczur, A., Castillo, C., Erdelyi, M., Masanes, J., Matthews, M., and Z., G. (2010). ECML/PKDD 2010 Discovery Challenge Data Set. Crawled by the European Archive Foundation. http://www.ecmlpkdd2010.org/articles-mostra-2041-eng-discovery_challenge_2010.htm.

[Boulton, 2007] Boulton, M. (2007). Five Simple Steps to designing with colour. http://www.markboulton.co.uk/journal/comments/five-simple-steps-to-designing-with-colour.

[de Boer and van Someren, 2008] de Boer, V. and van Someren, M. (2008). Classifying Web Pages with Visual Features.

[Design, 2008] Design, C. (2008). 40 most beautiful and inspirational website designs of 2008. http://www.crazyleafdesign.com/blog/top-40-beautiful-and-inspirational-website-designs-of-2008/.

[Joblove and Greenberg, 1978] Joblove, G. and Greenberg, D. (1978). Color spaces for computer graphics. *ACM SIGGRAPH Computer Graphics*, 12(3):25.

[Kruger, 1994] Kruger, A. (1994). Median-cut color quantization. *Dr Dobb's Journal-Software Tools for the Professional Programmer*, 19(10):46–55.

[MacEvoy, 2005] MacEvoy, B. (2005). Modern Color Models: CIELAB. http://www.handprint.com/HP/WCL/color7.html#CIELAB.

[Müller, 1948] Müller, A. (1948). *Die Moderne Farbenharmonielehre*. Chromos Winterthur.

[Quinlan, 1993] Quinlan, J. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann.

[W3C, 2008] W3C (2008). Web content accessibility guidelines (wcag) 2.0. http://www.w3.org/TR/UNDERSTANDING-WCAG20/visual-audio-contrast-contrast.html.