# **Color Grading Feature Documentation**

## **1. Introduction**

The Color Grading feature enhances website aesthetics and user experience by evaluating color choices. Here’s why it matters:

* **Visual Appeal**: Well-chosen colors make your site visually engaging.
* **Branding**: Consistent colors reinforce brand identity.
* **Accessibility**: Proper contrast ensures readability for all users.
* **Emotional Impact**: Colors evoke feelings and influence behavior.

## **2. How It Works**

1. Color Extraction:
   * Scrapes website HTML and CSS to identify colors.
   * Detects primary, secondary, and accent colors.
2. Algorithmic Evaluation:
   * Custom algorithm assesses color scheme based on:
     + **Contrast**: Readability (text vs. background).
     + **Harmony**: Coherence of color combinations.
     + **Accessibility**: Compliance with standards.
     + **Emotional Associations**: Psychological impact.

## **3. Usage Guide**

1. Assessing a Website:
   * Input URL or upload HTML/CSS files.
   * Receive color score and detailed report.
2. Interpreting Results:
   * Understand color quality and recommendations.
   * Iterate based on feedback.

## **4. Work Distribution**

Timeline: 2 Weeks

* Backend & Algorithm : Animesh
* Frontend : Vidya
* Detailing & Designing : Ankita

## **5. Technical Elements**

## **6. References**

* [7 Rules For Choosing A Website Color Scheme | Elementor](https://elementor.com/blog/website-color-schemes/#)
  + Useful to show our theory is impactful for users.
* [Testing Web Design Color Contrast | Articles | web.dev](https://web.dev/articles/testing-web-design-color-contrast)
  + Article for color improvements in websites
* [6 Tips to Choose a Stunning Website Color Scheme (wordstream.com)](https://www.wordstream.com/blog/website-color-scheme)
  + For contrast and combination.
* [How to Choose Website Color Combination (eiosys.com)](https://www.eiosys.com/blog/website-color-combination/)
* [56 Website Color Schemes (With Examples) (websitesetup.org)](https://websitesetup.org/website-color-schemes/)
  + Situational color schemes. (May be useful for designing algorithm)
* [What Is Visual Hierarchy?](https://www.shutterstock.com/blog/establish-visual-hierarchy)
* <https://elementor.com/blog/how-to-design-effective-buttons-for-wordpress-websites/>
* [Color Tutor for websites](https://elementor.com/blog/how-to-design-effective-buttons-for-wordpress-websites/)
* [ACT Guidelines](https://www.w3.org/WAI/standards-guidelines/act/rules/afw4f7/proposed/) for Contrast
* [Impact of colors on marketing](https://www.emerald.com/insight/content/doi/10.1108/00251740610673332/full/html)

**Points to consider:**

* The primary colors are generally the more dominant colors in the site, accounting for background colors, logo colors, menu colors, etc., and secondary colors are often used as accent colors, among other use cases.
* Consistency is actually one of the fundamental values in creating a color scheme for your website. Because brand personality is so crucial to a successful website and business, having a consistent color palette solidifies your brand identity, as your repeated use of color and styling will create associations between your brand and your audience.
* it’s much easier to distinguish between the heading and the subheading when they are given two different colors. This way, the more “important” text is in a dark color, and the “less important” text is in a contrasting, lighter color.
* Instead, when choosing your color scheme, think about simplicity. A really complicated, busy color scheme often confuses the eye.
* To account for the 10% in the 60/30/10 rule, accent colors are mandatory and used sparingly throughout the website.

**ALGORITHM STRUCTURE**

### **1. Contrast**

* **Parameter**: Contrast Ratio
* **Description**: Measures the readability of text against its background.
* **Formula**:
* \text{Contrast Ratio} = \frac{L1 + 0.05}{L2 + 0.05}
* where (L1) and (L2) are the relative luminance of the lighter and darker colors, respectively.
* **Scoring**:
  + Score based on WCAG guidelines:
    - 3:1 for large text (minimum acceptable)
    - 4.5:1 for normal text (good)
    - 7:1 for enhanced readability (excellent)

### **2. Harmony**

* **Parameter**: Color Harmony Score
* **Description**: Evaluates the coherence of color combinations.
* **Method**: Use color theory principles (e.g., complementary, analogous, triadic schemes).
* **Scoring**:
  + Assign scores based on the harmony of color pairs or groups:
    - Complementary: High score
    - Analogous: Moderate score
    - Clashing colors: Low score

### **3. Accessibility**

* **Parameter**: Accessibility Compliance
* **Description**: Checks if the color scheme meets accessibility standards.
* **Method**: Ensure sufficient contrast for text and interactive elements.
* **Scoring**:
  + Compliance with WCAG 2.1:
    - AA level: Good score
    - AAA level: Excellent score
    - Non-compliance: Low score

### **4. Emotional Associations**

* **Parameter**: Emotional Impact Score
* **Description**: Evaluates the psychological impact of colors.
* **Method**: Use a predefined mapping of colors to emotions (e.g., blue for calm, red for excitement).
* **Scoring**:
  + Assign scores based on the desired emotional impact:
    - Positive emotions (aligned with brand): High score
    - Neutral emotions: Moderate score
    - Negative emotions: Low score

### **5. Overall Color Score**

* **Parameter**: Composite Score
* **Description**: Combines all individual scores into a final score.
* **Method**: Weighted average of all parameters.
* **Formula**:
* \text{Composite Score} = w\_1 \times \text{Contrast Score} + w\_2 \times \text{Harmony Score} + w\_3 \times \text{Accessibility Score} + w\_4 \times \text{Emotional Impact Score}
  + (w\_1, w\_2, w\_3, w\_4) are the weights assigned to each parameter based on their importance.

### **Example Calculation**

Suppose you have the following scores:

* Contrast Score: 8/10
* Harmony Score: 7/10
* Accessibility Score: 9/10
* Emotional Impact Score: 6/10

If you assign equal weights to all parameters ((w\_1 = w\_2 = w\_3 = w\_4 = 0.25)), the Composite Score would be:

\text{Composite Score} = 0.25 \times 8 + 0.25 \times 7 + 0.25 \times 9 + 0.25 \times 6 = 7.5

**Contrast Evaluation**

When evaluating contrast, it’s important to focus on specific elements rather than comparing all colors with each other.   
**1. Text vs. Background**

* **Primary Focus**: Ensure that text (headings, body text, links) has sufficient contrast against its background.
* **Method**: Calculate the contrast ratio between text color and background color using the WCAG formula.

### **2. Interactive Elements**

* **Buttons and Links**: Ensure buttons and links are easily distinguishable from their surroundings.
* **Method**: Check the contrast between button/link text and button/link background, as well as the button/link background against the page background.

### **3. Form Elements**

* **Input Fields**: Ensure input fields (text boxes, dropdowns) have sufficient contrast for both the field itself and the placeholder text.
* **Method**: Evaluate the contrast between input field borders and background, as well as input text and field background.

### **4. Icons and Graphics**

* **Icons**: Ensure icons are clearly visible against their background.
* **Method**: Calculate the contrast ratio between icon color and background color.

### **5. Focus Indicators**

* **Keyboard Navigation**: Ensure focus indicators (outlines, borders) are clearly visible when navigating with a keyboard.
* **Method**: Check the contrast between focus indicator color and the element’s background.

### **Example Implementation**

Here’s a simplified example of how we might implement contrast checking for text vs. background in Python:

from colorsys import rgb\_to\_hsv

def luminance(r, g, b):

a = [v / 255.0 for v in (r, g, b)]

a = [v / 12.92 if v <= 0.03928 else ((v + 0.055) / 1.055) \*\* 2.4 for v in a]

return 0.2126 \* a[0] + 0.7152 \* a[1] + 0.0722 \* a[2]

def contrast\_ratio(color1, color2):

L1 = luminance(\*color1)

L2 = luminance(\*color2)

if L1 > L2:

return (L1 + 0.05) / (L2 + 0.05)

else:

return (L2 + 0.05) / (L1 + 0.05)

# Example colors in RGB

text\_color = (255, 255, 255) # White

background\_color = (0, 0, 0) # Black

ratio = contrast\_ratio(text\_color, background\_color)

print(f"Contrast Ratio: {ratio:.2f}")

### **Prioritizing Elements**

* **High Priority**: Text vs. background, buttons, and links.
* **Medium Priority**: Form elements and icons.
* **Low Priority**: Decorative elements and non-essential graphics.