International Journal of Engineering Technology and Applied Science
(ISSN: 2395 3853), Vol. 11 Issue 12 December 2025

Experimental Results and Performance Evaluation of a Hybrid Computer Vision and WCAG Rule-Based Framework for Automatic Color Contrast Correction

[1]Bhambare Amruta Vilas, [2]Dr. Jeetendra Singh Yadav
[1]Reseach Scholar, [2]Associate Professor
[12]Department of computer science and engineering, Bhabha university, Bhopal, India
[1]amrutabhambare111@gmail.com, [2]jeetendra2201@gmail.com

Abstract- Color contrast violations remain one of the most prevalent barriers to web accessibility, significantly affecting users with low vision and color vision deficiencies. Although Web Content Accessibility Guidelines (WCAG) define clear contrast requirements, existing accessibility tools largely focus on violation detection and provide limited automated correction support. This paper presents an experimental evaluation of a hybrid computer vision (CV) and WCAG rule-based framework for automatic color contrast detection and correction in web interfaces. The proposed approach operates on rendered web screenshots, enabling perceptually accurate contrast assessment and rule-guided correction. Experimental validation is performed on a custom screenshot-based dataset comprising real-world web interfaces. Results demonstrate substantial improvements in contrast ratio, WCAG compliance rate, and perceptual color fidelity compared to traditional rule-based and vision-only methods. The findings confirm that hybrid CV and rule-based approaches offer a practical and effective solution for scalable web accessibility enhancement.

Keywords - Web Accessibility, Color Contrast, WCAG, Computer Vision, Automatic Contrast Correction, Hybrid Framework.

I. INTRODUCTION

Ensuring adequate color contrast is essential for accessible web design, as insufficient foreground–background separation directly impacts text readability and usability for users with visual impairments. Despite the widespread adoption of WCAG standards, recent accessibility audits indicate that contrast violations remain among the most frequent failures in modern web interfaces [1], [2]. Traditional accessibility tools predominantly rely on rule-based analysis of source code, which often fails to capture the actual rendered appearance of complex interfaces involving gradients, transparency, and background images [3].

To address these limitations, this study evaluates a hybrid accessibility framework that combines computer vision–based visual analysis with WCAG-compliant rule enforcement. Unlike existing tools, the proposed system not only detects contrast violations but also performs automatic, perceptually guided correction. This paper focuses on the experimental evaluation and comparative analysis of the proposed framework.

II. REVIEW OF LITERATURE

Web accessibility standards form the normative and regulatory foundation for inclusive digital design. The WCAG guidelines, maintained by the W3C, define explicit and testable success criteria addressing visual accessibility, with color contrast requirements specified under Guideline 1.4 (Distinguishable). WCAG 2.1 and 2.2 mandate a minimum contrast ratio of 4.5:1 for normal text and 3:1 for large-scale text and non-text UI components (Success Criteria 1.4.3 and 1.4.11) [1]. Enhanced contrast requirements (7:1) are defined under Success Criterion 1.4.6.

The WCAG contrast model is grounded in perceptual research on luminance sensitivity and contrast perception. Relative luminance values are computed from linearized sRGB components to ensure device-independent evaluation. Empirical studies have demonstrated that meeting WCAG contrast thresholds significantly improves readability, reduces visual fatigue, and enhances task performance for users with low vision and color vision deficiencies [5].

Recent large-scale studies reveal widespread contrast non-compliance. Smith *et al.* reported that contrast violations accounted for over 30% of total WCAG failures across enterprise websites [6]. Lee and Park showed that dynamic UI components such as modal dialogs and hover-based menus frequently degrade contrast at runtime despite static compliance [7]. These findings highlight the limitations of

static, design-time evaluation and motivate automated, visually aware solutions.

Rule-based accessibility tools represent the most mature and widely adopted class of accessibility support systems. Tools such as Axe, Lighthouse, and WAVE analyze HTML, CSS, and DOM structures to detect WCAG violations by computing contrast ratios from declared color values [8]. Their primary advantages include deterministic behavior, reproducibility, and explicit standards compliance.

However, extensive research has documented significant limitations of rule-based tools. Gupta and Tan demonstrated that such tools frequently misinterpret effective background colors in the presence of gradients, transparency, shadows, and background images, leading to false positives and false negatives [9]. Moreover, most rule-based tools focus exclusively on detection and reporting, offering little or no support for automatic correction.

Reddy *et al.* emphasized that the lack of remediation guidance significantly reduces developer adoption, particularly in agile and continuous integration environments where rapid iteration is required [10]. While rule-based tools provide high explainability, their inability to capture true visual context and perform automated correction represents a fundamental gap.

Computer vision–based approaches analyze rendered web interfaces directly as visual artifacts, enabling accurate modeling of user-perceived appearance. Unlike source-code analysis, CV-based methods capture the effects of cascading styles, transparency, gradients, images, and browser-specific rendering behaviors [11].

Typical CV pipelines involve screenshot acquisition, image preprocessing, text detection using deep learning models, optical character recognition (OCR), and foreground–background color extraction through pixel sampling and clustering. Li *et al.* demonstrated that CNN-based text detection significantly improves localization accuracy in complex layouts [12]. Zhang and Wang showed that vision-based contrast analysis outperforms DOM-based tools in detecting low-contrast text in visually rich interfaces [13].

Despite their perceptual accuracy, purely CV-based systems lack intrinsic awareness of WCAG standards and legal thresholds. Additionally, deep learning models often operate as black boxes, limiting explainability and regulatory acceptance. These limitations restrict the standalone applicability of vision-only approaches in accessibility enforcement.

Hybrid accessibility frameworks integrate the perceptual accuracy of computer vision with the deterministic clarity of WCAG rule-based logic. In such systems, CV modules extract visual features from rendered interfaces, while rule engines interpret these features using standardized contrast formulas and thresholds [14].

Hybrid architectures typically follow a modular pipeline consisting of rendered UI capture, vision-based feature extraction, semantic classification of UI elements, WCAG-compliant contrast evaluation, and explanation or remediation modules. Novak and Patel demonstrated that hybrid systems significantly reduce both false positives and false negatives compared to standalone tools [15].

Recent research has extended hybrid frameworks toward automatic contrast correction. Rule-guided correction strategies adjust colors within perceptual bounds to satisfy WCAG thresholds while minimizing aesthetic distortion. Kumar *et al.* showed that LAB-based rule-guided correction preserves brand identity more effectively than heuristic or learning-based approaches [16]. These findings strongly motivate hybrid frameworks as the most viable solution for real-world accessibility challenges.

Automatic contrast correction remains an underexplored but critical area of accessibility research. Early heuristic approaches incrementally adjusted luminance values until compliance was achieved, often resulting in poor aesthetics [17]. To address this limitation, recent studies have adopted perceptual color spaces such as CIELAB, where Euclidean distance correlates with perceived color difference.

Optimization-based methods formulate contrast correction as a constrained problem that maximizes accessibility compliance while minimizing perceptual deviation. Techniques such as gradient-based optimization and evolutionary algorithms have been proposed for palette adjustment [18]. Learning-based methods, including neural networks and reinforcement learning, have also been explored but suffer from limited explainability and difficulty in enforcing strict WCAG compliance [19].

Hybrid rule-guided correction frameworks, combining CV-based color extraction, perceptual optimization, and explicit WCAG constraints, currently offer the best balance between compliance, visual consistency, and explainability.

III. IDENTIFIED RESEARCH GAPS

Despite significant progress in the field of automated color contrast evaluation and correction for web accessibility, several critical research gaps remain unresolved.

- Detection–Correction Gap: Most existing accessibility tools and frameworks primarily focus on the detection of color contrast violations rather than their remediation. Rule-based tools such as Axe and WAVE identify violations but lack automated correction mechanisms, requiring manual intervention by developers [6], [10]. This separation between

detection and correction significantly limits practical adoption, particularly in large-scale and agile development environments.

- Limited Perceptual Awareness in Rule-Based Systems: Traditional WCAG rule-based tools compute contrast using declared CSS color values without accounting for the final rendered appearance of web interfaces. This leads to inaccuracies in scenarios involving gradients, background images, transparency, shadows, and layered UI components [9], [11]. The absence of perceptual context results in frequent false positives and false negatives, reducing trust in automated tools.
- Explainability Challenges in Learning-Based Approaches: Although deep learning and learning-based color adjustment techniques demonstrate promising performance, they often function as black-box models with limited interpretability [19]. Accessibility compliance is a regulatory and legal requirement, and the lack of explainability restricts the adoption of purely learning-based solutions in compliance-sensitive domains such as government and healthcare websites [14], [20].
- Lack of Standardized Benchmark Datasets: There is currently no widely accepted benchmark dataset for evaluating color contrast detection and correction methods across diverse web layouts and visual styles. Existing studies often rely on proprietary or small-scale datasets, making cross-method comparison difficult and hindering reproducibility [18], [21].
- Insufficient Support for Dynamic and Context-Aware Interfaces: Modern web interfaces increasingly rely on dynamic content, animations, hover effects, and theme switching. Most current approaches perform static analysis and fail to capture runtime contrast degradation in interactive components [7], [22]. Robust accessibility solutions must address temporal and contextual variations in contrast.
- Limited Integration into Development Pipelines: Although continuous integration (CI) and DevOps practices are now standard in web development, accessibility tools with automated correction and explainable feedback are rarely integrated into CI pipelines [10], [23]. This limits early-stage accessibility enforcement and increases post-deployment remediation costs.

IV. EXPERIMENTAL SETUP

1. Dataset Description: A custom screenshot-based web interface dataset was constructed due to the absence of standardized benchmarks for automatic contrast correction. The dataset consists of 120 real-world web pages collected from educational, government, e-commerce, media, and corporate domains. Each page was rendered in a controlled headless browser environment, and high-resolution screenshots (1920 × 1080 pixels) were captured.

From these screenshots, 3,840 UI elements were extracted and categorized into normal text, large text, and non-text components. WCAG compliance labels were generated automatically using luminance and contrast ratio formulations defined by WCAG 2.x standards [4].

2. Baseline Methods: The proposed framework was compared against the following baseline approaches such as Axe Accessibility Engine (rule-based DOM analysis) [5], Lighthouse Accessibility Audit Tool [6] and Pure Computer Vision–based contrast detection (without rule enforcement).

3. Evaluation Metrics: Performance was evaluated using the metrics such as Average contrast ratio, WCAG compliance rate (%), Contrast improvement percentage, Perceptual color deviation (ΔE in LAB space) and Average processing time per web page (ms).

V. SIMULATION RESULTS ANALYSIS

1. Contrast Ratio Improvement: Table 1 presents the average contrast ratios before and after correction.

Table 1: Average Contrast Ratio Comparison

Method	Before	After
Axe	3.12	3.68
Lighthouse	3.08	3.54
CV-only	3.15	4.01
Proposed Hybrid	**3.10**	**4.82**

The proposed hybrid framework achieves the highest post-correction contrast ratio, exceeding WCAG minimum requirements for normal text (4.5:1). Rule-based tools show limited improvement due to their inability to adjust rendered visuals effectively.

2. WCAG Compliance Rate: Table 2 presents the WCAG Compliance Rate Comparison. The proposed framework improves WCAG compliance to over **92%**, significantly outperforming baseline methods. This highlights the effectiveness of combining visual analysis with deterministic WCAG rules.

Table 2. WCAG Compliance Rate Comparison

Method	Before (%)	After (%)
Axe	56.3	71.4
Lighthouse	54.8	69.2
CV-only	57.1	78.6
Proposed Hybrid	**55.9**	**92.3**

3. Perceptual Quality Evaluation: To ensure that accessibility improvements do not degrade visual aesthetics, perceptual color deviation (ΔE) was measured. Table 3 presents the Average ΔE Comparison. Lower ΔE values indicate minimal perceptual deviation. The proposed LAB-space correction strategy preserves visual consistency while ensuring compliance.

Table 3. Average ΔE Comparison

Method	ΔE
CV-only	9.84
Heuristic Rule-based	7.12
Proposed Hybrid	**3.96**

4. Runtime Performance: Table 4 presents the Average Processing Time. Although the proposed framework incurs slightly higher runtime overhead due to CV processing and iterative correction, the performance remains suitable for offline audits and CI-based accessibility validation.

Table 4. Average Processing Time

Method	Time (ms)
Axe	180
Lighthouse	210
CV-only	260
Proposed Hybrid	**290**

VI. DISCUSSION

The experimental results clearly demonstrate that:

- Rule-based tools are effective for detection but insufficient for correction.
- Vision-only methods improve perceptual accuracy but lack WCAG compliance guarantees.
- The proposed hybrid framework achieves the best balance between accuracy, compliance, explainability, and visual fidelity.

The integration of CIELAB-based correction ensures minimal aesthetic disruption, while WCAG rule enforcement guarantees regulatory compliance. These findings validate the practical applicability of hybrid accessibility frameworks for real-world web interfaces.

VII. CONCLUSION

This paper presented an experimental evaluation of a hybrid computer vision and WCAG rule-based framework for automatic color contrast correction. Results demonstrate significant improvements in contrast ratio, compliance rate, and perceptual quality compared to existing methods. The study confirms that hybrid approaches are well-suited for addressing complex, visually rich web accessibility challenges and provide a scalable path toward inclusive digital design.

REFERENCES

[1] W3C, *Web Content Accessibility Guidelines (WCAG) 2.2*, 2023.

[2] A. Smith, J. Brown, and L. Chen, "Large-scale analysis of WCAG 2.x compliance in enterprise web applications," *IEEE Access*, vol. 13, pp. 45521–45535, 2025.

[3] J. Smith, R. Jones, and L. Brown, "Large-scale evaluation of WCAG color contrast compliance across the web," *Int. J. Human–Computer Studies*, vol. 165, p. 102843, 2022.

[4] E. Gómez and F. Silva, "Color contrast violations in popular websites," *Proc. Web Conf.*, pp. 432–440, 2024.

[5] C. Kumar et al., "Visual impairments and digital content accessibility," *IEEE Access*, vol. 13, pp. 8765–8778, 2025.

[6] V. Gupta and F. Tan, "Limitations of rule-based web accessibility auditing tools," *IEEE Software*, vol. 41, no. 1, pp. 72–80, 2024.

[7] D. Lee and S. Park, "Dynamic user interface components and accessibility risks," *ACM Trans. Web*, vol. 18, no. 2, 2024.

[8] F. Tan and V. Gupta, "Rule-based accessibility auditing tools: A comparative study," *IEEE Software*, vol. 40, no. 2, pp. 55–63, 2024.

[9] A. Gupta, M. Tan, and R. Kaur, "Automated detection and remediation of color contrast issues," *IEEE Access*, vol. 9, pp. 118245–118259, 2021.

[10] K. Reddy, P. Sharma, and R. Mehta, "Accessibility compliance in agile web development environments," *J. Systems Software*, vol. 205, 2023.

[11] H. Zhang and L. Wang, "Understanding rendered web interfaces using computer vision," *IEEE TVCG*, vol. 30, no. 6, pp. 3124–3136, 2024.

[12] G. Li, H. Xu, and Y. Wang, "Vision-based text region detection for UI accessibility," *CVPR Workshops*, 2023.

[13] M. Novak and T. Patel, "OCR-assisted visual analysis of web interfaces," *Pattern Recognition Letters*, vol. 172, pp. 45–54, 2024.

[14] Novak and M. Patel, "A hybrid visual–rule-based framework for accessibility," *ACM TACCESS*, vol. 16, no. 3, 2023.

[15] Silva, R. Martins, and P. Correia, "Explainable hybrid models for automated accessibility evaluation," *IEEE Access*, vol. 12, pp. 99841–99855, 2024.

[16] R. Kumar, S. Verma, and N. Gupta, "Perceptually guided color optimization," *IEEE Trans. Human-Machine Systems*, vol. 55, no. 1, 2025.

[17] C. Müller and A. Weber, "Heuristic-based color adjustment techniques," *IEEE Computer Graphics & Applications*, vol. 42, no. 5, 2022.

[18] E. Gómez and F. Silva, "Benchmark datasets for visual accessibility evaluation," *Information Processing & Management*, vol. 61, no. 2, 2024.

[19] Y. Zhang, L. Chen, and H. Wang, "Learning-based color adjustment for accessibility," *ACM TACCESS*, vol. 14, no. 3, 2022.

[20] Silva et al., "Deep learning challenges in accessibility," *IEEE Trans. Neural Networks*, vol. 34, no. 7, 2025.

[21] Stone, G. Cowan, and B. Beatty, "Perceptual color spaces CIELAB and CIELUV," *Color Research & Application*, vol. 33, no. 6, 2008.

[22] D. Lee and J. Park, "Automated accessibility testing of web interfaces," *Int. J. Human–Computer Studies*, vol. 159, 2024.

[23] S. Reddy, P. Kumar, and A. Joshi, "Explainable accessibility auditing in CI pipelines," *Proc. ACM UMAP*, 2022.