

SAFE-COLOR: Color Fidelity Benchmarks and Thresholds for Safety-Critical Object Detection

Marvin Damschen, Ramana Reddy Avula, Mazen Mohamad

Dependable Transport Systems, RISE Research Institutes of Sweden, 501 15 Borås, Sweden

Email: {marvin.damschen, ramana.reddy.avula, mazen.mohamad}@ri.se

Abstract—Color fidelity is often overlooked in simulation-based validation for autonomous vehicles, yet even minor color mismatches can undermine the reliability of AI-driven perception systems. In this paper, we systematically examine how controlled deviations in color reproduction—quantified by ΔE —affect object detection accuracy across 32 variants of YOLO. Using a Macbeth ColorChecker, we derive calibrations for key color transforms (brightness, contrast, hue, gamma, saturation and color bias) and apply these to the COCO validation set. Our evaluations demonstrate that increasing ΔE yields significant drops in detection metrics, especially for safety-critical categories such as pedestrians and cyclists. Based on these findings, we propose ΔE thresholds that define acceptable color fidelity in camera simulations (e.g., $\Delta E \leq 3$ for $\Delta mAP \leq 1\%$). Furthermore, we contribute these transformed datasets and scripts as a publicly available benchmark, enabling reproducible comparisons and guiding future research on color-based vulnerabilities in automated driving and other safety-critical domains.

Index Terms—Color Fidelity, Object Detection, Autonomous Vehicles, Simulation-Based Validation, Safety-Critical Systems

I. INTRODUCTION

The deployment of autonomous systems in safety-critical domains—ranging from self-driving cars to industrial robotics—has accelerated the use of simulation-based verification and validation (V&V) frameworks [1, 2]. High-fidelity simulations enable rigorous and cost-effective testing of perception algorithms, especially object detection models, across diverse operational conditions. However, sensor fidelity remains a critical challenge: even minor inconsistencies between simulated and real-world sensor outputs can degrade model performance and raise concerns about the validity of tests [3]. Recognizing this, the *New Assessment/Test Method for Automated Driving* (NATM) [4] and related initiatives stress the need for assessing *Accuracy* and *Correctness* of simulated aspects such as geometry, illumination, and particularly color reproduction.

Although camera models in simulation have received attention, current industrial standards rarely define explicit thresholds for color accuracy [4, 5]. Color fidelity is vital for perception reliability, yet many evaluation efforts focus on high-level performance metrics—e.g., mean average precision (mAP)—without isolating the influence of color deviations on object detection performance. This gap complicates the reproducibility of simulation tests and undermines their correlation with real-world reliability.

In this paper, we address this gap by proposing a systematic approach to determine quantitative thresholds for color accu-

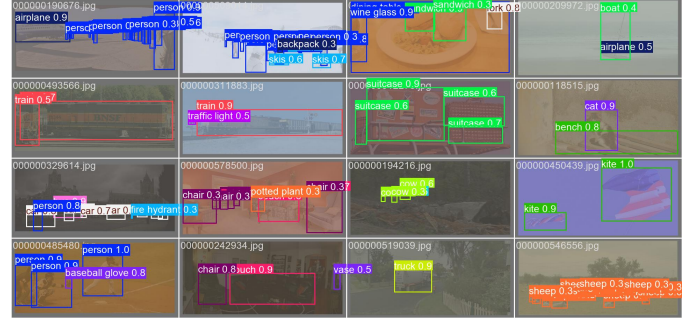


Fig. 1. 6 color transformations and 32 YOLO model variants are evaluated for establishing ΔE thresholds. An excerpt of results of YOLO11x on images transformed with “Pull Mean” at $\Delta E = 21$ is shown above.

racy, measured via ΔE , the industry standard for measuring perceptual color difference. Through controlled color transformations of established benchmark images (COCO 2017 [6]), we examine how varied ΔE levels affect YOLO-based object detection. Our empirical findings link specific ΔE values to quantifiable changes in detection performance, thereby offering a foundation for standardized sensor-fidelity assessment protocols in simulation. The key contributions of this work are as follows:

- *Systematic Evaluation of Object Detection Performance:* We analyze 32 YOLO variants under controlled color deviations, offering comprehensive insights into sensor fidelity impacts (see example in Fig. 1).
- *Benchmark transformations and Dataset:* Suite of COCO validation images modified according to a set of benchmark color transformations at distinct ΔE levels, forming a unique dataset for reproducible research on color fidelity in object detection¹.
- *Definition of Color Accuracy Thresholds:* By measuring detection performance across varying ΔE levels, we propose benchmarks to guide acceptable simulation fidelity.

This work lays a foundation for establishing rigorous color-fidelity benchmarks, bridging a critical gap in simulation-based validation frameworks for safety-critical autonomy.

II. BACKGROUND AND RELATED WORK

The development of Automated Driving and Advanced Driver Assistance Systems (AD/ADAS) requires extensive

¹<https://doi.org/10.5281/zenodo.14864429>

<https://github.com/RISE-Dependable-Transport-Systems/SAFE-COLOR>

testing due to the huge and diverse number of scenarios that need to be validated. While physical testing remains essential, it is fundamentally impractical to exhaustively cover all scenarios within real-world time constraints – estimates suggest hundreds of years would be required to achieve sufficient coverage for certification [7]. To address this, virtual testing has become a well-established practice in the automotive and mobile machinery domains, where simulation is utilized to cover the test scenarios needed for AD/ADAS V&V [8]. However, simulation and virtual testing present several challenges, particularly the need to ensure that the simulated environment and sensors achieve an acceptable fidelity level to represent the real-world environment and conditions [9].

Recent regulatory and standardization efforts, including UNECE’s New Assessment/Test Method for Automated Driving (NATM) Guidelines [4] and *SAE International’s comprehensive approach for the validation of virtual testing toolchains* [10], have recognized the importance of ensuring the reliability of simulation models and the credibility of virtual toolchains used for simulation-based V&V. These methodologies emphasize traceable alignment between model requirements and scenario-specific test objectives, necessitating rigorous verification via quantifiable metrics.

However, in many cases, these methodologies do not specify the exact metrics to be used. For example, NATM requires evidence of calculation verification as part of the models and simulation analysis for credibility assessment of virtual toolchains. This involves estimating numerical errors affecting models and simulations. While Annex III - Appendix 3 in NATM [11] provides examples of tools for validating different components, such as using the MacBeth chart [12] to determine the color space of a camera model, it often leaves the choice of specific metrics to the implementers.

Color Fidelity Metrics and Impact on Object Detection

Object detection is a critical feature in AD/ADAS systems. Hence, a significant amount of tests are devoted to ensure that this feature is reliable. An important advancement in the field of computer vision for real-time object detection is YOLO (You Only Look Once) [13, 14], which has shown its ability to deliver accurate and timely object detection efficiently. To evaluate relevant components like cameras, multiple metrics can be used to measure color accuracy. In this work, we utilize the ΔE_{00} metric (denoted as ΔE in the following), the updated CIEDE2000 color-difference formula [15] to ensure high-fidelity assessments.

Recent studies have highlighted the degradation in perception performance due to realistic environmental influences (e.g., raining conditions) or image corruptions [16, 17]. The important role of color cues in enhancing object detection performance was particularly studied and highlighted. Wang et al. [18] demonstrate that color information substantially aids pedestrian detection, particularly in complex backgrounds where geometric features alone may be insufficient. Similarly, Singh et al. [19] reveal that Convolutional Neural Networks (CNNs) leveraged by YOLO models are highly sensitive to

color variations, with color discrepancies leading to notable declines in detection accuracy. These findings underscore the necessity of maintaining high color fidelity within simulation environments to ensure robust object detection and real-world representative results.

Despite the acknowledged importance of color fidelity in object detection, existing research has not adequately explored the direct relationship between ΔE metrics and object detection performance. Specifically, there is a lack of empirical studies that establish quantitative ΔE thresholds correlating with acceptable levels of detection accuracy degradation. Addressing this gap is the focus of our research.

III. METHODOLOGY

Our methodology systematically assesses the impact of color accuracy deviations, measured using the ΔE metric, on object detection performance. While various linear and non-linear color transformations can achieve a specific ΔE value, in this work, we focus on a set of single-parameter transformations to ensure controlled and interpretable modifications to color accuracy. In the following, we elaborate on the three main steps of our methodology: selection and calibration of transformations, creation of the dataset, and assessment of object detection performance.

A. Selection and Calibration of Transformations

We propose a set of color transformations characterized by a single parameter, enabling controlled adjustments to obtain and evaluate a specific ΔE value:

- Brightness: Uniform modification of pixel intensity.
- Contrast: Scaling differences between pixel intensities and their mean.
- Saturation: Modification of color intensity to make colors appear more vivid or muted.
- Gamma: Non-linear mapping of pixel values to adjust image brightness.
- Hue: Alteration of color tones by rotating hue channel.
- Pull Mean: Adjustment of colors toward the image’s mean color to simulate color bias (see Fig. 1).

Brightness and contrast changes often arise from exposure differences and sensor hardware variations, while gamma alterations mirror lens or hardware color-response nonlinearity. Saturation and hue shifts can represent inaccurate white-balancing or color-filter array variations. Table I shows the sRGB-to-sRGB formulas of the single-parameter transformations. We denote an sRGB pixel by $\mathbf{p} = (p_r, p_g, p_b) \in [0, 1]^3$. The notation $\text{clip}(\cdot)$ represents channel-wise clamping to $[0, 1]$, RGB2HSV and HSV2RGB denote standard RGB $\leftrightarrow$ HSV conversions, and RGB2Lab and Lab2RGB are standard conversions between sRGB and Lab (D65). Additionally, linearize_srgb and delinearize_srgb refer to the piecewise gamma definitions in the sRGB standard. Negative and positive parameters are evaluated for brightness, contrast, saturation and gamma each. This results in a total of 10 transformations evaluated for each ΔE target value, denoted by ΔE_{target} (see Table II).

TABLE I
SINGLE-PARAMETER COLOR TRANSFORMATIONS IN SRGB.

Brightness ($\mathbf{p}, \beta$) = $\text{clip}(\mathbf{p} + \beta)$
Contrast ($\mathbf{p}, c$) = $\text{clip}(0.5 + c(\mathbf{p} - 0.5))$
Saturation ($\mathbf{p}, s$) = $\text{HSV2RGB}(H, sS, V)$, where $(H, S, V) = \text{RGB2HSV}(\mathbf{p})$.
Gamma ($\mathbf{p}, \gamma$) = $\text{clip}(\text{delinearize_srgb}([\text{linearize_srgb}(\mathbf{p})]^\gamma))$
Hue ($\mathbf{p}, h$) = $\text{HSV2RGB}((H + \frac{h}{360}) \bmod 1, S, V)$, where $(H, S, V) = \text{RGB2HSV}(\mathbf{p})$.
Pull Mean ($\mathbf{p}, \alpha$) = $\text{Lab2RGB}(\overline{\text{Lab}} + \alpha(\text{RGB2Lab}(\mathbf{p}) - \overline{\text{Lab}}))$, where $\overline{\text{Lab}}$ is the mean Lab color of the image.

For a given transformation on all pixels of the image with a single parameter $f(\text{img}, x)$, we first calibrate x using a Macbeth ColorChecker chart. The Macbeth chart is an industry standard to measure the color accuracy of natural objects as captured by cameras [12]. For a given ΔE_{target} value and transformation f , we find $x_{f, \Delta E_{\text{target}}}$ such that:

$$\Delta E \left(f \left(\begin{array}{c} \text{Macbeth Chart} \\ \text{Image} \end{array}, x_{f, \Delta E_{\text{target}}} \right), \begin{array}{c} \text{Macbeth Chart} \\ \text{Image} \end{array} \right) = \Delta E_{\text{target}}$$

B. Creation of the Dataset

The transformations $f(\text{img}, x_{f, \Delta E_{\text{target}}})$ are applied to all 5000 images of the COCO 2017 validation dataset [6]. This results in 10 subsets for evaluation of ΔE_{target} (one for each transformation, $x_{f, \Delta E_{\text{target}}}$ is fixed for a given f and ΔE_{target}).

COCO was selected for its extensive diversity in object categories, complex real-world scenarios, and established use as a benchmark in object detection research. This diversity ensures that performance metrics are representative of a wide range of applications, making the findings broadly applicable to safety-critical systems and comparable to other results in the literature. COCO images are JPEG-compressed, which means that opening and storing the images can have an impact on object detection performance if compression parameters are not kept. We minimize this effect by storing images using the original images' quantization tables during compression. To further account for its impact on detection performance, we treat the operation of opening and saving the image without modifications as the baseline in our evaluation in Section IV, denoting it with $\Delta E = 0$.

C. Assessment of Object Detection Performance

The study employed 32 YOLO models through the Ultralytics Python package [20], spanning multiple versions (YOLOv3u [21], YOLOv5u [22], YOLOv8 [23], YOLOv9 [24], YOLOv10 [25], and YOLO11 [20]) and sizes (from 2.1M parameters for YOLOv9t to 155.5M parameters for YOLOv5x6u) to ensure a comprehensive and generalizable analysis. The models were used as trained by the respective authors without modification. The inclusion of multiple architectures and model sizes allows us to evaluate the sensitivity of color fidelity across different design choices, ranging from

lightweight, real-time models to large-scale, high-accuracy variants. This diversity ensures that our findings are applicable to a wide range of practical applications, from embedded systems to high-performance computing environments in safety-critical domains. YOLO was selected for its prominence in real-time object detection tasks, renowned balance of accuracy and computational efficiency, and its established deployment in a variety of safety-critical applications such as autonomous driving. These attributes make YOLO ideal for studying the effects of color fidelity, as its performance under altered conditions has direct implications for real-world system reliability. We do, however, consider the threat to external validity due to limiting the experiment on YOLO. This is mitigated by the fact that YOLO models are widely recognized and used both in industry and academic research, and that we use a set of 32 different YOLO variants.

Results were generated using custom Python scripts that employed Ultralytics 8.3.66 in "validation" mode. The scripts were run on an Intel Core i9-13900KF machine with an NVIDIA GeForce RTX 4090 and 64 GB of RAM. We generated results for $\Delta E = 0$ to 12 in increments of 1 as our main area of interest. Results for $\Delta E_{\text{target}} = 13$ to 51 were generated in increments of 3. Only results until $\Delta E = 45$ are discussed in the following, as none of the transforms was able to obtain a ΔE above 47.77 (decreasing brightness) as discussed in the next section. Obtaining all results took approx. 120 hours.

We use mean average precision (mAP) to quantify overall detection accuracy. mAP is a widely adopted metric, as it provides a balanced measure of both precision and recall across different confidence thresholds and Intersection over Union (IoU) levels. It is ideal for safety-critical systems because it ensures correct detections and consistent accuracy across all classes and conditions. Unlike single-point accuracy measures, mAP evaluates model performance over a range of IoU thresholds, ensuring robustness against varying object scales, occlusions, and background complexities. We employ a variant of mAP that represents the mean value of Average Precision (AP) across all detection classes, where AP is calculated using the detected bounding boxes and 10 IoU thresholds, ranging from 50% to 95% in 5% steps, commonly reported as AP@50:5:95 [26]. In our study, mAP is computed over the 80 COCO object categories and key safety-critical classes, such as "person" and "bicycle" that are discussed below, to assess detection performance degradation under color distortions, enabling the derivation of quantitative thresholds for acceptable color fidelity.

IV. EXPERIMENTAL RESULTS

Table II presents the parameter values for the 10 benchmark transformations discussed in Section III-A. Parameters are determined by minimizing the difference between ΔE_{target} , and ΔE obtained on MacBeth after a given transformation using Brent's method as implemented in SciPy's `minimize_scalar` function [27]. The parameter ranges searched for each transform are as follows: Brightness (β) – [-1.0, 1.0], Contrast

TABLE II
PARAMETER VALUES FOR EACH TRANSFORMATION AND ΔE_{TARGET} . MAX. OBTAINED ΔE DIFFERENCE TO $\Delta E_{\text{TARGET}} \leq 0.3\%$.

$f \setminus \Delta E_{\text{target}}$	1	2	3	4	5	6	7	8	9	10	11	12	15	18	21	24	27	30	33	36	39	42	45
brightness-dec	-0.0093	-0.0220	-0.0343	-0.0465	-0.0587	-0.0709	-0.0832	-0.0953	-0.1074	-0.1193	-0.1317	-0.1441	-0.1811	-0.2188	-0.2577	-0.2973	-0.3378	-0.3810	-0.4268	-0.4791	-0.5405	-0.6077	-0.7001
brightness-inc	0.0095	0.0223	0.0352	0.0479	0.0607	0.0734	0.0865	0.0996	0.1126	0.1259	0.1390	0.1522	0.1925	0.2354	0.2839	0.3350	0.3900	0.4532	0.5276	0.6225	0.7694	-	-
contrast-dec	0.9535	0.8921	0.8340	0.7783	0.7249	0.6733	0.6234	0.5753	0.5291	0.4845	0.4415	0.3999	0.2837	0.1791	0.0845	0.0011	-	-	-	-	-	-	-
contrast-inc	1.0442	1.1081	1.1776	1.2508	1.3291	1.4103	1.4958	1.5859	1.6834	1.8002	1.9514	2.1454	3.0593	5.0655	14.4311	-	-	-	-	-	-	-	-
gamma-dec	0.9707	0.9328	0.8966	0.8621	0.8287	0.7961	0.7646	0.7341	0.7042	0.6753	0.6469	0.6191	0.5395	0.4635	0.3906	0.3206	0.2535	0.1890	0.1247	0.0633	-	-	-
gamma-inc	1.0288	1.0691	1.1101	1.1523	1.1958	1.2406	1.2871	1.3355	1.3857	1.4382	1.4933	1.5509	1.7412	1.9598	2.2120	2.5088	2.8723	3.3394	3.9717	4.9201	6.3604	9.6009	-
hue	1.4364	3.3818	5.3371	7.3022	9.3238	11.3517	13.3927	15.5292	17.6992	19.9119	22.1867	24.5929	32.6264	42.0117	-47.6028	-57.2558	-68.2895	-81.9459	-99.5016	-119.1750	-144.1176	-	-
pull_mean	0.9635	0.9139	0.8653	0.8179	0.7716	0.7263	0.6821	0.6388	0.5964	0.5553	0.5150	0.4758	0.3636	0.2593	0.1586	0.0584	-	-	-	-	-	-	-
saturation-dec	0.9476	0.8810	0.8186	0.7593	0.7023	0.6476	0.5946	0.5436	0.4943	0.4464	0.4001	0.3547	0.2268	0.1099	0.0057	-	-	-	-	-	-	-	-
saturation-inc	1.0545	1.1345	1.2236	1.3359	1.4989	1.6968	1.9930	2.5098	5.8104	12.7169	21.8924	35.0254	-	-	-	-	-	-	-	-	-	-	-

(c) – [0.0, 50.0], Saturation (s) – [0.0, 50.0], Gamma (γ) – [0.01, 10.0], Hue (h) – [-180, 180], Pull Mean (α) – [0.0, 1.0]. The maximum ΔE difference for the reported parameters is below 0.3%. As seen in Table II, the maximum achievable ΔE differs between transformations (e.g., extreme contrast reduction yields a completely gray image with $\Delta E \approx 26$ on MacBeth). Using these calibrated parameters, we generated data subsets from the COCO validation images and evaluated 32 YOLO models on each subset. The results provide insight into how different color fidelities, measured by ΔE , affect object detection performance in a safety-critical context.

A. Object Detection Performance

Fig. 2 shows the detection performance of YOLO11x when evaluated on COCO validation images subjected to the 10 benchmark transformations at varying ΔE . As shown, among all the transformations the increase and decrease in contrast affect the detection performance the most, while hue adjustments have limited impact even at high ΔE . High contrast exaggerates edges and object boundaries that YOLO relies on for bounding-box proposals. Conversely, low contrast might obscure critical object boundaries, causing bounding-box detection to degrade. In comparison, hue changes primarily alter color tones without significantly affecting the structural features essential for detection. A similar effect is observed across all 32 models, which highlights the importance of the contrast parameter for object detection using YOLO models. At $\Delta E = 24$, reducing contrast results in an almost plain gray image and mAP drops to zero.

B. Worst-Case Performance Degradation and ΔE Thresholds

To explore thresholds on color fidelity requirements of camera sensor models, we identify the *worst-case* transformation (either contrast increase / decrease or saturation increase) at each ΔE until $\Delta E = 24$, and measure its effect on detection performance. Figures 3 and 4 compare the detection performance of all 32 models under their worst-case transformation at each ΔE . They present the overall mAP and its degradation (ΔmAP) across all 80 detection classes, alongside the AP and its degradation (ΔAP) for the safety-critical “person” and “bicycle” classes. The results indicate that while the AP for the “person” detection class is initially higher than the mAP across all 80 detection classes at $\Delta E = 0$, its ΔAP increases slightly more compared to mAP as ΔE grows. Similarly, the ΔAP for the “bicycle” detection class exceeds the ΔmAP for all

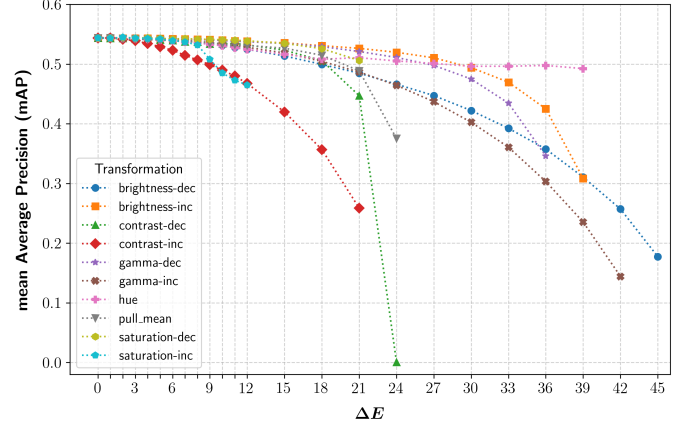


Fig. 2. Detection performance (mAP) of YOLO11x evaluated on data subsets created by applying different transformations on the COCO validation images.

80 classes as ΔE increases until $\Delta E = 15$, indicating certain object categories may be more sensitive to color fidelity.

Fig. 4 acts as a practical guideline to specify the ΔE threshold for a given acceptable ΔmAP and further illustrates performance degradation (in log scale), categorizing YOLO models as Nano, Small, Medium, Large, or Extra Large. Each category exhibits an approximately exponential increase in the loss of mAP as ΔE grows. While larger models (e.g., “Extra Large”) generally have a higher performance in mAP than smaller ones, they follow a similar trajectory and, too, converge toward zero performance beyond $\Delta E \approx 24$. This aligns with earlier observations that even state-of-the-art networks are ultimately limited by excessive color distortions.

Notably, at $\Delta E \leq 6$ all models retain competitive performance with $\Delta \text{mAP} < 3\%$. For average-case scenarios, one could set $\Delta E = 3$ or $\Delta E = 12$ as the upper limit to keep ΔmAP below 1% or 10%, respectively, including a safety margin. For safety-critical scenarios such as detecting persons, a ΔmAP below 1% is also achievable with $\Delta E \leq 3$, while for detecting bicycles, this is only possible with a strict constraint of $\Delta E \leq 1$ (see Fig. 4 (b) and (c)). In practice, $\Delta E \leq 1$ can be difficult to achieve for a camera sensor model, however, Fig. 4 (c) shows that the ΔE threshold can be increased linearly with each percentage point ΔAP increase that can be tolerated between $\Delta E = 1$ and $\Delta E = 11$. Thus, for safety-critical scenarios a threshold between $\Delta E = 1$ and $\Delta E = 11$ would result in a ΔAP below 1% and 11%, respectively.

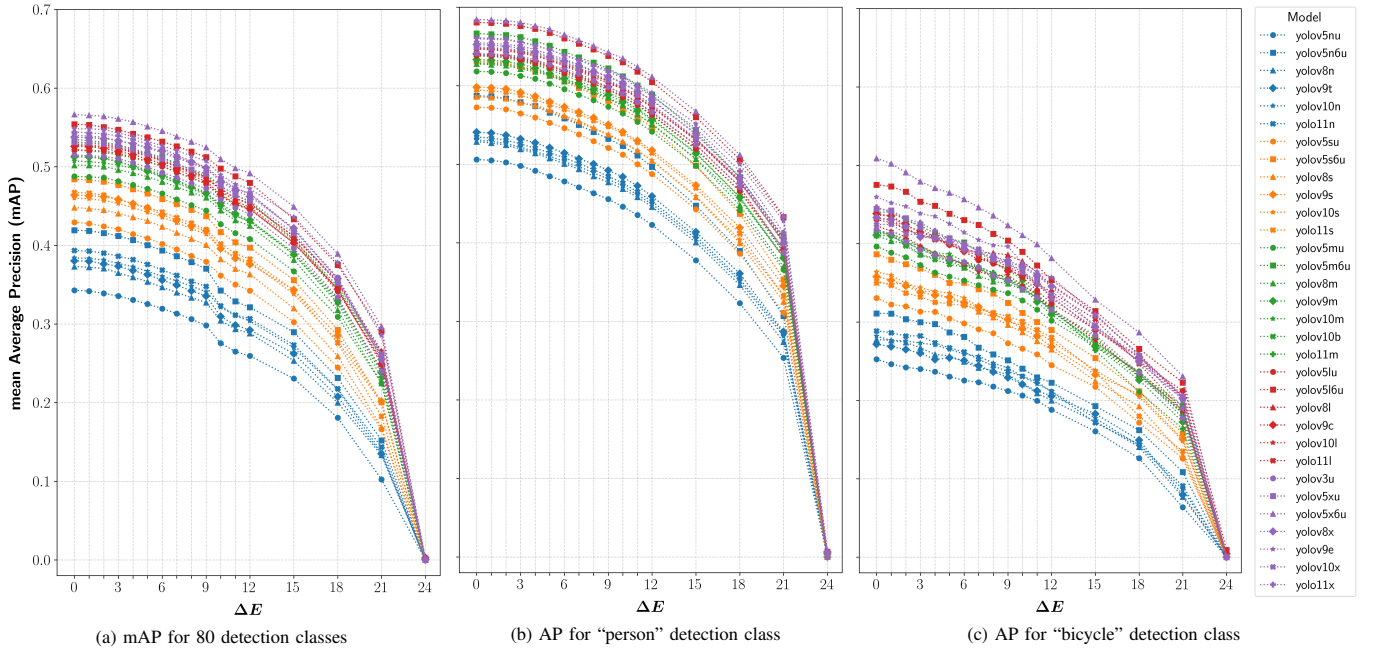


Fig. 3. Detection performance of different models with worst-case transformation at each ΔE .

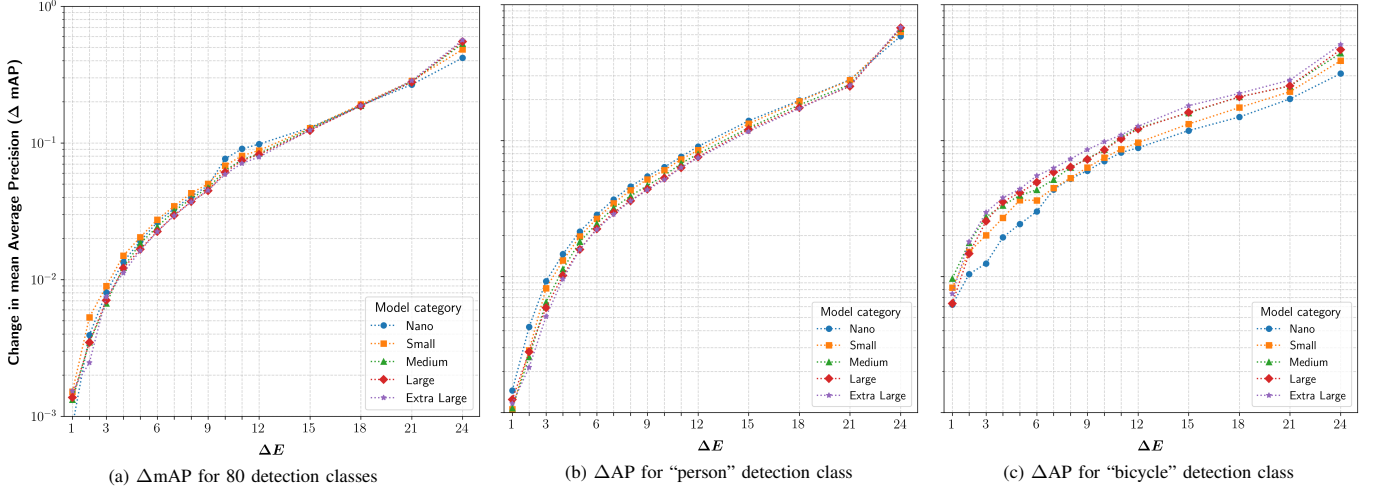


Fig. 4. Detection performance degradation with worst-case transformation compared to performance on original COCO, worst case per model category.

V. DISCUSSION

This study underscores the critical role of color fidelity in safety-critical object detection, particularly for simulation-based validation. While YOLO models exhibit robustness to minor color transformations, significant deviations in ΔE lead to substantial performance losses, especially for safety-critical classes such as “person” and “bicycle”. The rapid decline in mAP with increasing ΔE highlights the necessity for strict fidelity requirements in simulated camera models. Our findings provide concrete thresholds for color accuracy in simulation environments. While geometry and illumination consistency are often prioritized, color fidelity also plays a crucial role in detection performance. We show that maintaining $\Delta E \leq 3$ is necessary to limit performance loss to under 1% for average-case scenarios, while for safety-critical applications, particularly in detecting persons and other vulnerable road users,

$\Delta E \leq 1$ is required to achieve the same performance loss threshold. Developers of simulation frameworks can use these benchmarks to improve sensor model fidelity, ensuring that virtual validation accurately reflects real-world conditions.

Despite the robustness of our methodology, certain limitations must be considered. Our transformations apply globally to images, whereas real-world color distortions are often spatially variant due to lens artifacts, illumination shifts, or localized sensor noise. Future work should explore spatially non-uniform transformations to assess whether localized color shifts produce similar degradation trends.

It is equally important to note that other image quality metrics—such as sharpness—warrant a similarly rigorous treatment. Validating sensor performance in terms of color fidelity only is not sufficient as other aspects like sharpness, noise, and dynamic range can affect object detection performance. For example, inadequate sharpness can result in blurred details that

directly affect object localization and classification, leading to an increased rate of false negatives in critical scenarios. Future work should integrate objective sharpness metrics and establish quantitative thresholds to ensure that image degradation due to blurring remains within acceptable limits. Such efforts, together with our proposed ΔE thresholds, will further bridge the gap between synthetic simulation environments and real-world sensor performance in safety-critical applications.

VI. CONCLUSION

In this work, we introduced SAFE-COLOR, a methodology and benchmark aimed at quantifying the role of color fidelity in safety-critical object detection for autonomous systems. By systematically applying calibrated transformations—validated via a Macbeth ColorChecker—to the COCO validation set, we demonstrated that even moderate deviations in ΔE can severely degrade detection accuracy, particularly for vulnerable classes such as pedestrians and cyclists. Our investigation of worst-case performance degradation across 32 YOLO models allowed us to establish quantitative ΔE thresholds that maintain acceptable detection performance, even as image quality degrades. Moreover, our analysis highlights that color fidelity is only one component of overall sensor performance. Drawing on recommendations from NATM and SAE, we emphasize the importance of defining thresholds for additional metrics such as image sharpness, noise, and dynamic range. Incorporating such metrics into simulation-based validation frameworks will further enhance the reliability of sensor models and, by extension, the safety of autonomous systems. SAFE-COLOR contributes publicly available datasets and transformation scripts to enable reproducible research, offering actionable guidance for simulation tool developers, regulatory bodies, and end-users.

ACKNOWLEDGMENT

AGRARSENSE is supported by the Chips JU and its members, including top-up funding from Sweden, Czechia, Finland, Ireland, Italy, Latvia, Netherlands, Norway, Poland and Spain (Grant Agreement No. 101095835).

REFERENCES

- [1] N. Rajabli, F. Flammini, R. Nardone, and V. Vittorini, "Software verification and validation of safe autonomous cars: A systematic literature review," *IEEE Access*, vol. 9, pp. 4797–4819, 2020.
- [2] J. A. Agirre, L. Etxeberria, R. Barbosa, S. Basagiannis, G. Giantamidis, T. Bauer, E. Ferrari, M. L. Esnaola, V. Orani, J. Öberg, *et al.*, "The valu3s ecsel project: Verification and validation of automated systems safety and security," *Microprocessors and microsystems*, vol. 87, p. 104349, 2021.
- [3] A. Ngo, M. P. Bauer, and M. Resch, "A multi-layered approach for measuring the simulation-to-reality gap of radar perception for autonomous driving," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, IEEE, 2021, pp. 4008–4014.
- [4] W. P. on Automated/Autonomous and C. Vehicles, "New Assessment/Test Method for Automated Driving (NATM) Guidelines for Validating Automated Driving System (ADS)," Economic Commission for Europe (ECE), Apr. 6, 2023. [Online]. Available: <https://unece.org/sites/default/files/2023-04/ECE-TRANS-WP.29-2023-44e.pdf>.
- [5] C. B. D. R. G. MC, G. W. S. C. T. F. and V. S., "Interpretation of eu regulation 2022/1426 on the type approval of automated driving systems," no. KJ-NA-31-842-EN-N (online), 2024, ISSN: 1831-9424 (online). DOI: 10.2760/86028(online).
- [6] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland*, Springer, 2014, pp. 740–755.
- [7] N. Kalra and S. M. Paddock, "Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability?" *Trans. Res. Part A: Policy and Practice*, vol. 94, pp. 182–193, 2016.
- [8] J. Fadaie, "The state of modeling, simulation, and data utilization within industry: An autonomous vehicles perspective," *arXiv preprint arXiv:1910.06075*, 2019.
- [9] S. Sagmeister, P. Kounatidis, S. Goblirsch, and M. Lienkamp, "Analyzing the impact of simulation fidelity on the evaluation of autonomous driving motion control," in *2024 IEEE Intelligent Vehicles Symposium (IV)*, IEEE, 2024, pp. 230–237.
- [10] I. A. for Mobility Testing Standardization, "IAMTS Best Practice for A Comprehensive Approach for the Validation of Virtual Testing Toolchains," SAE Industry Technologies Consortia, IAMTS0001202104, Apr. 1, 2021, Accessed: 2024-10-21. [Online]. Available: <https://www.sae.org/standards/content/iamts0001202104/>.
- [11] W. P. on Automated/Autonomous and C. Vehicles, "Proposal for a second iteration of the New Assessment/Test Method for Automated Driving - Master Document," Economic Commission for Europe (ECE), Apr. 12, 2022. [Online]. Available: https://unece.org/sites/default/files/2024-03/ECE_TRANS_WP.29_2022_57e.pdf.
- [12] C. S. McCamy, H. Marcus, J. G. Davidson, *et al.*, "A color-rendition chart," *J. App. Photog. Eng.*, vol. 2, no. 3, pp. 95–99, 1976.
- [13] F. Sultana, A. Sufian, and P. Dutta, "A review of object detection models based on convolutional neural network," *Intelligent computing: image processing based applications*, pp. 1–16, 2020.
- [14] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, "A review of Yolo algorithm developments," *Procedia computer science*, vol. 199, pp. 1066–1073, 2022.
- [15] G. Sharma, W. Wu, and E. N. Dalal, "The CIEDE2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations," *Color Research & Application*, vol. 30, no. 1, pp. 21–30, 2005. DOI: <https://doi.org/10.1002/col.20070>.
- [16] G. Volk, S. Müller, A. Von Bernuth, D. Hospach, and O. Bringmann, "Towards robust cnn-based object detection through augmentation with synthetic rain variations," in *2019 IEEE intelligent transportation systems conference (ITSC)*, IEEE, 2019, pp. 285–292.
- [17] E. Rusak, L. Schott, R. S. Zimmermann, J. Bitterwolf, O. Bringmann, M. Bethge, and W. Brendel, "A simple way to make neural networks robust against diverse image corruptions," in *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16*, Springer, 2020, pp. 53–69.
- [18] J. S. Jin, C. Xu, M. Xu, Q. Wang, J. Pang, L. Qin, S. Jiang, and Q. Huang, "Justifying the importance of color cues in object detection: A case study on pedestrian," in *The Era of Interactive Media*, Springer, 2013, pp. 387–397.
- [19] A. Singh, A. Bay, and A. Mirabile, "Assessing the importance of colours for cnns in object recognition," *arXiv preprint arXiv:2012.06917*, 2020.
- [20] G. Jocher and J. Qiu, *Ultralytics YOLO11*, version 11.0.0, 2024. [Online]. Available: <https://github.com/ultralytics/ultralytics>.
- [21] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *arXiv preprint arXiv:1804.02767*, 2018.
- [22] G. Jocher, *Ultralytics YOLOv5*, 2020. DOI: 10.5281/zenodo.3908559. [Online]. Available: <https://github.com/ultralytics/yolov5>.
- [23] G. Jocher, A. Chaurasia, and J. Qiu, *Ultralytics YOLOv8*, version 8.0.0, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>.
- [24] C.-Y. Wang and H.-Y. M. Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information," 2024.
- [25] A. Wang, H. Chen, L. Liu, K. Chen, Z. Lin, J. Han, and G. Ding, "YOLOv10: Real-Time End-to-End Object Detection," *arXiv preprint arXiv:2405.14458*, 2024.
- [26] R. Padilla, S. L. Netto, and E. A. B. da Silva, "A survey on performance metrics for object-detection algorithms," in *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, 2020, pp. 237–242. DOI: 10.1109/IWSSIP48289.2020.9145130.
- [27] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, *et al.*, "Scipy 1.0: Fundamental algorithms for scientific computing in python," *Nature methods*, vol. 17, no. 3, pp. 261–272, 2020.