JOURNAL OF GREEN COMPLEX ENGINEERING, 2026, Vol.4, No.1,61~68
https://doi.org/10.59810/greenplexresearch.v4i1.284
e-ISSN 3025-9673

Turning Turnitin Highlights into Structured Data with Report-Aware HSV Segmentation and GPU-Enabled OCR

Mutiara Yusuf*, Rizki Yusliana Bakti, Titin Wahyuni

Informatics Engineering Program, Faculty of Engineering, Universitas Muhammadiyah Makassar, Makassar, Indonesia

ABSTRACT

Turnitin similarity reports contain rich spatial, color, and textual information, yet highlighted passages are usually inspected visually rather than converted into structured data. This study develops a report-aware pipeline that transforms color-coded Turnitin highlights into analyzable records. The method combines PDF rendering, page-type triage, hue–saturation–value (HSV) segmentation, morphological processing, connected-component analysis, geometric filtering, and GPU-enabled optical character recognition (OCR), with PDF text-layer inspection as supporting information. The evaluation corpus comprised 106 reports and 3,057 pages. Page triage identified 1,315 highlight-bearing pages and routed 56.98% of pages away from detailed processing. HSV segmentation produced eight color classes and 11,471 pre-filter components. After filtering and text extraction, 7,530 structured records were retained; 76.75% satisfied the predefined good-readability criterion, 20.04% were very short fragments, and 3.21% were noisy, garbled, or empty. Document workload was strongly right-skewed. The results establish systems-level feasibility for auditable extraction without treating textual similarity as automatic evidence of plagiarism or the readability indicator as OCR accuracy.

ARTICLE HISTORY

Received June 18, 2026
Received in revised form July 19, 2026
Accepted August 20, 2026
Available online August 27, 2026.

KEYWORDS

Turnitin, HSV color segmentation, optical character recognition, document analysis, similarity analytics

1. Introduction

Academic integrity is increasingly mediated by digital systems, but a similarity score is not a self-interpreting measure of misconduct. Contemporary reviews distinguish direct copying from paraphrased, semantic, cross-language, and other forms of reuse [1], while institutional responses are strongest when detection is combined with awareness, monitoring, and evaluation [2]. AI-supported assessment adds another layer: lecturers value automation and analytics but also raise concerns about ethical governance, privacy, bias, and preservation of professional judgment [3]. For computational research, the implication is clear: similarity evidence may be automated, but its interpretation must remain separable from a misconduct decision.

Most plagiarism-related computing studies focus on comparing content rather than parsing the visual structure of an already generated report. OCR can recover image-borne text that would otherwise be unavailable to text-only similarity systems [4-6], while document-analysis research shows that reliable extraction depends on layout interpretation, region localization, and structured parsing rather than a single recognition call [7,8]. These studies address a necessary problem—making heterogeneous document content machine readable—but they usually begin with raw documents and then estimate similarity. The present study starts at a different point in the workflow: a commercial similarity report has already been generated, and the objective is to convert its visual evidence into structured data.

A Turnitin report contains more than an overall percentage. Its pages encode the location of matching passages, match-group colors, compact source identifiers, and the textual context surrounding each highlight. In routine use, these elements are inspected manually, one page and one passage at a time. Manual review is suitable for individual decisions, but it does not naturally support cohort-level analytics, reproducible auditing, or construction of paragraph-level datasets. The technical gap is therefore one of document engineering: how can color, geometry, page context, and text be recovered from a multi-page report while preserving provenance?

Several studies establish the relevant building blocks. Image-borne text inside scientific PDFs can be extracted with OCR so that it does not escape similarity checking [9], and image processing can expose character-injection strategies that interfere with text-based document comparison [10]. Scientific-document analysis likewise emphasizes layout-aware decomposition, metadata localization, and structured extraction [11-13]. These findings motivate a staged architecture in which irrelevant pages are routed away early, color regions are localized, non-target graphics are filtered geometrically, and only then is OCR applied.

The closest methodological precedent is highlighted-text extraction itself. Lee et al. [14] demonstrated that HSV segmentation combined with OCR can recover Korean-English highlighted text and evaluated the pipeline using intersection-over-union and character-level recognition metrics. This evidence establishes that colored highlights are computationally recoverable. The novelty requirement for the present study is therefore narrower and more defensible: rather than claiming a new combination of HSV and OCR, this work extends highlighted-text extraction to a report-aware, multi-page parsing problem that includes page triage, eight match-group color classes, source-ID

CONTACT: Mutiara Yusuf 105841114722@student.unismuh.ac.id Informatics Engineering Program, Faculty of Engineering, Universitas Muhammadiyah Makassar, Makassar, Indonesia

rejection, record provenance, and highly heterogeneous document workloads.

HSV is attractive because Turnitin's markup uses color as a visual code and because hue can be separated from brightness. Nevertheless, color-space selection is task dependent. Comparative image-segmentation studies show that RGB, Lab, HSV, and other representations can behave differently under different algorithms and visual conditions [15-17]. Adaptive segmentation further illustrates why fixed thresholds may lose robustness when appearance changes [18]. For controlled report rendering, a lightweight and interpretable HSV pipeline is therefore plausible, but its thresholds should not be assumed to generalize automatically across interfaces, export modes, compression conditions, or scanned copies.

Once extracted, structured text can feed later similarity analysis. Recent reviews and implementations span lexical matching, TF-IDF and cosine similarity, embedding-based document comparison, and hybrid semantic approaches [19-22]. These methods solve a different stage of the problem from the present study. Here, lexical or semantic comparison is positioned after extraction: the contribution is the upstream acquisition layer that converts visually highlighted report content into structured, provenance-preserving text records.

The manuscript therefore defines its contribution as auditable data acquisition, not automated plagiarism adjudication. This tighter positioning avoids overstating the meaning of a Turnitin flag and distinguishes the engineering achievement—recovering structured evidence—from the later normative decision about whether a passage is acceptable.

The study addresses four objectives: (1) quantify page-level workload that can be routed away before expensive processing; (2) characterize the HSV color-component distribution; (3) assess the usability of extracted text using a transparent readability taxonomy; and (4) quantify document-level workload heterogeneity relevant to scalable deployment. The working hypothesis is that a report-aware pipeline can transform a large proportion of visual similarity annotations into readable, provenance-preserving records while avoiding unnecessary processing of non-highlight pages.

2. Methods

2.1 Study design and data source

The evaluative corpus comprised 106 Turnitin PDF reports generated for student proposals and undergraduate theses in an Informatics Study Program at an Indonesian university. Together, the reports contained 3,057 pages. The PDFs included native or rendered document text, color-coded similarity highlights, source-ID elements, and report-specific graphics. The document was therefore treated as a multimodal artifact: the text layer carried lexical content, while the rendered page carried spatial and color information that plain-text extraction would not preserve.

Analytical units differed across stages. The document-level denominator was 106 reports; page classification used 3,057 pages; HSV processing generated 11,471 pre-filter color components; and post-filter extraction yielded 7,530 structured highlight records. These denominators were kept separate because a pre-filter connected component is not equivalent to a final text record, and each page can contain multiple components and multiple records. The source dataset available for this manuscript provided aggregate outputs rather than manually annotated masks or line-by-line transcriptions. Consequently, no unreported precision, recall, F1-score, IoU, character error rate (CER), or word error rate (WER) values were inferred. Because the corpus contains student-generated material and similarity-report metadata, this article reports aggregate results and does not reproduce student names, document titles, or extracted passage text; raw reports are not distributed with the manuscript.

2.2 Pipeline architecture

Figure 1 summarizes the workflow. Each PDF page was rendered at approximately 200 dpi to establish a consistent image representation for color analysis while preserving the original PDF for text-layer inspection. A page-type triage stage first identified pages likely to contain relevant highlights. Candidate pages were converted from RGB to HSV and pixels were assigned to one of eight empirically calibrated classes: red, orange, yellow, green, cyan, blue, purple, and pink.

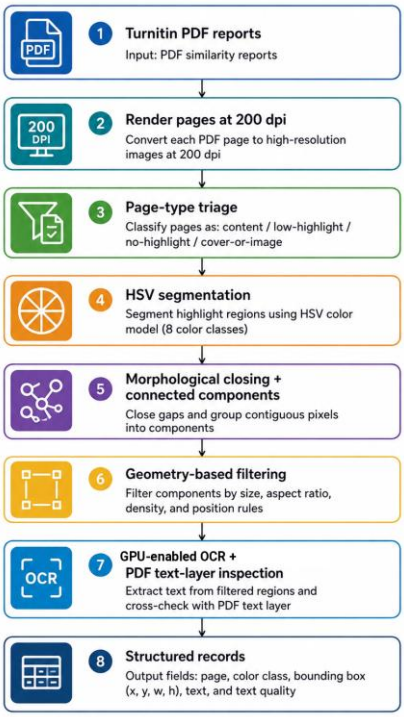

Figure 1. Report-aware workflow for converting Turnitin highlights into structured records.

Binary color masks were processed with morphological closing to reconnect fragments belonging to the same highlighted line. Connected-component analysis then converted masks into candidate regions. The staged design follows the broader document-engineering principle of separating localization, structure analysis, and transcription [23], while HSV-based segmentation provides an interpretable mechanism for isolating color-coded regions [11,13]. Candidate regions were filtered by size and aspect ratio to suppress compact source-ID labels and other colored artifacts before OCR.

EasyOCR with CUDA-enabled GPU execution was used as the principal recognition engine, and the PDF text layer was inspected with pdfplumber as supporting information about text availability. GPU-enabled recognition is appropriate when one page can generate many text crops and when a small number of documents create large recognition batches. Large-scale document-processing research similarly shows that GPU-backed OCR can be valuable for bursty, high-volume workloads [24]. Final records were designed to preserve document identifier, page number, color class, bounding-box coordinates, extracted text, and an operational text-quality category.

2.3 Page-type triage

Page triage used the fraction of pixels satisfying calibrated color criteria. If N_c denotes colored pixels and $H \times W$ is page area, the colored-pixel ratio was $r_{color} = N_c/(H \times W)$. Pages with $r_{color} < 0.0005$ (0.05%) were classified as no_highlight; pages from 0.0005 to 0.005 were classified as low_highlight; and pages above 0.005 (0.5%) were classified as content, subject to a separate cover_or_image rule for large graphical objects. Content and low_highlight pages were retained as highlight-bearing classes.

The purpose of triage was routing, not semantic classification. Early exclusion of irrelevant content is a common scalability principle in document analysis because downstream layout analysis and OCR are comparatively expensive. Modern scholarly-document and structured-extraction systems likewise benefit from identifying relevant page structures before applying deeper processing [12,25].

2.4 HSV segmentation and region filtering

RGB values were normalized to [0,1]. For each pixel, $C_{max} = \max(R',G',B')$, $C_{min} = \min(R',G',B')$, and $\Delta = C_{max} - C_{min}$. Value was $V = C_{max}$; saturation was zero when $C_{max} = 0$ and otherwise Δ/C_{max}; hue was calculated piecewise according to the maximum channel. The implementation used the OpenCV convention in which hue is represented on a 0–179 scale and saturation/value on 0–255 scales.

Thresholds were calibrated empirically from report highlights. This approach is computationally efficient and transparent under a controlled visual grammar, but comparative segmentation research shows that color representation and preprocessing alter region quality [15,16]. Adaptive segmentation can improve robustness when appearance varies [26]. The present study deliberately retained the simpler threshold approach because report colors and rendering resolution were comparatively standardized. Exact per-color HSV bounds were not recoverable from the aggregate source outputs used for this revision and therefore are not reconstructed or invented; a production release should archive the complete threshold configuration with the software version and rendering settings.

After segmentation, morphological closing used an approximately 15×3 pixel horizontal kernel. Connected components returned bounding boxes (x, y, w, h), and aspect ratio was defined as $AR = w/h$. Highlight bodies were expected to satisfy approximately $w \geq 80$ pixels, $8 \leq h \leq 60$ pixels, and $AR \geq 4$. Compact source-ID labels were characterized by approximately $w \leq 40$ pixels, $h \leq 40$ pixels, and $AR \leq 2.5$. Morphology and geometry provide interpretable post-segmentation control, consistent with the broader use of region-based processing after color localization [27-29]. Overlap remains a potential failure mode because intersecting or closely adjacent highlight regions can be merged into ambiguous components.

2.5 Text extraction, quality taxonomy, and statistics

The retained crops were recognized with EasyOCR. Extraction quality was summarized using five operational categories. Good records had an alphabetic-character ratio of at least 60%; noisy records had a ratio from 40% to 60%; garbled records fell below 40%; tiny records contained fewer than 10 characters; and empty records contained no text. With N_{alpha} representing alphabetic characters and N_{total} the total recognized characters, the ratio was $r_{alpha} = N_{alpha}/N_{total}$. Tiny was kept separate from recognition noise because a correctly read short phrase may still be semantically inadequate for paragraph-level analysis.

This taxonomy measured readability, not OCR accuracy. Recognition accuracy requires aligned ground-truth text and metrics such as CER or WER. Document-layout and OCR research shows that localization, text-line analysis, and recognition quality interact [30], while OCR language models are commonly evaluated through word- or character-level error rather than heuristic readability categories [31]. Accordingly, the present study reports the 76.75% good category only as an operational usability indicator.

Page-type and text-quality proportions were calculated from reported counts. Descriptive 95% Wilson intervals were computed for selected proportions. Because pages and records were nested within reports, these intervals were not treated as cluster-adjusted population inference. Color diversity was summarized using normalized Shannon entropy $H/\ln(K)$, where $H = -\Sigma p_i \ln(p_i)$, and concentration using the Herfindahl–Hirschman index, $HHI = \Sigma p_i^2$. Document workload was characterized by minimum, maximum, mean, median, and the share contributed by the ten densest reports.

Reproducibility depends on recording the configuration of every stage. A deployable implementation should retain rendering resolution, HSV bounds, morphological kernel size, connected-component rules, geometric thresholds, OCR language packs, OCR confidence values when available, hardware details, and software versions alongside the extracted records. This metadata is important because document structure and segmentation outcomes change with preprocessing, representation, and model configuration [15,16,18]. The present workflow controls one major source of variability by rendering at a fixed resolution, but future releases should store normalized bounding boxes in addition to pixel coordinates. A further improvement is stage-wise logging: each candidate region should retain whether it was merged by morphology, excluded by geometry, sent to OCR, or recovered from a native text layer. Such provenance would turn error analysis from a manual visual exercise into a queryable diagnostic process.

2.6 Interpretive boundary

The pipeline structures similarity evidence; it does not decide plagiarism. A highlighted passage may represent a legitimate quotation, properly cited overlap, common

technical language, a template phrase, or problematic copying. Reviews of plagiarism detection emphasize that reuse patterns vary substantially and that automated scores do not remove the need for contextual interpretation [1]. Future supervised models should therefore use human-validated labels rather than treating the existence of a highlight as a positive misconduct label. This boundary is also consistent with institutional approaches that combine technology with human evaluation [2].

3. Results and Discussion

3.1 Page routing and extraction yield

The 3,057 pages were classified as content (637; 20.84%), low_highlight (678; 22.18%), no_highlight (1,737; 56.82%), and cover_or_image (5; 0.16%). Content and low_highlight together represented 1,315 pages (43.02%), whereas 1,742 pages (56.98%) were routed away from detailed extraction. The descriptive Wilson interval for highlight-bearing pages was 41.27–44.78%, and the corresponding interval for routed-away pages was 55.22–58.73%. The complete dataset and pipeline accounting are summarized in Table 1, while the page-type distribution is visualized in Figure 2.

Table 1. Dataset and pipeline accounting

Stage / unit	n	Share / summary	Interpretation
PDF reports	106	—	Document-level corpus
All pages	3,057	100%	Page-classification denominator
Content	637	20.84%	Substantial highlighted content
Low-highlight	678	22.18%	Sparse but relevant highlights
No-highlight	1,737	56.82%	Routed away from detailed extraction
Cover/image	5	0.16%	Non-target pages
Pre-filter HSV components	11,471	8 color classes	Intermediate segmentation objects
Final highlight records	7,530	100% of final set	Post-filter structured records
Highlights/report	106 reports	Mean 71.04; median 44.5; range 4–633	Right-skewed workload

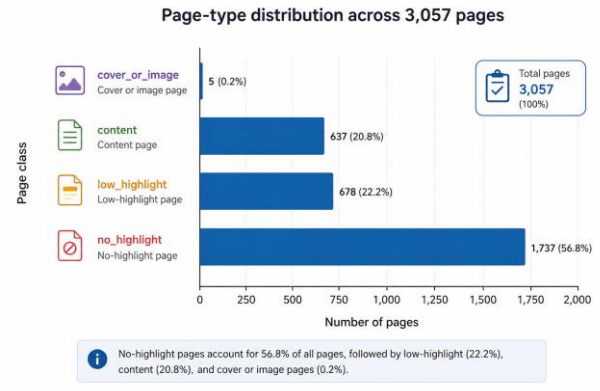

Figure 2. Page-type distribution across 3,057 pages.

This result makes page triage a first-order scalability mechanism rather than a minor preprocessing detail. More than half of the observed page volume did not enter the same segmentation and OCR workload as highlight-bearing pages. The 56.98% fraction should not be translated directly into wall-clock speedup because triage has its own computational cost and later processing cost varies by page. Even so, the structural reduction is substantial and follows the broader document-processing strategy of narrowing the material sent to expensive downstream extraction [25].

After morphology, connected-component analysis, and geometric filtering, 7,530 final records entered the structured output. The difference between 11,471 pre-filter components and 7,530 final records should not be described as a detection rate: the two stages use different object definitions. The pre-filter count represents connected color regions, whereas the final count reflects regions retained or consolidated after report-specific geometry. This denominator discipline is necessary for reproducible pipeline reporting.

3.2 HSV component structure

Across eight calibrated classes, blue produced 4,144 pre-filter components (36.13%), purple 2,064 (17.99%), pink 1,403 (12.23%), green 1,283 (11.18%), orange 846 (7.38%), cyan 742 (6.47%), yellow 717 (6.25%), and red 272 (2.37%). Blue and purple together represented 54.12% of all 11,471 components. Normalized Shannon entropy was 0.870 and HHI was 0.204, indicating meaningful diversity with moderate concentration rather than domination by a single class. Figure 3 shows the complete eight-class distribution.

The asymmetry is technically important. Calibration errors in blue or purple would affect a large portion of candidate regions, but the other six classes still represent 45.88% of components. A one- or two-color detector would therefore be unsuitable. At the same time, the result should not be interpreted semantically: Turnitin colors identify match groups, not severity or intent. Segmentation research similarly cautions that color-space behavior depends on task, algorithm, and visual conditions [17,18], so robustness should be tested under palette and rendering changes rather than inferred from this single environment.

3.3 Text usability and recognition limitations

Of the 7,530 final records, 5,779 were classified as good (76.75%; descriptive 95% Wilson interval, 75.78–77.69%). Tiny fragments accounted for 1,509 records (20.04%; 19.15–20.96%), noisy records for 63 (0.84%), garbled records for 42 (0.56%), and empty records for 137 (1.82%). Noisy, garbled, and empty categories together comprised 242 records, or 3.21% (2.84–3.64%). Figure 4 presents the full text-quality profile.

The dominant limitation is therefore not simply unreadable OCR. One fifth of final records are short enough to be context poor even when recognized correctly. For any later semantic analysis, these fragments should be mapped back to complete sentences or paragraphs in the native PDF text layer. That step would preserve the visual provenance of the highlight while restoring linguistic context. OCR research shows that errors introduced during localization or recognition can propagate into downstream text processing

[30,32], so future work should separately quantify localization error, transcription error, and context-reconstruction error.

Formal benchmarking remains necessary. The direct highlighted-text study by Lee et al. [14] reported localization and character-level recognition metrics, whereas the present results support only a readability taxonomy. CER and WER should be calculated on manually transcribed highlight crops stratified by color, crop length, page type, and document quality. Results should also be reported separately for native-text PDFs and image-heavy reports because OCR performance depends on layout, resolution, and visual conditions [31,33].

The text-quality profile also reveals an error-propagation problem that becomes important once the extraction layer feeds a similarity model. Character substitutions alter lexical features, missing words reduce overlap scores, and segmentation errors can detach a phrase from the paragraph that supplies its semantic context. The impact is therefore not linear. A small OCR error in a common word may have little effect, whereas an error in a distinctive technical term can alter retrieval or clustering. OCR-specific language modeling can reduce word-level error in specialized domains [30], while downstream plagiarism and document-similarity systems rely on lexical and semantic representations whose quality depends on the recovered text [34]. A future parser should therefore preserve raw OCR output, any corrected form, the crop image, and a confidence or validation status so that recognition uncertainty can be separated from similarity-model performance.

3.4 Workload heterogeneity and GPU-enabled OCR rationale

Highlight counts were strongly heterogeneous across 106 reports. The minimum was 4 records and the maximum 633; the mean was 71.04 and the median 44.5, producing a mean-to-median ratio of 1.596. The ten densest reports contained 633, 398, 354, 338, 332, 199, 190, 185, 183, and 174 records. Together, they generated 2,986 records, or 39.65% of the final corpus.

This distribution provides a stronger rationale for GPU-enabled OCR than an average workload alone. Recognition demand arrives in bursts: most reports are moderate, but a small set can create hundreds of crops. GPU batching is well matched to this queue structure, and large-scale OCR research demonstrates substantial throughput gains when document-processing pipelines are designed around CPU/GPU parallelism [24]. However, the present manuscript does not claim a measured speed advantage because a controlled CPU-versus-GPU benchmark was not available. Future experiments should report hardware, batch size, rendering time, segmentation time, OCR time, throughput, and high-percentile latency.

3.5 From extracted records to similarity analytics

The structured dataset creates an interface between document vision and later similarity analysis. Each record can retain text, page, bounding box, and match-group color. That provenance enables aggregation by sentence, paragraph, document, or cohort while preserving a route back to the visual evidence. Semantic comparison methods can then be applied after acquisition rather than being entangled with the extraction stage. Current plagiarism and document-similarity research spans lexical matching, hybrid weighted similarity, TF-IDF/cosine methods, and embedding-based representations [19-22]. A single lexical score is therefore insufficient to represent every form of reuse.

An analogous lexical or semantic model could operate on the present dataset only after highlighted regions have been converted into clean, contextualized text. TF-IDF/cosine approaches, hybrid similarity models, and embedding-based representations provide suitable downstream examples [35]. This distinction keeps the extraction problem separate from later similarity scoring: HSV segmentation and OCR are acquisition stages, whereas lexical or semantic models constitute a subsequent analytical layer.

A mature system should expose both the extracted evidence and its uncertainty. A downstream interface should allow users to inspect the original region, recovered text, surrounding paragraph, and any similarity score rather than presenting a binary plagiarism label. This design principle preserves auditability and reduces the risk that an engineering convenience becomes an unsupported disciplinary judgment.

Generalizability should be examined at the level of the report interface rather than assumed from the number of documents. The 106-report corpus is heterogeneous in highlight density, but the reports originate from one institutional context and appear to share a common Turnitin rendering grammar. A new interface version could alter saturation, source-label geometry, typography, or ancillary graphics without changing the underlying similarity semantics. Static thresholds would then face domain shift even though the conceptual task remained identical. Adaptive segmentation provides one response to appearance variability [26], while document-structured-extraction benchmarks emphasize the importance of testing across heterogeneous document layouts [36]. The appropriate validation design is therefore cross-template testing: calibrate on one report style, test on another, and report both localization and transcription degradation.

3.6 Comparison with related vision approaches and limitations

The current architecture favors classical computer vision because the report uses a relatively standardized visual grammar. HSV-based segmentation is attractive when color classes are known and interpretable [27,28], while document-analysis work shows the value of separating localization, layout interpretation, and transcription [36,37]. More complex learned models may be appropriate when layout or appearance varies substantially, but they also increase data and computational requirements. For a report parser, the appropriate criterion is therefore not maximum model complexity but the simplest method that remains robust across realistic report variation.

Five limitations are central. First, no manually annotated mask benchmark is available, so IoU, precision, recall, and F1 cannot be reported. Second, the readability taxonomy is not equivalent to CER/WER. Third, static HSV thresholds

may shift with interface updates, compression, scanning, or altered color management, as adaptive-segmentation research illustrates [26]. Fourth, pixel-based geometry depends on the fixed 200-dpi rendering scale. Fifth, overlapping highlight colors can generate ambiguous or merged components. These limitations constrain claims to extraction yield, composition, provenance, and operational usability.

Future validation should therefore use a stratified ground-truth sample. Each sampled page should be manually annotated for highlight masks, color class, source-ID objects, and exact text. Classical HSV thresholds should be compared with adaptive alternatives [26], and OCR evaluation should use CER and WER following established document-recognition practice [14]. Document-level resampling should account for clustering. Only after this validation should paragraph-level similarity models or predictive decision support be evaluated.

Accordingly, the present evaluation should be interpreted as a systems-level feasibility study rather than a benchmark of detection accuracy. Its strongest evidence concerns throughput reduction, output composition, provenance preservation, and operational text usability. Claims about localization accuracy, transcription correctness, or deployment reliability remain explicitly reserved for a future ground-truth study. The derived descriptive diagnostics are summarized in Table 2. This separation reduces the risk of overstating performance while making the next validation step clear and reproducible.

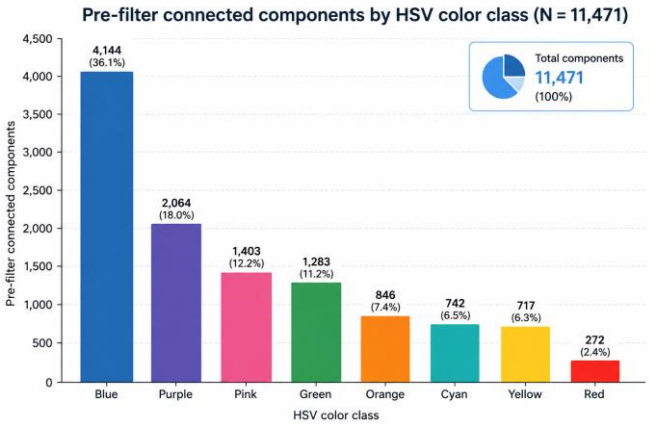

Figure 3. Distribution of 11,471 pre-filter HSV color components.

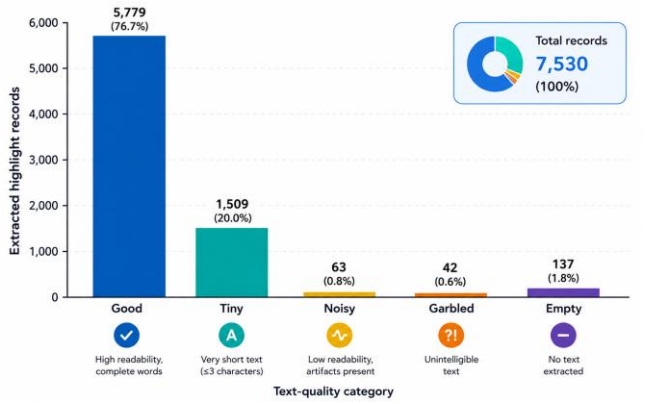

Figure 4. Text-quality profile of 7,530 final structured highlight records.

Table 2. Derived quantitative diagnostics from reported aggregate counts

Diagnostic	Estimate	95% descriptive interval	Meaning
Highlight-bearing pages	43.02%	41.27–44.78%	content + low_highlight
Pages routed away	56.98%	55.22–58.73%	no_highlight + cover_or_image
Good text records	76.75%	75.78–77.69%	Alphabetic ratio ≥60%
Tiny text records	20.04%	19.15–20.96%	Fewer than 10 characters
Noisy + garbled + empty	3.21%	2.84–3.64%	Degraded or absent text
Blue + purple share	54.12%	—	Largest two HSV classes
Normalized Shannon entropy	0.870	—	Color diversity; 1 = uniform
HSV HHI	0.204	—	Moderate component concentration
Mean/median count	1.596	—	Right-skew indicator
Top-10 concentration	39.65%	—	Share of records in 10 densest reports

From a practical engineering perspective, the structured representation supports quality-control functions that are difficult with a single similarity percentage. Institutions could audit the distribution of match fragments by document section, identify recurrent short phrases that inflate similarity without substantive concern, or prioritize unusually dense reports for human review. None of these uses requires the system to infer misconduct. They require reliable acquisition, context reconstruction, and transparent linkage to the original report. This distinction is consistent with the broader evolution from simple lexical matching toward multimodal, hybrid, and semantically informed plagiarism analysis [7,38]. It also clarifies why the pipeline should be evaluated as document infrastructure: its success is measured by whether it preserves evidence accurately enough for later analysis, not by whether it replaces academic judgment.

4. Conclusion

This study demonstrates that color-coded Turnitin similarity reports can be transformed into structured, machine-readable data through a report-aware pipeline that combines page triage, HSV segmentation, morphology, connected-component analysis, geometric filtering, and GPU-enabled OCR. Across 106 reports and 3,057 pages, 56.98% of pages were routed away from detailed processing. HSV segmentation produced 11,471 pre-filter components across eight color classes, and post-filter extraction produced 7,530 structured records. Of these, 76.75% satisfied the operational good-readability criterion, while 20.04% were very short and 3.21% were noisy, garbled, or empty.

The principal contribution is an auditable acquisition layer rather than a new plagiarism verdict. The system preserves page, color, location, and text provenance so that later lexical or semantic models can operate on structured evidence. The study should be interpreted as systems-level feasibility rather than as a benchmark of localization or OCR accuracy. Ground-truth localization metrics, CER/WER, cross-version robustness, and human-validated semantic labels are required before the pipeline is used for high-stakes decisions.

Declarations

Declaration of generative AI and AI-assisted technologies in the manuscript preparation process. During the preparation and revision of this work, the authors used OpenAI's ChatGPT to assist with language refinement, readability, internal consistency, and checks of figure and table referencing. After using this tool, the authors reviewed and edited the content as necessary and take full responsibility for the content of the published article.

CRediT authorship contribution statement. Mutiara Yusuf: Conceptualization, Methodology, Software, Investigation, Formal analysis, Data curation, Visualization, Writing – original draft. Rizki Yusliana Bakti: Validation, Supervision, Writing – review & editing. Titin Wahyuni: Validation, Supervision, Writing – review & editing. All authors read and approved the final manuscript.

Funding. This research received no specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Declaration of competing interest. The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability. The data analyzed in this study were derived from Turnitin similarity reports used for methodological evaluation. Because the source reports contain student-generated content and potentially identifiable academic information, the raw reports are not publicly available. Aggregate analytical data supporting the findings of this study are available from the corresponding author upon reasonable request, subject to institutional data-protection requirements.

References

[1] Wilson K, Sy J, Hylton K, Guthrie-Dixon N, Myers T. Assessing the chemistry 'cookbook' culture—Caribbean tertiary students' perceptions of plagiarism in General Chemistry I laboratory reports. Journal of Academic Ethics. 2025;23(3):867-83.

[2] Guruge DB, Kadel R. Towards a holistic framework to mitigate and detect contract cheating within an academic institute—A proposal. Education Sciences, 13(2), 148. 2023.

[3] Choiriyah S, Ramadhan S, Nugroho A, Pembangunan HRP, Muharom F. Artificial intelligence-driven learning assessment in faculties of education: An exploratory study. Munaddhomah. 2025;6(3):482-95.

[4] Kodali RK, Shekhar T, Boppana L. Automated plagiarism detection in Moodle. IEEE Region 10 Annual International Conference Proceedings2023. p. 176-81.

[5] Kavatage AV, Menon S, Devi SR, Patil S, Shilpa S. Multi-model essay evaluation with optical character recognition and plagiarism detection. Lecture Notes on Data Engineering and Communications Technologies2023. p. 749-59.

[6] Chaudhari S, Shende AR, Zade TG, Bhakare UN, Dhutraj PS, Ghugal MP. Online assignment plagiarism checker with OCR. IEEE International Students' Conference on Electrical, Electronics and Computer Science2024.

[7] Kumar A, Choudhary P, Dutta A, Vij A. Multi-modal plagiarism detection in the era of AI: A comprehensive review and novel framework proposal. 2025 2nd International Conference on Advanced Computing and Emerging Technologies2025.

[8] Chandel H, Omar V, Narooka P, Choudhary A, Agarwal K, Vishnoi A. Transforming assignment evaluation: Deep learning approaches for assignment analysis and AI-driven plagiarism detection. Lecture Notes in Networks and Systems2026. p. 372-81.

[9] Sharmeen A, Agarwal N, Suman A, Agarwal S, Kumar K. Process design to self-extract text from images for similarity check. Lecture Notes in Networks and Systems2022. p. 109-17.

[10] Namburu A, Surendran A, Balaji SV, Mohan S, Iwendi C. DocCompare: An approach to prevent the problem of character injection in document similarity algorithm. Mathematics, 10(22), 4256. 2022.

[11] Amutha M, Guru Prasaath RG, Guru Prashath M. OCR-based document parsing and evaluation using AI. Proceedings of the International Conference on Sustainable Communication Networks and Application2024. p. 1407-13.

[12] Raval P, Bhaidasna H. Metadata extraction from scholarly document using deep learning. Proceedings of the International Conference on Inventive Computing and Informatics2025. p. 656-60.

[13] Ospan A, Mansurova M, Sailau A, Sarsembayeva T. X-CUT++: A layout-aware OCR pipeline for Kazakh-language newspapers. KazNU Bulletin Mathematics, Mechanics, Computer Science Series. 2026;130(2):181-98.

[14] Lee S, Kim Y, Baek M, Jang J, Kim A, Kim Y, et al. Automated Korean-English bilingual highlighted text extraction using HSV segmentation. 2026 International Conference on Electronics, Information, and Communication2026.

[15] Abdelsadek DA, Al-Berry MN, Ebied HM, Hassaan M. Impact of using different color spaces on image segmentation. Lecture Notes on Data Engineering and Communications Technologies2022. p. 456-71.

[16] Al Garea S, Das S, Callaway M. Clustering based flood area segmentation benchmarking in different colour spaces. Proceedings of the IEEE International Conference on Future Machine Learning and Data Science2024. p. 337-43.

[17] Armağan H. Color based segmentation with k-means clustering algorithm and numerical analysis of the effect of color spaces on image quantities. El-Cezeri Journal of Science and Engineering. 2022;9(4):1506-17.

[18] Budak L, Grbic R, Cetic N, Kastelan I. Color image segmentation based on thresholding for advanced driver assistance systems. IEEE Zooming Innovation in Consumer Technologies Conference2022. p. 271-6.

[19] Narsale A, Nannaware S, Patil V, Sapkale R, Patil R. NLP-driven plagiarism detection: A smarter approach. 2025 2nd International Conference on Integration of Computational Intelligent System2025.

[20] Paladugu RK, Kancherla GR. Identification of word level information based semantic similarity using extended GloVe embeddings for clustering and classification analysis. International Journal of Engineering Trends and Technology. 2024;72(8):212-27.

[21] Zadgaonkar A, Dhoot A, Chinchghare I, Rahangdale P, Iyer G, Bhattacharya P. An approach for measuring unstructured text document similarity using LDA-BERT embedding model. ISML—Intelligent Systems and Machine Learning Conference2024. p. 19-24.

[22] Bhoyar M, Bhagat L, Bhagat M, Bhongade R. InferSync TextLens: Cognitive PDF comparison information technology. Lecture Notes in Electrical Engineering2025. p. 323-34.

[23] Bio R, Santos J. Real-time multi-color object detection on ESP32 embedded systems. 2026 International Symposium of Systems, Advanced Technologies and Knowledge2026.

[24] Bhatt R, Pandey S, Sharma N, Pathak N, Nandiraju SKK. Integrating YOLO, CNN, and EasyOCR for AI-powered fake medicine detection. Lecture Notes in Networks and Systems2026. p. 495-504.

[25] Makawana M, Mehta RG. A novel network-based paragraph filtering technique for legal document similarity analysis. Artificial Intelligence and Law. 2023.

[26] AlRifaee M, Almanasra S, Hnaif A, Althunibat A, Abdallah M, Alrawashdeh T. Adaptive segmentation for unconstrained iris recognition. Computers, Materials & Continua. 2024;78(2):1591-609.

[27] Chavan S, Yu X, Saniie J. High precision analog gauge reader using optical flow and computer vision. IEEE International Conference on Electro Information Technology2022. p. 171-5.

[28] Fataniya B, Shah D, Jariwala R. Clothing segmentation using HSV features and morphological algorithms. Proceedings of the International Conference on Innovations in Data Communication Technologies and Applications2025. p. 1256-60.
[29] Deng H, Ge H, Lian J, Wang H. Research on neural network model based on an intersection-free segmentation algorithm and HSV feature classification. International Academic Exchange Conference on Science and Technology Innovation2024. p. 164-8.
[30] Arisa EZ, Yunanto AA, Fariza A. IPR registration support system with OCR and fuzzy string matching. Proceedings of the IEEE Information Technology International Seminar2025. p. 250-5.
[31] Nguyen M, Tran K, Huynh TH. Legibility vs. extractability: Crafting visual defenses against automated OCR Lecture Notes in Computer Science2026. p. 3-14.
[32] Baclayon CAL, Costelo KOR, Flores JJL, Sta. Romana CLC. Automated handwritten essay evaluation in Moodle: Leveraging Google Vision OCR and Mistral 7b. ACM International Conference Proceedings Series2024. p. 240-6.
[33] Nair B, Sreekumar R. Comparative analysis of HSV and UV-based approaches denoising degraded document images. Procedia Computer Science. 2024;235:2370-82.
[34] Rohit R, Gandheesh SA, Suriya KS, Basa Pati P. System for enhancing accuracy of noisy text using deep network language models. IEEE International Conference for Convergence in Technology2023.
[35] Madoa S, Mulyadi I, Darniati D. Implementasi metode TF-IDF dan cosine similarity pada sistem pencarian artikel yang relevan. Journal of Muhammadiyah's Application Technology. 2026;5(2):227-44.
[36] Ananthagoma M, Aditya Sethuraam S, Varun M, Sree Varshini V, Sai Gugan P, Srihari S. ColorTrackMapper: A framework for advanced color segmentation in industrial imaging systems. 2025 International Conference on Smart and Sustainable Technology2025.
[37] Andrade JDS, Júnior SLDA, Filho PLDP, Júnior JA, Miranda GV, Rodrigues PJ, Tusset AM. Comparative study between digital image processing algorithms and YOLOv11 in the segmentation of lettuce cultivars. Communications in Computer and Information Science2026. p. 298-311.
[38] Guo Q, Huang G, Wang Y, Zhou Y. A text dependent copy-paste plagiarism and text-rewriting plagiarism model. Journal of Intelligent & Fuzzy Systems. 2025;48(6):849-58.