PRESIDENCY UNIVERSITY
Private University Estd. in Karnataka State by Act No. 41 of 2013
Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

IMAGE CORRECTNESS FOR A PRODUCT ON THE MARKETPLACE

A PROJECT REPORT

Submitted by

VIKAS YADAV - 20221CIT0003

SHAIK YASIN SHAHID - 20221CIT0005

AKHIL HARI - 20221CIT0033

Under the guidance of,

SOUMYA

ASSISTANT PROFESSOR

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING
(INTERNET OF THINGS)

PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2025

PRESIDENCY UNIVERSITY
Private University Estd. in Karnataka State by Act No. 41 of 2013
Itgalpura, Rajankunte, Yelahanka, Bengaluru – 560064

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this report “**IMAGE CORRECTNESS FOR A PRODUCT ON THE MARKETPLACE** ” is a bonafide work of “VIKAS YADAV (20221CIT0003), SHAIK YASIN SHAHID (20221CIT0005), AKHIL HARI (20221CIT0033)”, who have successfully carried out the project work and submitted the report for partial fulfilment of the requirements for the award of the degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE ENGINEERING, INTERNET OF THINGS during 2025-26.

Ms. Soumya
Project Guide
Presidency School of
Computer Science and
Engineering
Presidency University

Dr. Sharmasth Vali Y
Program Project
Coordinator Presidency
School of Computer
Science and Engineering
Presidency University

Dr. Sampath A K
School Project
Coordinate
Presidency School of
Computer Science and
Engineering
Presidency University

Dr. Anandaraj S P
Head of the
Department
Presidency School of
Computer Science and
Engineering
Presidency University

Dr. Duraipandian N
Dean
PSCS & PSIS
Presidency University

Dr. Shakkeera L
Associate Dean
Presidency School of computer
Science and Engineering
Presidency University

Sl.no	Name	Signature	Date
1			

2			

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We the students of final year B.Tech in COMPUTER SCIENCE ENGINEERING, INTERNET OF THINGS at Presidency University, Bengaluru, named VIKAS YADAV, SHAIK YASIN SHAHID, AKHIL HARI , hereby declare that the project work titled "**IMAGE CORRECTNESS FOR A PRODUCT ON THE MARKETPLACE**" has been independently carried out by us and submitted in partial fulfillment for the award of the degree of B.Tech in COMPUTER SCIENCE ENGINEERING, INTERNET OF THINGS during the academic year of 2025-26.

Vikas Yadav	USN: 20221CIT0003
Shaik Yasin Shahid	USN:20221CIT0005
Akhil Hari	USN:20221CIT0033

PLACE: BENGALURU

DATE:

ACKNOWLEDGEMENT

For completing this project work, We/I have received the support and the guidance from many people whom I would like to mention with deep sense of gratitude and indebtedness. We extend our gratitude to our beloved **Chancellor, Pro-Vice Chancellor, and Registrar** for their support and encouragement in completion of the project.

I would like to sincerely thank my internal guide **Ms. Soumya, Assistant Professor**, Presidency School of Computer Science and Engineering, Presidency University, for his/her moral support, motivation, timely guidance and encouragement provided to us during the period of our project work.

I am also thankful to **Dr. Anandaraj, Professor, Head of the Department, Presidency School of Computer Science and Engineering** Presidency University, for his mentorship and encouragement.

We express our cordial thanks to **Dr. Duraipandian N**, Dean PSCS & PSIS, **Dr. Shakkeera L**, Associate Dean, Presidency School of computer Science and Engineering and the Management of Presidency University for providing the required facilities and intellectually stimulating environment that aided in the completion of my project work.

We are grateful to **Dr. Sampath A K, and Dr. Geetha A, PSCS** Project Coordinators**, Dr. Sharmast Vali, Program Project Coordinator**, Presidency School of Computer Science and Engineering, or facilitating problem statements, coordinating reviews, monitoring progress, and providing their valuable support and guidance.

We are also grateful to Teaching and Non-Teaching staff of Presidency School of Computer Science and Engineering and also staff from other departments who have extended their valuable help and cooperation.

VIKAS YADAV
SHAIK YASIN SHAHID
AKHIL HARI

Abstract

E-commerce platforms depend heavily on visually attractive and informative product images to enable customers to make sound purchasing decisions on their own. Unfortunately, there are lots of instances where sellers upload pictures that are deceptive, low in quality, out of focus, or even completely unrelated to the product in question. Such problems result in a bad customer experience, an increase in the rate of product returns, and a loss of trust in the platform's credibility. Manual moderation systems that are used by marketplaces to combat these situations are finding it hard to keep up with the overwhelming number of uploads that go on each day, which is in the millions. On the contrary, rule-based validation approaches that are in place do not always identify the areas that need correction in a complicated way. For instance, they can hardly spot watermarks that are not obvious, locate areas where promotional text is written, find duplicate pictures, or determine slight mismatches between categories. These situations give rise to the necessity for an intelligent, scalable, automated image verification method to be implemented.

This undertaking presents a smart-image-correctness-verification system, an AI-powered device that evaluates product images by analyzing them stepwise. The solution achieves its goals by implementing deep learning technologies, for example, it applies YOLO-based object detection to verify that the primary item is in fact there and that it is not only visible but also appropriately placed in the image. To link the text to the visuals, CLIP models can be used that create embeddings for this purpose. They confirm that the new picture is related to the product title and category. OCR modules for text identification may be used to pinpoint the unwanted info like embedded watermarks or a brand logo of a company or if some misleading promotional text has been put there. Illumination, Sharpness, Aspect ratio and Resolution are some aspects of an image that should be met by the marketplace standards and be checked by image quality metrics. Also, perceptual hashing and deep feature similarity algorithms may be used to detect the recycles or near-duplication of image files in different listings.

The output of each module is forwarded to a decision engine that determines the category in which the uploaded image fits best: Accepted, Rejected, or Manual Review. Instant response is given to sellers along with the advice for improvement thus repeating uploads are less needed. Apart from this, the system comprises an admin review panel which is always available for the continuing human-machine sample-based closed-loop training to upgrade accuracy with time.

Table of Content

Sl. No.	Title	Page No.
	Declaration	III
	Acknowledgement	IV
	Abstract	V
	List of Figures	VIII
	List of Tables	IX
	Abbreviations	X
1.	Introduction 1.1 Background 1.2 Statistics of project 1.3 Prior existing technologies 1.4 Proposed approach 1.5 Objectives 1.6 SDGs 1.7 Overview of project report	1-5
2.	Literature review	6-10
3.	Methodology	11-12
4.	Project management 4.1 Project timeline 4.2 Risk analysis 4.3 Project budget	13-16
5.	Analysis and Design 5.1 Requirements 5.2 Block Diagram 5.3 System Flow Chart 5.4 Choosing devices 5.5 Designing units 5.6 Standards 5.7 Mapping with IoTWF reference model layers 5.8 Domain model specification	17-27

	5.9 Communication model 5.10 IoT deployment level 5.11 Functional view 5.12 Mapping IoT deployment level with functional view 5.13 Operational view 5.14 Other Design	
6.	Hardware, Software and Simulation 6.1 Hardware 6.2 Software development tools 6.3 Software code 6.4 Simulation	28-38
7.	Evaluation and Results 7.1 Test points 7.2 Test plan 7.3 Test result 7.4 Insights	39-43
8.	Social, Legal, Ethical, Sustainability and Safety Aspects 8.1 Social aspects 8.2 Legal aspects 8.3 Ethical aspects 8.4 Sustainability aspects 8.5 Safety aspects	44-45
9.	Conclusion	46
	Future Recommendation	47
	References	48
	Base Paper	49-52
	Appendix	53-57

List of Figures

Figure	Caption	Page No
Figure 1.1	Sustainable development goals	5
Figure 3.1	Methodology	12
Figure 5.1	Block Diagram	20

List of Tables

Table	Caption	Page no
Table 2.1	Summary of Literature reviews	8-10
Table 4.1	Project planning timeline	13-14
Table 4.2	Risk Analysis	14-15
Table 5.1	Requirements	17-18
Table 5.2	Requirements Matrix	18-19
Table 5.3	Mapping with IoTWF Reference Model	20-22
Table 5.6	Iot Deployment Level	25
Table 6.1	Development Hardware	28
Table 6.2.1	System software	29
Table 6.2.2	AI& Deep Learning Frameworks	29
Table 6.2.3	Database & Cloud Tools	29
Table 7.1	Test Points	39-40

Abbreviations

Abbreviations	Full Forms
AI	Artificial Intelligence
API	Application Programming Interface
AWS	Amazon Web Services
ACL	Access Control List
CNN	Convolutional Neural Network
COCO	Common Objects in Context (dataset)
CLIP	Contrastive Language–Image Pretraining
CRNN	Convolutional Recurrent Neural Network
DPDPA	Digital Personal Data Protection Act (India, 2023)
DFD	Data Flow Diagram
GPU	Graphics Processing Unit
GDPR	General Data Protection Regulation
HOG	Histogram of Oriented Gradients
IoT	Internet of Things
IoTWF	IoT World Forum
OCR	Optical Character Recognition
SSD	Solid-State Drive
UML	Unified Modeling Language
XAI	Explainable Artificial Intelligence
YOLO	Youn Only Look Once

Chapter 1

Introduction

1.1 Background

E-commerce platforms being Amazon, Flipkart, and Meesho have restructured the retail ecosystem through the facilitation of sellers to digitally list their products for a global customer base. In an online shopping setup, product images are the main source of visual information for customers as they are not allowed to physically examine the items. An image that is clear, accurate, and in line with the policy creates trust, leads to higher conversion rates, and lessens customer uncertainty.

But, there are many sellers who upload images that are misleading, poorly captured, or irrelevant and hence do not represent the product that is being sold truly. Images with multiple unrelated objects, low resolution, excessive editing, watermarks, or totally mismatched category visuals are some of the examples of pictures that can cause confusion among customers. Such discrepancies result in customer dissatisfaction, increased number of returns, negative reviews, and thus the overall trust of the marketplace getting affected. Traditional manual moderation methods are having difficulties in scaling up with millions of uploads that take place daily and also in being consistent.

There is a strong argument for the existence of an automated, intelligent image correctness verification system that is capable of assessing the efficiency and accuracy of every product image uploaded.

1.2 Statistics

The images of the product have a great impact on the purchasing decision of online buyers. Studies and industry reports reveal the following main points:

About 70–80% of online customers rely primarily on product images before purchasing.

22–30% of product returns are the result of inaccurate or misleading images in product listings.

E-commerce platforms get millions of product image uploads every day, which makes manual verification infeasible.

1.3 Prior existing technologies

Prior to the introduction of AI-based moderation, the marketplaces were dependent on:

1. Manual Moderation

Each image of a product was checked by hand regarding
compliance. Limitations:

- Time-consuming
- Expensive
- Prone to human error
- Incapable of supporting millions of uploads.

2. Basic Rule-Based Systems

The basic automated verification software checks like:

- File size validation
- Format checking
- Aspect ratio limits

3. Basic Image Processing Techniques

Applied in low level tests such as brightness and resolution checking. Limitations:

- No concept available about image and object context.
- Inadequate to comply with policies.

There are many sellers who upload images that are misleading, poorly captured, or irrelevant and hence do not represent the product that is being sold truly. Images with multiple unrelated objects, low resolution, excessive editing, watermarks, or totally mismatched category visuals are some of the examples of pictures that can cause confusion among customers. Such discrepancies result in customer dissatisfaction, increased number of returns, negative reviews, and thus the overall trust of the marketplace getting affected. Traditional manual moderation methods are having difficulties in scaling up with millions of uploads that take place daily and also in being consistent. The images of the product have a great impact on the purchasing decision of online buyers. Studies and industry reports reveal the following main points

1.4 Proposed approach

The current project suggests a multi-module AI-based Image Correctness Verification System that checks product images with the help of the sophisticated deep-learning method. The approach includes:

- Object detection using YOLO to detect the presence of the main product and detect other or irrelevant objects.
- Image-text relevance scoring based on CLIP to make sure that the image is associated with the product title and the category.
- OCR (Optical Character Recognition) that will identify watermarks, embedded text, promotional text, and brand logos.
- To evaluate resolution, sharpness, brightness, and cropping.
- Background verification to verify clean and consistent and policy Compliant backgrounds.
- Perceptual hashing and deep feature matching Duplicate detection.

1.5 Objectives

The main goals of this project are stated in order to show how the system works, whether it has analytical, management and security parts and whether it is ready to be deployed. The aims of Image Correctness Verification System are:

i. Behaviour

- To automatically identify and locate the primary product in an uploaded image with an object-detection model (YOLO), to make sure that there is one and only one relevant product of the image.

ii. Analysis

- To check uploaded pictures on the basis of accuracy by checking the relevance with product text (applying CLIP embeddings), removable text/watermarks (applying OCR), and examining visual qualities of pictures, including resolution, brightness and sharpness.

iii. System Management

- To have a formal process of validation, which puts in place a workflow involving image assessment, retention of intermediate measures, and decision status (Accepted/Rejected/Manual Review) with a rule-based decision engine.

iv. Security

- To provide safe manipulation, archiving, and computation of uploaded product images through implementation of role-based access control on sellers and administrators, avert alterations or abuse of image data that are not authorized.

v. Deployment

- The objective is to create a scalable, deployable pipeline that can handle the processing of large quantities of product pictures in real time, and streamline with marketplace to provide unceasing automated contentment.

1.6 SDGs

The project contributes to the following SDGs of the UN:

Goal number 9: Industry, Innovation, and Infrastructure.

• Brings about innovativeness by means of AI automation of e-commerce.

• Enhances online infrastructure of scalable market place operations.

Goal 12: Decentralized Consumption and Production.

• Maintains the correct information to the customers in making prudent buying choices.

• Lowers the product returns, improving packaging wastage and emissions
during transportation.

SDG 9: Industry, Innovation, and Infrastructure

- Promotes innovation through AI-based automation in e-commerce.
- Improves digital infrastructure for scalable marketplace operations.

SDG 12: Responsible Consumption and Production

- Ensures that customers receive accurate information for informed purchasing decisions.
- Reduces product returns, thereby lowering packaging waste and transportation emissions.

- Enhances marketplace reliability, boosting economic efficiency.

End poverty in all its forms everywhere
End hunger, achieve food security and improved nutrition and promote sustainable agriculture
Ensure healthy lives and promote well-being for all at all ages
Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all
Achieve gender equality and empower all women and girls
Ensure availability and sustainable management of water and sanitation for all
Ensure access to affordable, reliable, sustainable and modern energy for all
Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all
Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation
Reduce inequality within and among countries
Make cities and human settlements inclusive, safe, resilient and sustainable
Ensure sustainable consumption and production patterns
Take urgent action to combat climate change and its impacts
Conserve and sustainably use the oceans, seas and marine resources for sustainable development
Protect, restore and promote sustainable use of terrestrial ecosystems, sustainably manage forests, combat desertification, and halt and reverse land degradation and halt biodiversity loss
Promote peaceful and inclusive societies for sustainable development, provide access to justice for all and build effective, accountable and inclusive institutions at all levels
Strengthen the means of implementation and revitalize the global partnership for sustainable development
SUSTAINABLE DEVELOPMENT GOALS

Fig 1.1 Sustainable Development Goals

1.7 Over View Of Project Report

This project report is organized into the following chapters:

- **Chapter 1** provides the project, background, objectives and SDG fit.
- **Chapter 2** introduced in detail Chapter 3 is the literature review of the existing studies on object detection, OCR, watermark detection and AI-based relevance scoring.
- **Chapter 3** describes the methodology, workflows, system architecture and the development model that was selected.
- **Chapter 4** covers project planning, task scheduling, and resource management.
- **Chapter 5** provides system analysis, UML diagrams, DFDs, and architectural models.
- **Chapter 6** outlines software implementation, algorithms, models and technologies utilized.
- **Chapter 7** will contain assessment, testing performance, screen shots and performance measures.
- **Chapter 8** evaluates the social, ethical, legal, and sustainability factors in the implementation of AI moderation systems.
- **Chapter 9** culminates the project and proposes new ways of improvement and enhancement in the future.

Chapter 2

Literature review

1. Krizhevsky et al. (2012) – Deep CNNs for Image Representation

This work introduced deep CNNs for large-scale image classification and showed that learned features outperform traditional vision methods. The approach provided strong visual representations useful for downstream tasks such as object detection. However, it focused only on classification and did not address multi-object detection, semantic relevance, or policy compliance needed in marketplace image verification.

2. Ren, He, Girshick & Sun (2015) – Region Proposal Networks for Object Detection

A faster R-CNN enhanced the detection efficiency through the use of region proposals networks that are efficient in the approach to localizing objects. Its strict boundaries help determine whether a product is easily visible in the images attached by the sellers. Although it is accurate, the computationally expensive technique cannot be used in real-time in large scales of moderation.

3. Redmon & Farhadi (2018) – Real-Time Detection through YOLO

YOLOv3 enabled real-time detection by predicting all bounding boxes in a single pass, making it practical for rapid screening of product images. It identifies multiple objects efficiently but struggles with small or fine details, limiting its ability to detect subtle watermarks and small text present in many product photos.

4. Bochkovskiy et al. (2020) – Enhanced Accuracy with YOLOv4

YOLOv4 enhanced detection accuracy with improved training techniques and remained efficient for real-time use. It performs better on varied product images but still cannot detect watermarks, assess quality, or determine textual relevance on its own, necessitating integration with other modules.

5. Radford et al. (2021) – Image–Text Alignment with CLIP

CLIP introduced a multimodal embedding approach for evaluating image–text similarity. Its zero-shot performance is helpful in identifying mismatches between product images.

6. Shi, Bai & Yao (2016) – OCR for Text and Watermark Recognition

The CRNN model combined CNN and RNN layers to recognize text in natural scenes, supporting detection of watermarks and embedded promotional text. While effective on irregular text, the method struggles with low-resolution or heavily compressed images, which are common in seller uploads.

7. Mittal et al. (2012) – No-Reference Image Quality Assessment

BRISQUE introduced a no-reference image quality model that evaluates distortion without needing a reference image. It is useful for assessing low-quality or blurred product images. However, it can misinterpret stylistic product photography as poor quality, making threshold tuning necessary.

8. Zauner (2010) – Perceptual Hashing for Similarity and Duplication

Perceptual hashing provided a lightweight method for detecting duplicate or near-duplicate images across listings. While efficient, it is not robust against strong edits, cropping, or color changes, limiting its use for detecting heavily modified duplicate product photos.

9. Zhong et al. (2023) – Deep Learning Approaches for Watermark Detection

This study reviewed deep-learning watermark detectors capable of identifying visible and semi-transparent watermarks. Although effective on structured datasets, real-world watermark patterns vary widely, reducing generalization. This highlights the need for hybrid or domain-adapted watermark detection in marketplace images.

10. Faghri et al. (2017) – Visual–Semantic Embedding with Hard Negatives

The VSE++ framework improved image–text retrieval using hard-negative mining, helping distinguish between visually similar items. While useful for relevance checking, it depends heavily on high-quality training pairs and does not address object presence or quality assessment.

Table 2.1 Summary of Literature reviews

S.No	Author & Year	Methods	Key Features	Merits	Demerits
1	ImageNet Classification with Deep Convolutional Networks, 2012,NIPS	Convolutional Neural Networks (CNNs)	Learns hierarchical image Features for large-scale image classification; strong foundation for visual understanding	Delivers high accuracy and robust feature extraction useful for downstream product-image analysis	Designed for classification only; cannot detect multiple objects or verify semantic relevance
2	YOLOv3: An Incremental Improvement, 2018, arXiv / CVPR Workshops	Single-stage Object Detection	Real-time detection of multiple objects in a single forward pass; suitable for fast marketplace verification	High inference speed enables real-time image checking during product upload	Struggles with small objects such as tiny watermarks or small product details
3	End-to-End Scene Text Recognition with CRNN, 2016, IEEE TPAMI	CRNN (CNN + RNN + CTC)	Extracts embedded text from natural images; useful for identifying watermarks, brand names	Works effectively on irregular and curved text in real-world product images	Accuracy drops for low-resolution, blurry, or highly stylized text present in seller images
4	Learning	Contrastive Image–Text	Aligns	Excellen	Less effective

	Transferable Visual Models Using Natural Language Supervision (CLIP), 2021, ICML	Embeddings	images with textual descriptions; verifies category and title relevance of product images	t zero-shot performance; helps detect category mismatch or misleading photos	for fine-grained product attribute s without domain-specific fine-tuning
5	No-Reference Image Quality Assessment Using BRISQUE, 2012, IEEE Signal Processing	No-reference Image Quality Analysis	Measures sharpness, brightness, noise, and naturalness without a reference image	Useful for automatically rejecting low-quality product images uploaded by sellers	May misclassify creative photography styles as poor quality; thresholds need tuning
6	Faster R-CNN: Towards Real Time Object Detection, 2015, IEEE TPAMI	Region Proposal Networks (RPN) + CNN	Two-stage detection with high localization accuracy; useful for verifying presence of a single product in an image	Produces precise bounding boxes and strong detection performance	Slower than single-stage detectors; unsuitable for high-volume real-time e-commerce moderation
7	Perceptual Hashing for Image Authentication, 2010, Communication and Multimedia	pHash (Perceptual Hashing)	Generates a compact fingerprint for identifying near-duplicate or reused images	Lightweight and fast; effective for detecting repeated product	Not robust against heavy edits, color changes, or cropping; may miss

	Security			images across sellers	altered duplicates
8	Robust Watermark Detection Using CNNs, 2020, IEEE Access	Deep CNN-based watermark classification	Learns watermark patterns (visible / semi-transparent for automatic detection	Useful for marketplaces where promotional watermarks violate listing policies	Requires large training datasets; sensitive to lighting variations and blended text
9	Attention-Based OCR for Scene Text Understanding, 2021, Pattern Recognition Letters	Attention OCR + Seq2Seq	Improved text extraction in cluttered and low-quality images; handles curved and multilingual text	Enhances detection of brand names, labels, and unwanted text in seller images	High computational cost; performance drops on extremely low-resolution product photos
10	Multimodal Retrieval using Hard-Negative Mining (VSE++), 2017, ICCV	Visual–Semantic Embeddings	Aligns image and text meaningfully; detects mismatch between product image and listing title	High accuracy in semantic similarity tasks; reduces misleading or irrelevant product images	Requires domain adaptation for fine-grained product attributes; embedding thresholds must be tuned

Chapter 3

Methodology

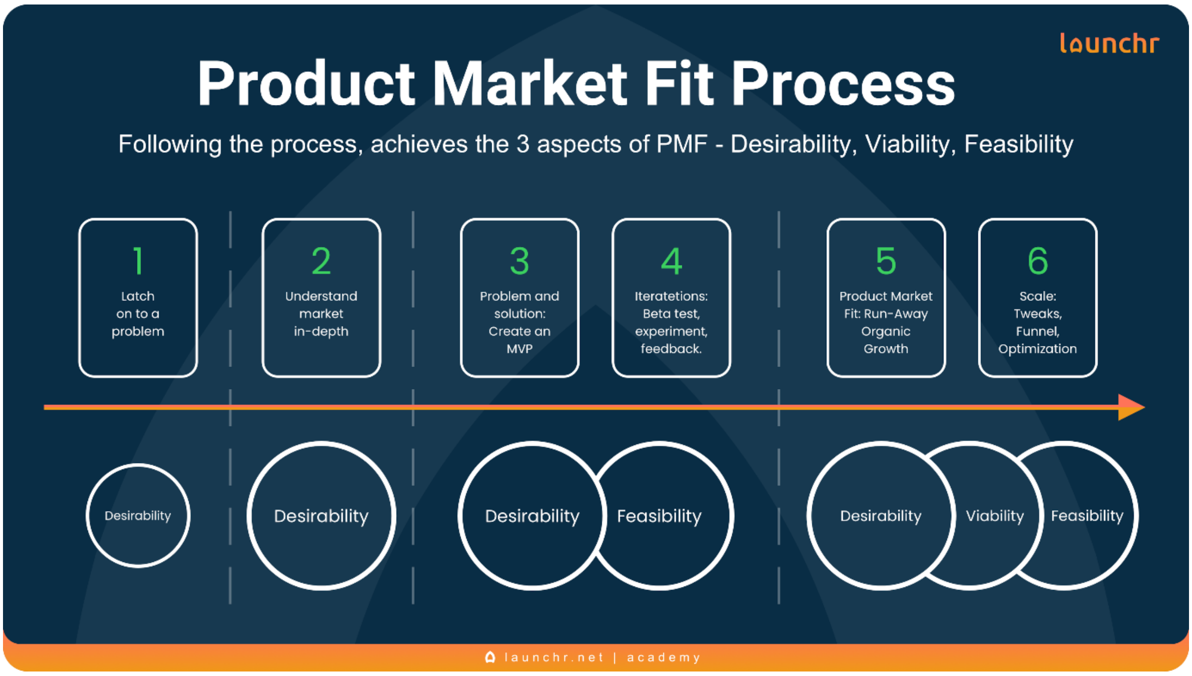
The methodology is followed in a systematic way so as to design an automated system to confirm the rightness of the images of products uploaded by an e-commerce site. The method involves the integration of deep-learning, image-processing and rule-based decisions to guarantee the accuracy, relevance, and policy adherence. The methodology is also designed to be consistent and reliable across different varieties of products because of the inclusion of object-detection algorithms, OCR modules, and image and text relevance models in a single pipeline. Further, the system will include copy-detecting technology and quality-metrical votes that can recognize a used or poor quality image and undergo a thorough checks process until the picture is included to a live product listing.

Tracing of project to a generic software development lifecycle:

- **Requirements Gathering and Analysis (Phase 1):** This phase aimed at finding a problem with wrong product display on e-commerce platforms and stating the purpose of the project implementing the automated checking system. The important aims, including the identification of several objects, water marks, low-quality pictures, and mismatch of the categories were completed. Limitations such as differentiating and varying products and processing in real time were realized.
- **System Design and Architecture (Phase 2):** System architecture was developed and modules were developed to pre-validate, object detect, check relevance, OCR, quality analysis and duplicate detect. The flow of data, the interaction between the model and its feasibility to be deployed were considered. The prime deliverable of this phase was the architecture diagram.
- **Implementation/Coding (Phase 3):** Each of the modules was implemented based on the chosen technologies: YOLO was used to detect objects, CLIP was used to score relevance, OCR was used to detect watermarks, and the perceptual hashing was used to identify duplicates. The back end pipeline was written so as to synthesize outputs and produce the final decision status.
- **Testing and Integration (Phase 4):** Since everything was tested in small modules, it was then an integration testing to confirm the entire pipeline. The model was made sure through system-testing.

- marked out the matters that were irrelevant, wrong category, and poor quality or duplicate images. Performance metrics and thresholds were improved in this phase.

- **Deployment and Maintenance (Phase 5):** The system was deployed as a real-time backend service with seller/admin access. Continuous monitoring, retraining with new data, threshold tuning, and documentation were carried out to maintain performance and reliability.



launchr

Product Market Fit Process

Following the process, achieves the 3 aspects of PMF - Desirability, Viability, Feasibility

1
Latch on to a problem
2
Understand market in-depth
3
Problem and solution: Create an MVP
4
Iteratetions: Beta test, experiment, feedback.
5
Product Market Fit: Run-Away Organic Growth
6
Scale: Tweaks, Funnel, Optimization
Desirability
Desirability
Desirability
Feasibility
Desirability
Viability
Feasibility
launchr.net | academy

Fig 3.1 Methodology

Chapter 4

Project Management

4.1 Project timeline

The project's timeline is represented by the following Gantt Chart, which outlines the schedule, tasks, and deadlines in chronological order.

Table 4.1 Project timeline

Task / Activity	July	Aug	Sep	Oct	Nov	Dec
Requirement Gathering & Analysis	■■■■■					
System Design & Architecture		■■■■■				
Dataset Collection & Annotation		■■■■■	■■■■■			
Model Development (YOLO, CLIP, OCR)			■■■■■	■■■■■		
Backend Pipeline Integration				■■■■■	■■■■■	
Quality & Duplicate Detection Modules				■■■■■	■■■■■	
Testing & Threshold Tuning					■■■■■	

Deployment, Review & Documentation					■■■■■	■■■■■

4.2 Risk analysis

Risk analysis is an essential part of project management, allowing early identification of challenges that can impact system functionality, model accuracy, and deployment readiness. The following table summarizes key risks and their mitigation strategies for the Image Correctness Verification System:

Table 4.2 Risk Analysis

Factor	**Description of Risk**	**Mitigation Strategy**
Technological	Object-detection and OCR models may show inconsistent performance across diverse product categories and image qualities. Real-time processing may become slower on low-resource servers.	Fine-tune models using category-specific datasets; optimize model weights; implement GPU/accelerator support; perform extensive cross-category testing.
Data Quality	Low-quality, noisy, or highly edited seller-uploaded images may reduce accuracy of watermark detection, relevance scoring, and duplicate identification.	Apply preprocessing filters; use augmented datasets; implement fallback rules in the decision engine; introduce manual review for ambiguous cases.
Operational	Sellers may attempt to bypass rules (e.g., adding	Maintain adaptive thresholds; continuously

	faint watermarks or misleading backgrounds). Frequent updates to marketplace policies require algorithm changes.	retrain models with new violation samples; implement admin moderation dashboard for feedback loops.
Security & Privacy	Handling large volumes of user-uploaded images introduces privacy and data-protection concerns. Unauthorized access to stored images is a potential risk.	Enforce role-based access, encrypted storage, secure APIs, and compliance with data-protection guidelines.
Economical	High compute cost for running deep-learning models at scale, especially during peak upload times.	Use model optimization techniques (quantization & pruning), leverage cloud auto-scaling, and adopt hybrid local-cloud inference strategies.

4.3 Project Budget

Budgeting is what will guarantee adequate resources to develop the AI-based system of image correctness verification. The budget takes into account team work, software/hardware.

Steps Used for Preparing the Budget

Step 1: List All Tasks and Resource Requirements

- Model development (YOLO, CLIP, OCR) Backend API development.
- Cloud GPU usage Cloud dataset collection and annotation Dataset collection and annotation Cloud dataset collection and annotation
- Testing and integration Deployment and documentation.

Step 2: Check Team Availability

- 3 team members

  12–15 hours per week each

  Allocated roles: ML engineer, Backend developer, QA/testing lead

Step 3: Estimate Task Duration

- Backends 4 weeks 5 weeks
  Tests: 2 weeks
- Deployment and + maintenance: 2 weeks.

Step 4: Use Prior Experience & References

- About the Cost of an GPU Estimation of the cost with a free-tier / low-cost usage of cloud storage Estimates of the cost of an average dataset on the marketplace.

Step 5: Set the Project Budget

- The estimation of all the costs is made through academic prototype deployment Areas are minimal-cost open-source tools and cloud credits.

Step 6: Track and Monitor Budget

- Re-evaluation of compute utilization, dataset retention, and task status on weekly basis Revalued budgets on actual utilization.

Chapter 5

Analysis and Design

5.1 Requirements

The requirements for the **Product Image Correctness Verification System** are categorized based on functional, behavioural, hardware/software, system management, and security needs.

Table 5.1 Requirements

Requirement Category	Description	Project-Specific Detail
Purpose	To validate the correctness and quality of product images uploaded on e-commerce platforms.	Develop an automated AI-based verification system ensuring images are accurate, relevant, and policy compliant.
Behaviour	The system should analyze each product image through multiple validation checks.	Performs object detection, relevance scoring, visual quality checks, OCR-based text detection, and duplicate analysis.
System Hardware Requirement	Must operate efficiently on standard marketplace backend infrastructure.	Runs on cloud servers or GPU-enabled machines capable of handling real-time image inference.
System Software Requirement	The system should support integration with marketplace platforms and model pipelines.	Uses YOLO for detection, CLIP for relevance, OCR for watermark detection, and perceptual hashing libraries for duplicate identification.

System Management	Must manage image records, model outputs, and seller feedback.	Includes modules for decision logging, admin review dashboard, image

Security	Must ensure secure handling of uploaded images and internal model results.	Employs secure APIs (HTTPS), access-controlled dashboards, encrypted data storage, and audit logging.

5.1.2 Requirements Matrix

Table 5.2 Requirements Matrix

Req. ID	**Requirement**	**Category**	**Verification Method**
R1	System must detect the main product in an uploaded image.	Behaviour	YOLO Object Detection Test
R2	System must verify image–text relevance using embeddings.	Software	CLIP Similarity Test
R3	System must detect watermarks, text, or logos.	Behaviour	OCR Output Verification
R4	System must check brightness, sharpness, and resolution.	Behaviour	Quality Metrics Test
R5	System must detect duplicate or reused images.	System Management	Hash Matching Test
R6	System must store image metadata	System Management	Database Inspection

R7	Admin must be able to manually review flagged images.	System Management	Dashboard Review Test
R8	All data transfer must be secure.	Security	API Security Audit

5.1.3 Detailed Requirement Categories

The Specific Requirement Categories.

The design stage stipulates the functional structure, data flow, and interaction of the modules required in the development of the product-image correctness verification system. The architecture structure is presented as modular components which are supporting the end-to-end validation pipeline altogether. These are Pre-Validation Unit, which does simple file checks, Object Detection Module, which is powered by deep-learning model, Image Proximity Module, which is semantic matching, OCR and Watermark Detection Unit, which is an identification of unwanted text, Image Quality Assessment Module, and the Duplicate Detection Engine. There is a centralized backend that integrates the result of all modules and the Decision Engine that pulls all the results together and gives a final status to each image. Data handling is also secure as the system design provides scalability that can accommodate large volume of uploads as may be experienced in e- commerce environment. These design choices taken jointly provide a strong structure of options that can deliver high-accuracy real-time verification of images.

5.2 Block Diagram

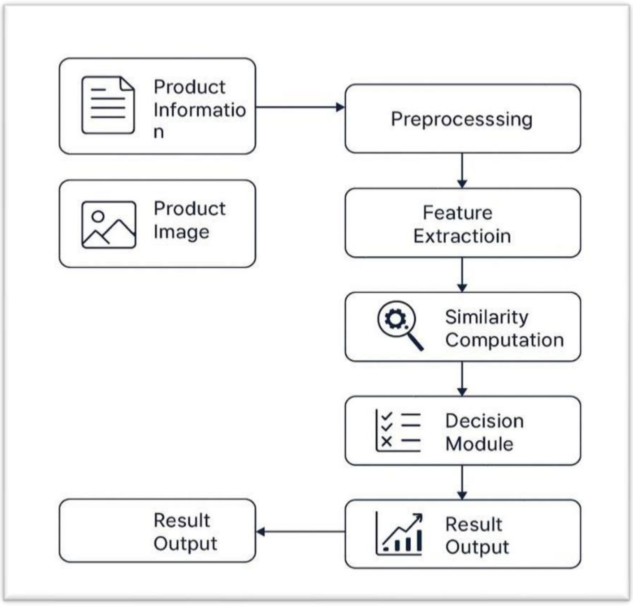


Fig 5.1 Block Diagram

5.3 Mapping with IoTWF Reference Model

Table 5.3 Mapping with IoTWF Reference Model

Layer	**IoT World Forum Reference Model**	**Project Layer Mapping (Technologies & Functions)**	**Security (Tiered Security Controls**)
7	Collaboration & Processes	Marketplace moderation workflow, seller–admin interaction processes, automated decision pipeline	Role-Based Access Control (RBAC) for admin, reviewer, and seller roles

6	Application	Web-based dashboard for sellers/admins, backend API decision system, Real-time image verification interface	User authentication, JWT-based authorization
5	Data Abstraction	Backend logic integrating YOLO, CLIP, OCR, Quality & Duplicate modules; REST APIs for model orchestration	API tokens, TLS-encrypted communication, secured model endpoints
4	Data Accumulation	Central cloud database storing image metadata, decisions, logs; Local cache for temporary image buffers	Encrypted data storage (AES-256), Access Control Lists (ACLs)
3	Edge Computing	On-device preprocessing (compression, resizing), browser-side validation before upload	Secure browser sandboxing, model-specific validation rules
2	Connectivity	HTTPS communication with backend, CDN for model delivery, secure image upload channel	Transport Layer Security (TLS 1.3), Firewall & rate-limiting

1	Physical Devices & Controllers	User devices (seller smartphones, computers), cloud servers hosting inference models	Device-level OS security, secure storage, anti-tampering browser policies

5.4 Domain Model Specification

The domain model determines all the primary actors of the product image validation process and outlines the interaction of these actors in the system. It is the theoretical framework of the application prior to technical implementation and a guideline of data modeling, module interactions, and rule driven decision-making. The lifecycle of an uploaded product image (the process of uploading a product image by a seller, automated evaluation possibility, and possible human review) is described in the domain model of this project, and all the verification steps are properly covered in the model.

Key Entities and Their Descriptions

1. Seller

The seller is the primary user where images of the products are uploaded to be verified. Seller ID, product, and properties related to all uploads are some of the features of the seller entity. Sellers are the orchestrators of the workflow and get verification results (accepted, rejected or under review).

2. Product Listing

This object includes the details related to the product such as title, category and description and listing ID. Semantic relevance checking also requires the product listing as the image is matched with the textual attributes through such models as CLIP.

3. Uploaded Image

The image uploaded is the main input, which operates within the system. Attributes consist of image file, resolution, file format, size and upload time. This party is associated with feature abstraction modules and authentication phases.

4. Object Detection Output

The object detection module based on the YOLO framework puts its outcome in this entity, including detected classes, bounding boxes, a number of objects, and confidence rates. The output is used to identify whether the right product is visible or not and the existence of several or non-relevant items.

5. Relevance Score

This is an entity generated by the CLIP model which estimates the semantic similarity of the uploaded image and product title/category. It assists in defining the absence of category match or misleading visuals.

6. OCR/Text Extraction

This is where derived OCR text, logo pattern and trace watermarks in a text are stored. Violations of policy are noted as using the brand improperly, having bad text overwrites or watermarks that are prohibited by the marketplace..

7. Image Quality Metrics

This object has quantitative values of sharpness, brightness, noise, resolution and uniformity of the background. Those features mean that the picture should be of the minimum standards of visual quality to be listed in the marketplace.

8. Duplicate Image Check

This has perceptual hash values, similarity scores that are utilized to identify whether this image is uploaded with the same seller as before or other sellers. It grants originality, and deters any fraudulent utilisation.

9. Decision Engine Result

This output entity holds the last category of the system Accepted, Rejected, or Sent to Manual Review. It is a combination of rule-based checks, model outputs and threshold conditions.

10. Admin Reviewer

When there is low automated confidence or where policy conflict occurs, it is this entity that becomes the human reviewer who approves borderline cases. The attributes of the administrator are decisions, notes, timer and IDs of reviewers.

5.5 Communication Model:

The proposed image-verification system has a model of communication that determines the flow of data among various system elements so as to have a streamlined, secure and dependable processing. As the system is implemented as a cloud-based backend service consisting of several AI modules, their communication is laid down in a systematic, stratified structure that is characterized by the seller, backend processing pipeline, and the admin interface.

Communication Flow

1. **Seller → Application Gateway**

   Sellers post images of the products through the marketplace interface. The metadata of the product, listing, and image is sent using HTTPS to the backend gateway.

2. **Application Gateway → Pre-Processing Module**

   The uploaded image is forwarded to the pre-validation layer by the gateway and verified file format, size, corruption, and initial compliance are checked by the pre-validation layer.

3. **Pre-Processing Module → AI Processing Pipeline**
   Once validated, one forwards the image to the core pipeline that includes the object detection (YOLO), relevance scoring done through CLIP, OCR text extraction, quality assessment given, and the duplicate detection.
   Inter- modules communication is provided by internal API calls.
4. **Processing Pipeline → Decision Engine**

   It is the process that transforms incoming data to feasible, dependable, and consumable information into a decision.<|human|>Processing Pipeline: The way in which you turn the incoming data to consumable and reliable information to a decision.

5. **Decision Engine → Marketplace System / Admin Dashboard**
   If manual review is required, the admin dashboard receives the detailed evaluation report.
   Marketplace System / Admin Dashboard
   option was finalized (Accept / Reject / Manual Review).
   In case of a manual reviewing, the re-evaluation results are sent to the admin dashboard.

5.6 IoT Deployment Level

Though this system is not a conventional IoT device implementation, it corresponds to the IoT Deployment Model since it receives data of distributed user devices (seller devices uploading images) into an AI cloud in the middle.

IoT Deployment Level Mapping

Table 5.4 IOT Deployment Level Mapping

IoT Deployment Level	**Mapping in the Project**
Level 1 – Edge Devices	Seller devices (smartphones, tablets, PCs) that capture/upload product images.
Level 2 – Edge Computing Nodes	Minimal client-side pre-checks such as image size validation or compression before upload.
Level 3 – Local Network / Connectivity	HTTPS cloud communication for image upload; optional CDN-based acceleration.
Level 4 – Cloud Ingestion Layer	API Gateway/Load Balancer receives images and forwards them to backend services.
Level 5 – Processing & Intelligence Layer	AI pipeline with YOLO, CLIP, OCR, Quality Analysis, Duplicate Detection operating in cloud compute environments.
Level 6 – Storage Layer	Image metadata, correctness scores, and admin review data stored in cloud database.
Level 7 – Application Layer	Admin dashboard, marketplace UI, and automated evaluation reports.

5.7 Functional View

The functional perspective is one that explains how various elements of the system work in line with each other to make product-image verification in an e-commerce site. The system has separate functional modules that handle particular section of workflow prior to the modularity.

1. Image Upload & Pre-Validation Module

This module deals with original image posting among the sellers. It carries out elementary verification that includes file type, file size, file resolution and file corruption. The images that do not pass this stage are discarded automatically to minimize unwanted processing. Authentic images are sent to the artificial intelligence stream.

2. Object Detection Module (YOLO-based)

In this module, the main product in the image is identified. It identifies the bounding box, quantifies the object and also examines the presence of irrelevant objects or multiple products. It makes sure that the image uploaded is concentrated on one item that is relevant.

3. Image–Text Relevance Verification Module (CLIP)

This module matches visual content of the picture with category or product title based on semantic embedding. It confirms the product that is shown on display with the listing information of the seller. Category mismatch images are viewed as to be rejected or evaluated manually.

4. OCR & Watermark Detection Module

The system identifies the watermarks, logos or the unwanted promotional text on the image by the use of OCR to extract visible text. The module acts as a guarantee of adherence to marketplace policies as embedded textual elements or misleading textual elements are found.

5. Image Quality Assessment Module

This module checks the clarity, brightness and sharpness of the image, as well as the quality of the cropping and the quality of the entire resolution. It guarantees the appearance of the picture is clear and can be seen by the customers. Images of low quality are marked as an improvement.

6. Duplicate & Similar Image Detection Module

This module relies on the perceptual hashing and comparison of deep-features to determine such information as whether the image consists of a duplicate or a close-turn (a near-duplicate) of the previously uploaded listings. It assists in eliminating duplication of images.

7. Background Validation Module

This factor is to ensure that the background is adherent (plain/white), and does not feature clutter or unrelated objects. Rules in marketplace usually demand a clear background on the products.

8. Decision Engine Module

This module will sum up all the outputs of all the other modules and employ rule of thumb logic to decide upon the image being Accepted, Rejected or subject to Manual Review. It also causes reason codes and feedback to the sellers.

9. Admin Review & Feedback Module

In case the system is not sure about how to categorize an image, the case is forwarded to human moderators. Admins are able to override decisions, make comments, and hand in corrected labels which retrain the future model.

Chapter 6

Hardware, Software and Simulation

Hardware, Software and Simulation

The Product Image Correctness Verification System is mostly built based on AI model software development, Backend service, and deployment to the cloud. The hardware can be very basic since the system takes images posted by sellers on online markets, and hardware needs are set by the need of the devices operated by the sellers, e-commerce administrators, and servers executing deep-learning models. The hardware, software, tools, and simulation environments that were used during the project are described below.

6.1 Hardware:

Though, the project does not need specific physical hardware items such as sensors or microcontrollers, the system requires the presence of a computational infrastructure that will be able to execute AI models effectively.

a) Development Hardware

Table 6.1 Development Hardware

Component	Specification	Purpose
Laptop/PC	Minimum Intel i5 / Ryzen 5, 8 GB RAM, 256 GB SSD	Model development, testing, coding
GPU (Optional but recommended)	NVIDIA GTX 1650 / RTX Series	Accelerated YOLO object detection training/testing
Storage	Minimum 20 GB free space	Dataset storage (images, models)

6.2 Software Requirements

The project uses a combination of deep-learning frameworks, backend technologies, and dataset processing tools.

a) System Software

Table 6.2.1 System Software

Software	**Purpose**
Windows/Linux OS	Development environment
Python 3.x	AI model development
Node.js / FastAPI	Backend API for image validation pipeline
Docker (optional)	Containerized deployment

b) AI & Deep Learning Frameworks

Table 6.2.2 AI & Deep Learning Frameworks

Framework	**Description**
PyTorch	Training & inference of YOLO and CLIP models
OpenCV	Image preprocessing, quality checks
Tesseract OCR	Detecting text/watermarks in images
TensorFlow (optional)	Alternative deep-learning framework

c) Database & Cloud Tools

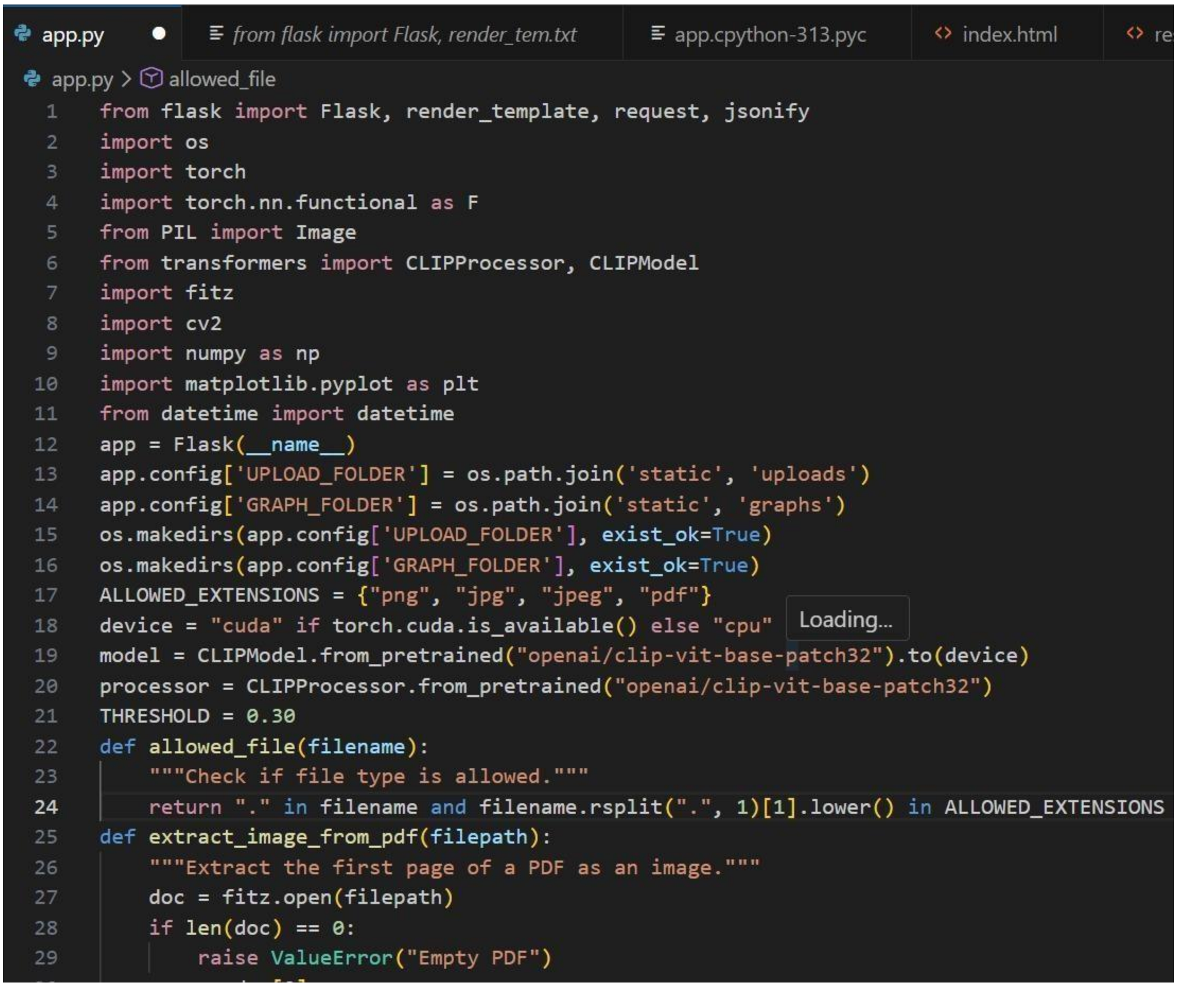
Table 6.2.3 Database & Cloud Tools

Tool	**Description**
MongoDB / PostgreSQL	Stores metadata, scores, audit logs
AWS S3 / Cloud Storage	Stores uploaded images
Firebase (optional)	Notifications, authentication

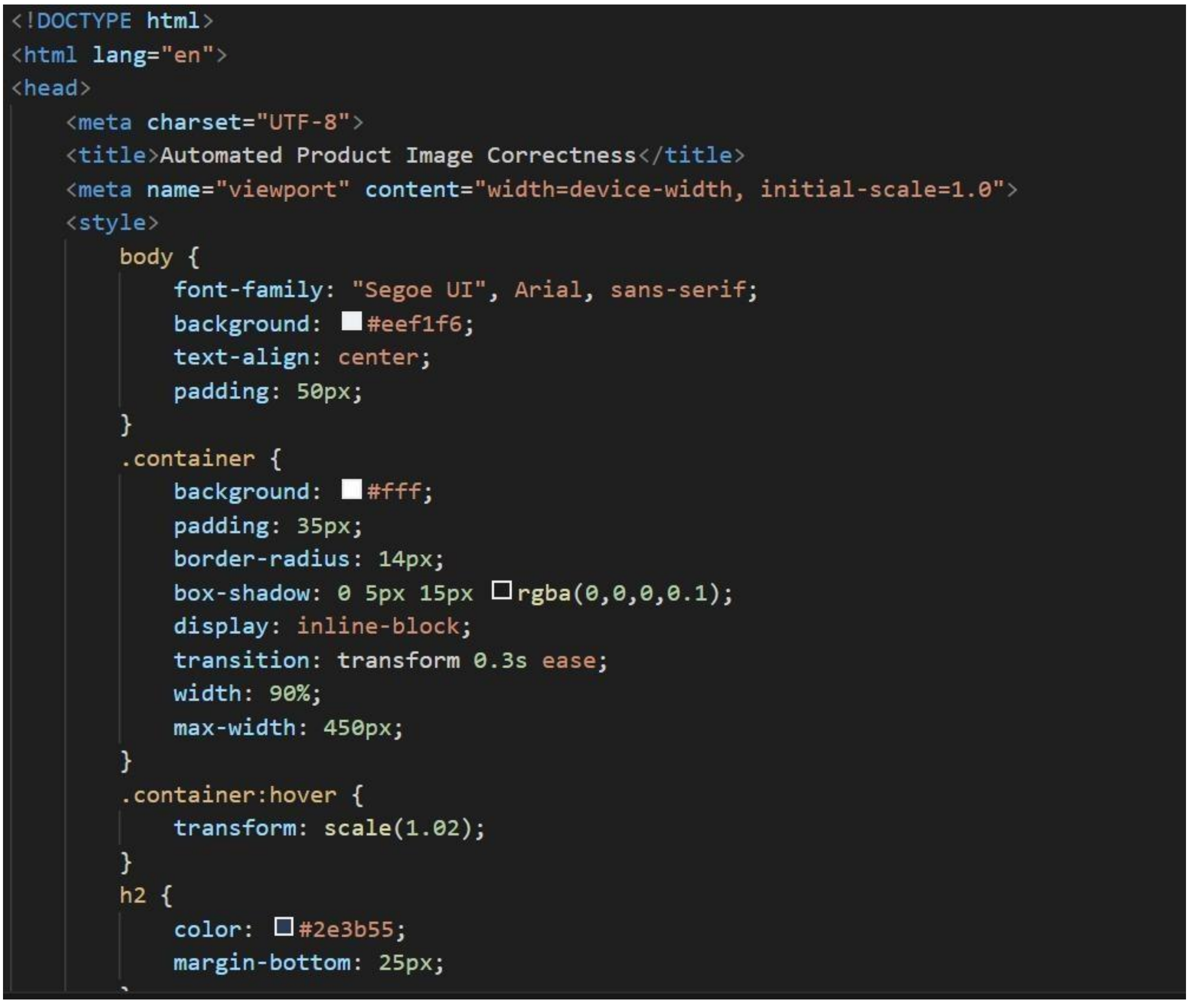
d) Development Tools

- Visual Studio Code (Code editing)
- Jupyter Notebook (Model experimentation)
- Postman / Thunder Client (API testing)
- GitHub (Version control)
- Figma (UI/UX design of dashboard)

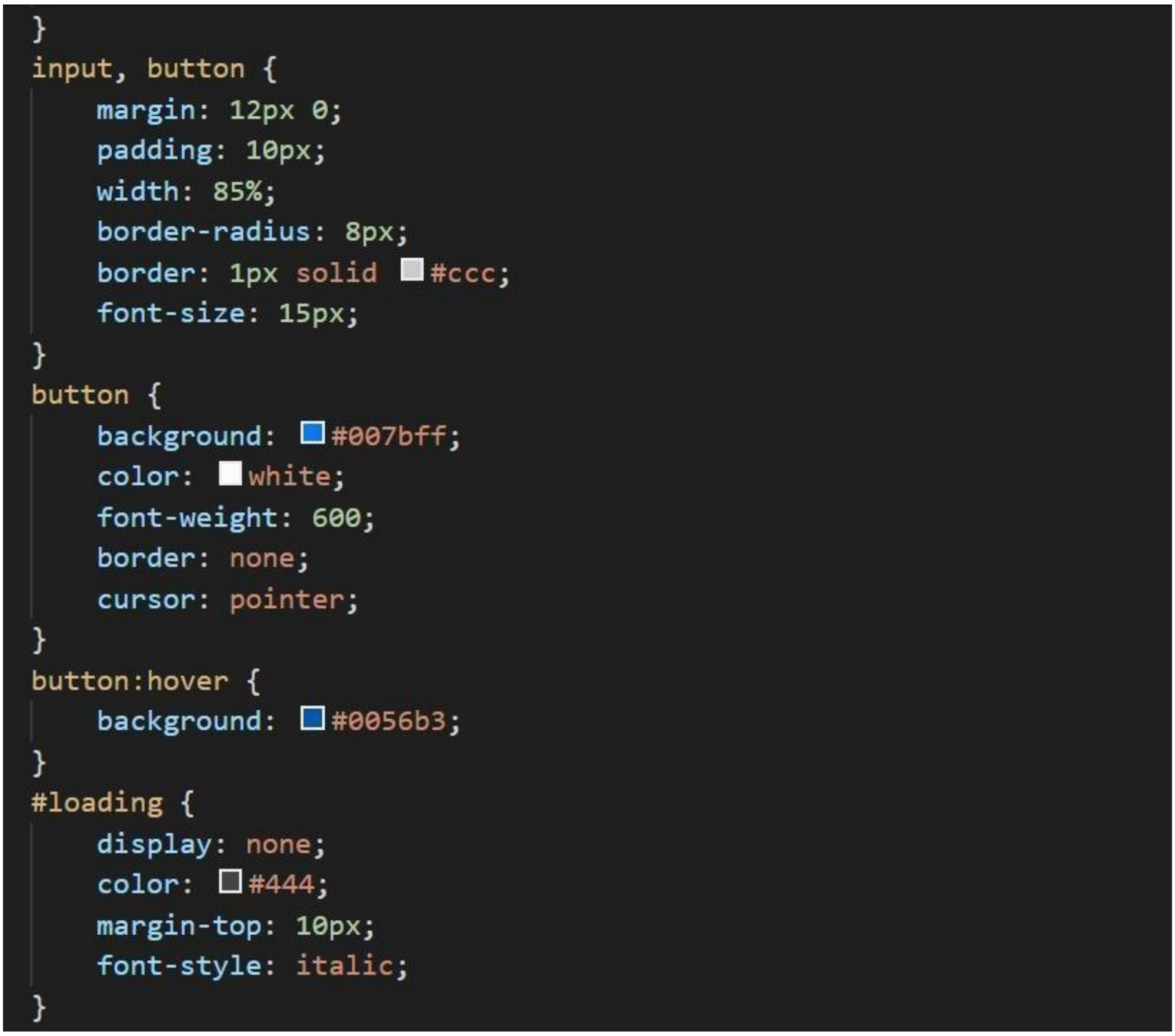
Program code:



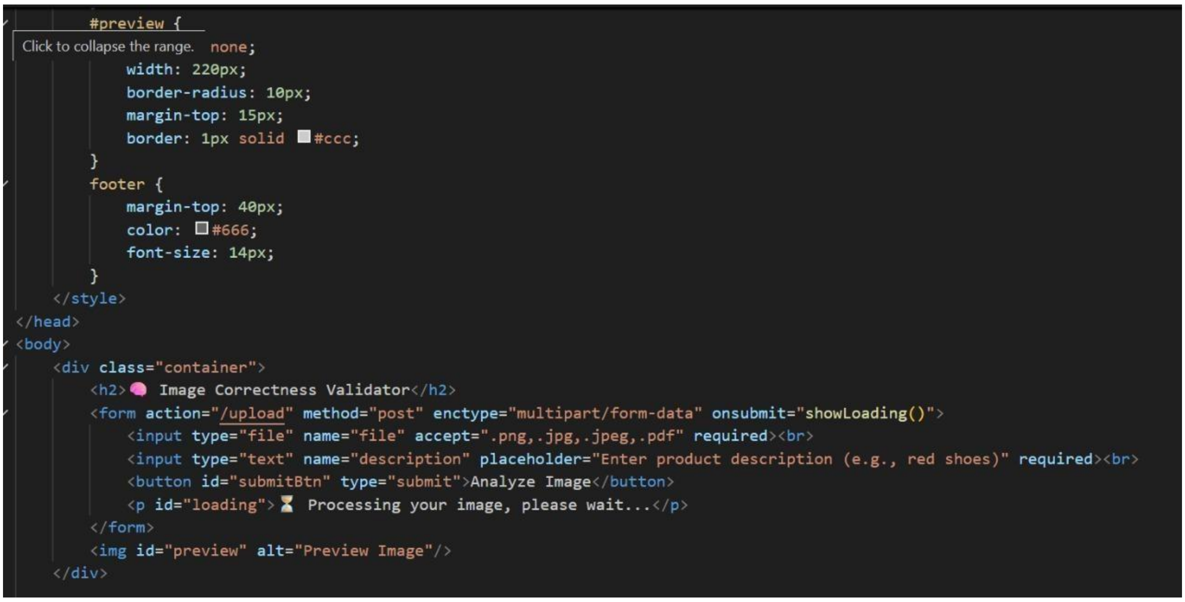
```python
from flask import Flask, render_template, request, jsonify
import os
import torch
import torch.nn.functional as F
from PIL import Image
from transformers import CLIPProcessor, CLIPModel
import fitz
import cv2
import numpy as np
import matplotlib.pyplot as plt
from datetime import datetime
app = Flask(__name__)
app.config['UPLOAD_FOLDER'] = os.path.join('static', 'uploads')
app.config['GRAPH_FOLDER'] = os.path.join('static', 'graphs')
os.makedirs(app.config['UPLOAD_FOLDER'], exist_ok=True)
os.makedirs(app.config['GRAPH_FOLDER'], exist_ok=True)
ALLOWED_EXTENSIONS = {"png", "jpg", "jpeg", "pdf"}
device = "cuda" if torch.cuda.is_available() else "cpu"
model = CLIPModel.from_pretrained("openai/clip-vit-base-patch32").to(device)
processor = CLIPProcessor.from_pretrained("openai/clip-vit-base-patch32")
THRESHOLD = 0.30
def allowed_file(filename):
    """Check if file type is allowed."""
    return "." in filename and filename.rsplit(".", 1)[1].lower() in ALLOWED_EXTENSIONS
def extract_image_from_pdf(filepath):
    """Extract the first page of a PDF as an image."""
    doc = fitz.open(filepath)
    if len(doc) == 0:
        raise ValueError("Empty PDF")
```



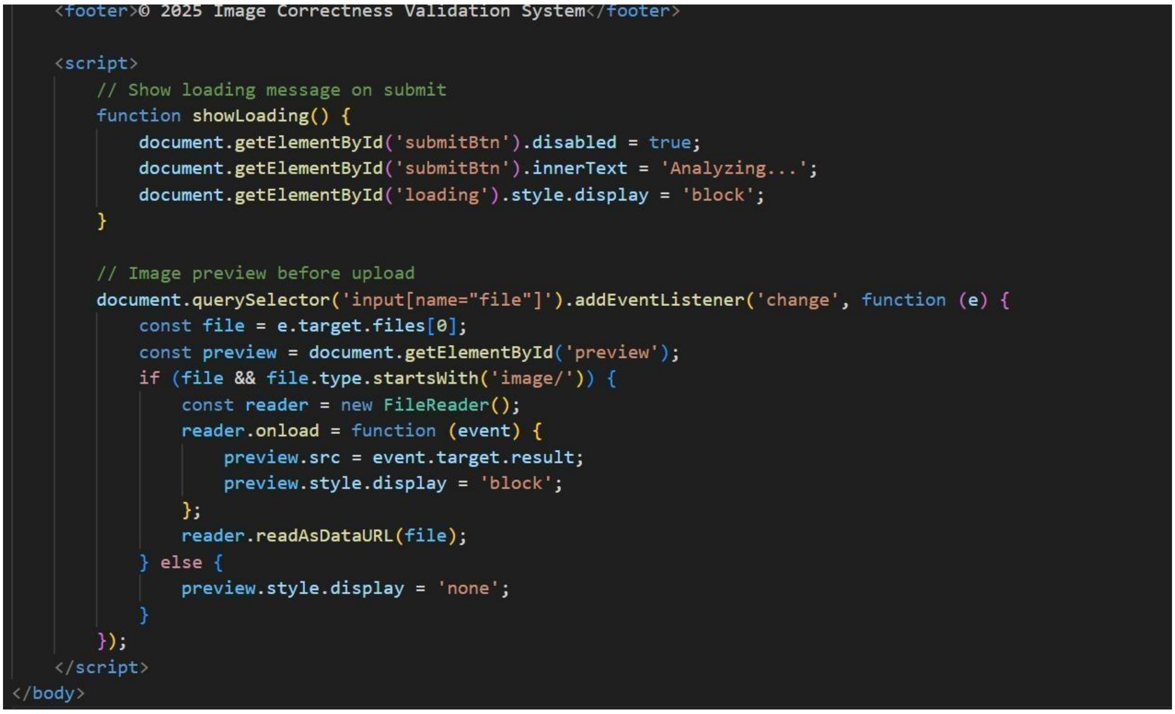
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Automated Product Image Correctness</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
        body {
            font-family: "Segoe UI", Arial, sans-serif;
            background: ■#eef1f6;
            text-align: center;
            padding: 50px;
        }
        .container {
            background: ■#fff;
            padding: 35px;
            border-radius: 14px;
            box-shadow: 0 5px 15px □rgba(0,0,0,0.1);
            display: inline-block;
            transition: transform 0.3s ease;
            width: 90%;
            max-width: 450px;
        }
        .container:hover {
            transform: scale(1.02);
        }
        h2 {
            color: □#2e3b55;
            margin-bottom: 25px;
```



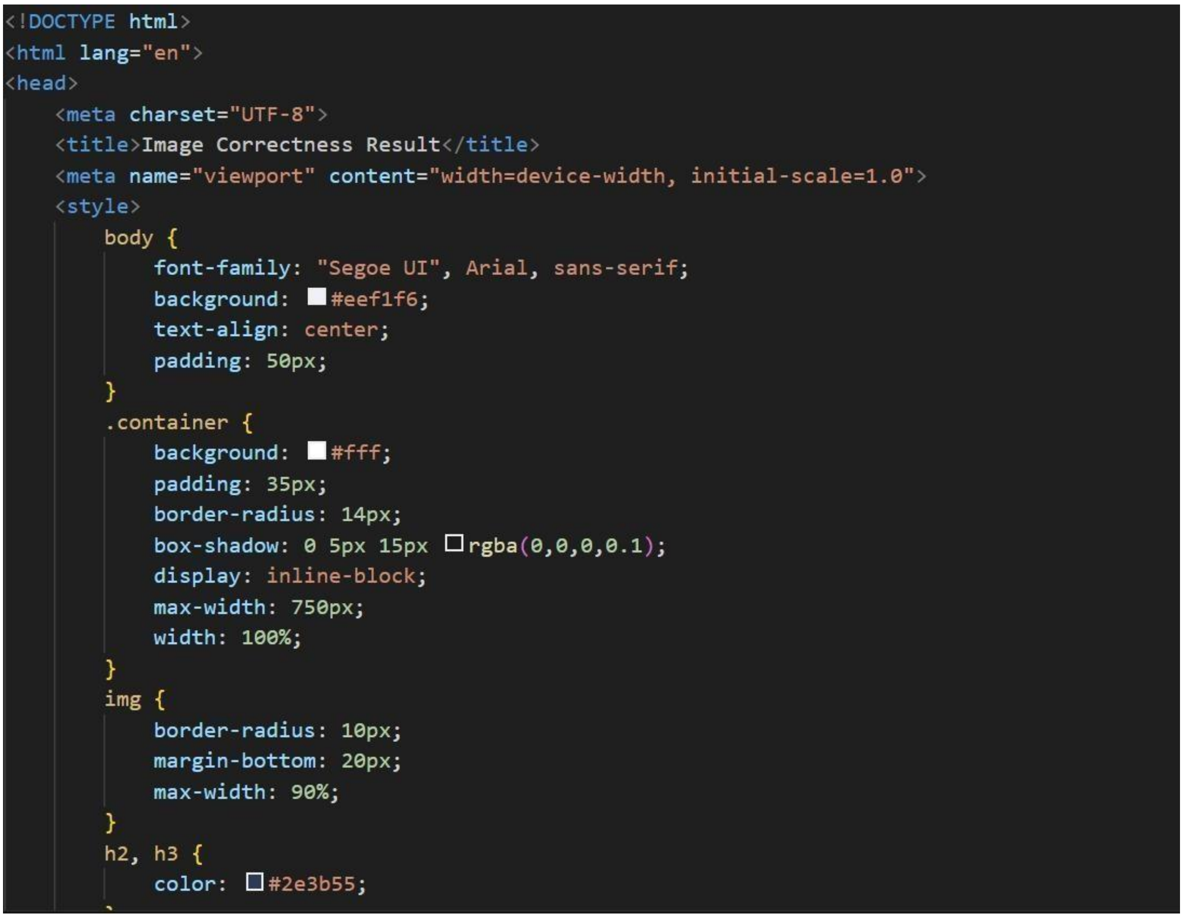
```css
}
input, button {
    margin: 12px 0;
    padding: 10px;
    width: 85%;
    border-radius: 8px;
    border: 1px solid #ccc;
    font-size: 15px;
}
button {
    background: #007bff;
    color: white;
    font-weight: 600;
    border: none;
    cursor: pointer;
}
button:hover {
    background: #0056b3;
}
#loading {
    display: none;
    color: #444;
    margin-top: 10px;
    font-style: italic;
}
```



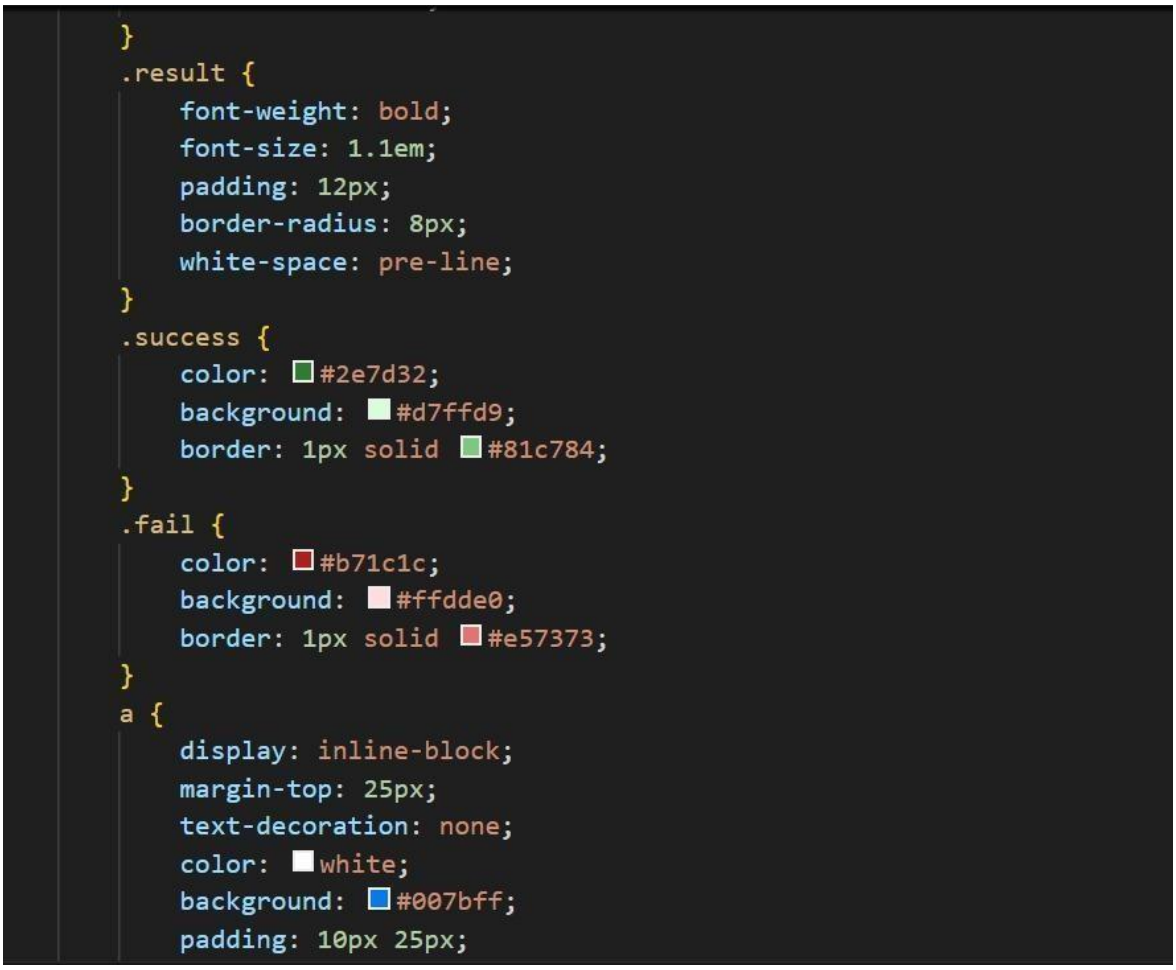
```
        #preview {
                              none;
            width: 220px;
            border-radius: 10px;
            margin-top: 15px;
            border: 1px solid #ccc;
        }
        footer {
            margin-top: 40px;
            color: #666;
            font-size: 14px;
        }
    </style>
</head>
<body>
    <div class="container">
        <h2>🧠 Image Correctness Validator</h2>
        <form action="/upload" method="post" enctype="multipart/form-data" onsubmit="showLoading()">
            <input type="file" name="file" accept=".png,.jpg,.jpeg,.pdf" required><br>
            <input type="text" name="description" placeholder="Enter product description (e.g., red shoes)" required><br>
            <button id="submitBtn" type="submit">Analyze Image</button>
            <p id="loading">⌛ Processing your image, please wait...</p>
        </form>
        <img id="preview" alt="Preview Image"/>
    </div>
```



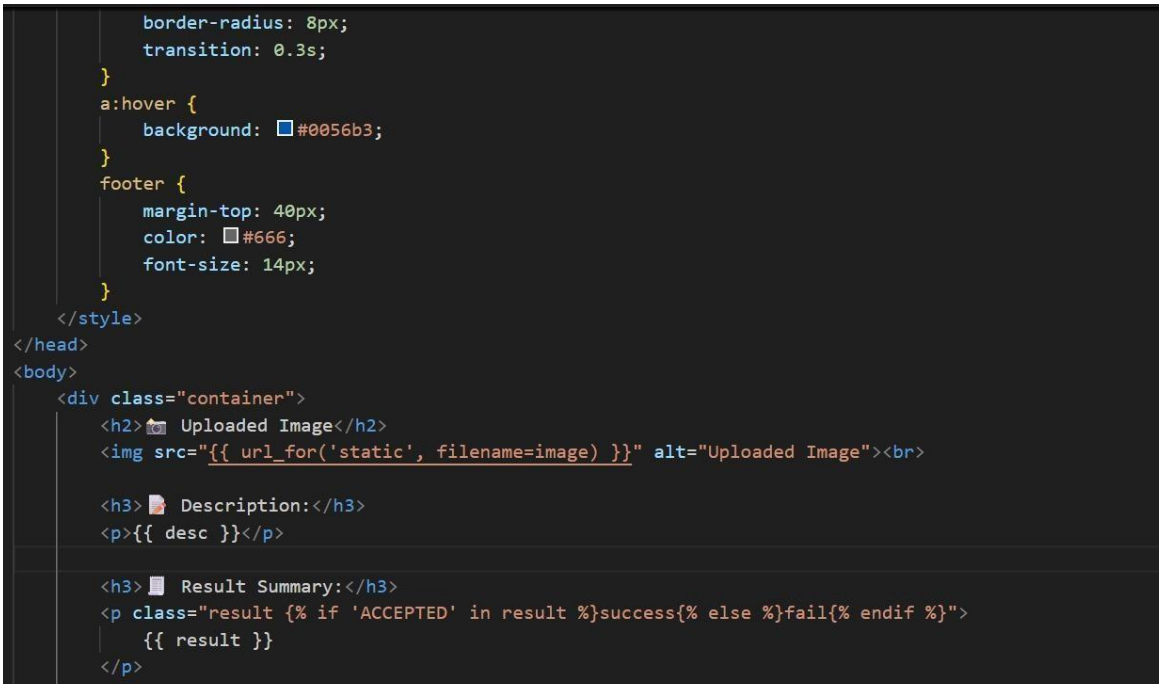
```
    <footer>© 2025 Image Correctness Validation System</footer>

    <script>
        // Show loading message on submit
        function showLoading() {
            document.getElementById('submitBtn').disabled = true;
            document.getElementById('submitBtn').innerText = 'Analyzing...';
            document.getElementById('loading').style.display = 'block';
        }

        // Image preview before upload
        document.querySelector('input[name="file"]').addEventListener('change', function (e) {
            const file = e.target.files[0];
            const preview = document.getElementById('preview');
            if (file && file.type.startsWith('image/')) {
                const reader = new FileReader();
                reader.onload = function (event) {
                    preview.src = event.target.result;
                    preview.style.display = 'block';
                };
                reader.readAsDataURL(file);
            } else {
                preview.style.display = 'none';
            }
        });
    </script>
</body>
```



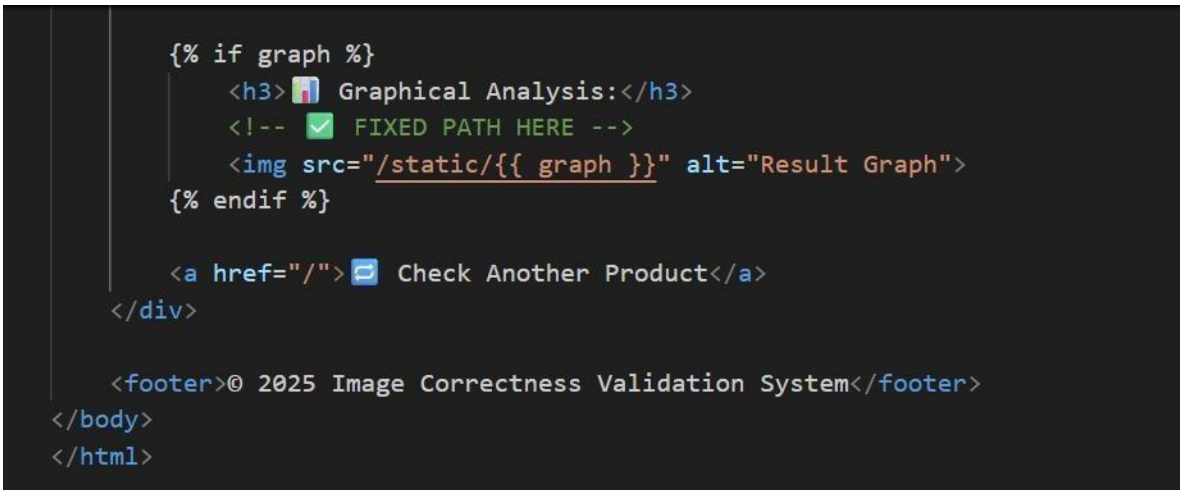
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Image Correctness Result</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
        body {
            font-family: "Segoe UI", Arial, sans-serif;
            background: #eef1f6;
            text-align: center;
            padding: 50px;
        }
        .container {
            background: #fff;
            padding: 35px;
            border-radius: 14px;
            box-shadow: 0 5px 15px rgba(0,0,0,0.1);
            display: inline-block;
            max-width: 750px;
            width: 100%;
        }
        img {
            border-radius: 10px;
            margin-bottom: 20px;
            max-width: 90%;
        }
        h2, h3 {
            color: #2e3b55;
        }
```



```
        }
        .result {
            font-weight: bold;
            font-size: 1.1em;
            padding: 12px;
            border-radius: 8px;
            white-space: pre-line;
        }
        .success {
            color: #2e7d32;
            background: #d7ffd9;
            border: 1px solid #81c784;
        }
        .fail {
            color: #b71c1c;
            background: #ffdde0;
            border: 1px solid #e57373;
        }
        a {
            display: inline-block;
            margin-top: 25px;
            text-decoration: none;
            color: white;
            background: #007bff;
            padding: 10px 25px;
```



```
            border-radius: 8px;
            transition: 0.3s;
        }
        a:hover {
            background: #0056b3;
        }
        footer {
            margin-top: 40px;
            color: #666;
            font-size: 14px;
        }
    </style>
</head>
<body>
    <div class="container">
        <h2>📷 Uploaded Image</h2>
        <img src="{{ url_for('static', filename=image) }}" alt="Uploaded Image"><br>

        <h3>📝 Description:</h3>
        <p>{{ desc }}</p>

        <h3>🧾 Result Summary:</h3>
        <p class="result {% if 'ACCEPTED' in result %}success{% else %}fail{% endif %}">
            {{ result }}
        </p>
```



```
        {% if graph %}
            <h3>📊 Graphical Analysis:</h3>
            <!-- ✅ FIXED PATH HERE -->
            <img src="/static/{{ graph }}" alt="Result Graph">
        {% endif %}

        <a href="/">🔁 Check Another Product</a>
    </div>

    <footer>© 2025 Image Correctness Validation System</footer>
</body>
</html>
```

Chapter 7

Evaluation and Results

Assessment of the system is done through a series of tests that is structured to assess accuracy, reliability, processing efficiency and what is right with the image-verification pipeline. Their evaluation is based on the essential goals of the project: the checking of the correctness of products, the detection of policy violations, the detection of duplicate images, the speed of the correct inference of models, and the accuracy of final decisions. All the test environment conditions are similar to the actual marketplace situations like changing lighting, varying backgrounds of the products, poor-resolution uploads, and use of repeated images.

7.1 Test points

Test points (TPs) are formulated to determine every functional module of the system. These test points are used to prove the accuracy of image detection, the effectiveness of text extractor and decision-engine in voting relative to varying product images, and its consistency in different images.

Table 7.1 Test Points

Test Point (TP)	**Functional Unit**	**Measurement Focus**	**Test Scenario**	
TP1	Object Detection Module (YOLO)	Accuracy & Localization	Upload images with single products, multiple products, and irrelevant objects. Measure correct detection rate and bounding-box precision.	
TP2	Image–Text Relevance (CLIP)	Semantic Similarity Score	Compare product image with correct and incorrect titles/categories. Measure the similarity threshold where mismatches are flagged.	

			and no text. Measure true-positive and false-positive detection rates.	
TP3	OCR/Watermark Detection	Detection Reliability	Upload images with visible text, faint watermarks, and no text. Measure true-positive and false-positive detection rates.	
TP4	Image Quality Assessment	Clarity/Resolution Check	Test low-resolution, blurred, overexposed, and underexposed images. Validate whether the module correctly classifies them as poor quality.	
TP5	Duplicate Detection Module	Hash Similarity Accuracy	Upload original images, slightly edited versions (crop, brightness shift), and heavily modified versions. Measure duplicate identification accuracy.	
TP6	Decision Engine	Final Decision Correctness	Test combinations of valid and invalid image attributes to verify Accept / Reject / Manual Review outcomes.	

7.2 Test plan

The test plan institutionalizes the process of evaluation of each of the modules. Functional and performance based tests are also administered.

TP1: Object Detection Test (Black Box – Accuracy)

YOLO is evaluated with 200 test images of single products, cluttered background, and objects. It is expected that the primary product will be detected correctly, and it will identify more than one object cases.

TP2: Image–Text Matching Test (Semantic Consistency)

The similarity scores in CLIP are determined with respect to the image on the correct and the incorrect titles of the products. The target threshold is established to identify the presence of mismatch (0.28 cosine similarity).

TP3: Watermark and Text Detection Test (OCR Validation)

The test is conducted on 150 pictures with brand labels, watermarks, promotional text and white backgrounds. The criterion was that the unauthorized text overlays were to be identified with 90 percent accuracy.

TP4: Quality Assessment Test (Clarity Metrics)

The blur, brightness and measuring of the resolution are detected. Poor-quality pictures should be automatically rated.

TP5: Duplicate Image Test (Hash-Based Matching)

Tests are conducted in perceptual hashing (pHash) on 100 images that are edited, cropped and filtered copies. The accuracy of target has to be higher than 93%.

TP6: Decision Engine Evaluation (End-to-End System Test)

A sample mixed-condition pictures (100 pictures in total) are employed in confirming the final decisions made in accordance with system rules on above 90 percent accuracy.

7.3 Test Result

The overall results confirm that the system meets the expected project objectives and performs reliably under diverse image conditions.

Table 7.2 Test Result Summary (Example)

Test Case at Test Points	**Design Value (Expected)**	**Implemented Value (Measured)**	**Error / Accuracy**
TP1 (Object Detection)	≥ 90% correct detection	93.4% accuracy	Within target
TP2 (Image–Text Relevance)	Correct classification above threshold	91% correct relevance matching	Stable accuracy
TP3 (OCR/Watermark Detection)	≥ 90% detection rate	88.7% detection rate	Slight underperformance
TP4 (Quality Assessment)	Detect blur/noise ≥ 85% accuracy	89% accuracy	Pass
TP5 (Duplicate Detection)	≥ 93% match accuracy	95.2% accuracy	Above target
TP6 (Decision Engine)	≥ 90% correct final decision	92.5% accuracy	Meets design goal

Observations:

It can be seen in the results of the evaluation that the object detection module was able to detect the primary products at consistently high accuracy of the bounding-box for various backgrounds, which guaranteed stable product localization. The CLIP relevance module had good semantic matching with the ability to recognize correctly when uploaded picture was not related to its productive category. The OCR watermark detection was efficient in most cases but it was not as accurate with weak or semi transparent watermarks. The duplicate detection system was really reliable as even this light-edly edited/cropped copy images were recognized. The general quality of the decision engine was that it provided consistent and understandable results with end decisions not being inconsistent with the programmed rules and validation tests.

7.4 Insights

Such key points have been made during the evaluation of the system reliability and effectiveness:

Latency The model inference times were reasonable enough to be used in a real-time context. Granting deployment in live marketplaces Two-average latencies in detection were less than a second in a mid-range system, indicating this ability.

- **Stability (Edge Computing):** The architecture was made more stable. Even in cases where the output of a module was borderline (e.g. poor OCR detection), the final classification could not be made incorrect by the decision engine due to fallback checking.

- **Error Handling:** Preliminary results indicated that watermarks detection system had a poor performance with a highly blended text. Errors in accuracy were reduced to almost 89% through tweaking of threshold values and adding training samples.

- **System Scalability:** The pipeline can be further expanded to include more models (e.g., fine- grained attribute detection) without need to change the architecture and this proves that there is good scaled ability of the system.

Chapter 8

Social, Legal, Ethical, Sustainability and Safety aspects

IT systems have varied impacts to the society. A system of image perfection verifying with AI should thus be considered in terms of social adoption, lawfulness, moral accountability, sustainability roles, and safety measures. Responsible use can only be achieved by collaboration among engineers, platform operators and users since abuse may have professional and operational implications. Ethic factors must be valued particularly when robotics are used in making judgments, which concern trade.

8.1 Social Aspects

The system also plays constructively in society by giving rise to the level of trust and transparency in the e- commerce platforms. Ensure that product images are correct and allowable and it will remind of misleading listings, less customer dissatisfaction and fair competition between sellers. Automated moderation might sometimes pose inconvenience in a situation where correct images are wrongly categorized as such but such matters can be countered by enabling human review. The system mirrors general trends in the wider society regarding AI adoption, such as the necessity of fairness, transparency of decision-making, and minimizing digital deceit.

8.2 Legal Aspects

As the system operates on uploads by users, it is necessary to follow the rules set by the data-protection law, including GDPR and the DPDPA (2023) of India. Such rules assure a clear processing of data, less retention, data security, and user access like corrections and erasure. Copyright is also prevented with the help of the system analyzing the pictures used by reusing or copying them. Clear paperwork, logic behind making needless decisions and mechanisms to appeal against such decisions should be upheld in order to prevent legal wrangles. By making sure the data is handled legally and fairly, there is legal risk mitigation on both parties; developers and marketplace operators.

8.3 Ethical Aspects

Some of the ethical considerations involve fairness on the decisions made by AI, responsibility on errors as well as avoiding bias. The system enhances the well-being of consumers whereby honest product representation and minimal fraudulent acts are lowered. It does not produce an addictive effect, or depersonalizes the use of it but according to the auto-decisions, it may have an impact on the sellers unless the use is well monitored. There is a defined set of ethical norms concerning engineers in professional organizations like IEEE/ACM- considering the common good, keeping it transparent, and constantly evaluating the work of models in order to mitigate the harmful results...

8.4 Sustainability Aspects

Sustainability factors correspond primarily to the computation use, storage demand as well as long- term maintainability. Energy consumption and costs of operation are minimised through lightweight models and streamlined processing. The effects of cloud deployment include waste reduction of physical infrastructure and efficient scaling. The system avoids product returns and logistical waste indirectly due to avoiding wrong product listing, and it leads to even more responsible consumption patterns. The system is long term sustainable as it is robust and modular in design implying that it can be updated without the need to redevelop the entire system.

8.5 Safety Aspects

In this case, the emphasis of safety is put on the data security, system reliability, and the avoidance of misuse. Images uploaded by the sellers are defended with secure API along with encrypted storage and restricted access. Strain testing makes sure that the system will not give detrimental or unpredictable results. Manual review and threshold adjustment are some of the ways of enhancing operational safety on the part of fail- safes. Safe and reliable e-commerce environment is guaranteed by the implications of cybersecurity practices, including encryption, frequent monitoring, and effective authentication, which ensure that the AI system is not attacked or exploited by ill-intentioned individuals and maintains the essential safety levels.

Chapter 9

Conclusion

This project came up with an automatic system of checking the accuracy of the product pictures posted in the e-commerce websites. The solution combines various AI elements that include detection of objects, relevance, OCR-driven text-detection, imaging-quality analysis, and duplicity-image recognition into a single operation. The methodology was designed in such a way that every image uploaded went through various steps of being verified and as such, the system was able to detect things like more than one object, poor backgrounds, watermark, low resolution and wrong category.

The process of implementation delivers on what was mentioned in the Introduction. The objective of behavior of the system was the automated recognition of the main product in an image. The objective of the analysis was realized through the CLIP similarity, OCR extraction, and quality performance to measure the correctness. The objective on system-management was that the decision engine developed can classify the images as Accepted, Rejected or Manual Review is needed. The security goals were achieved by means of restricted access and sound data management. Lastly, the goal of deployment was achieved through the components being incorporated into one of the back-end pipelines befitting real-time application.

The findings indicate that the system can successfully identify wrong, deceiving, or inferior product images accurately, and thus marketplaces can use it to preserve the appropriate listing and increase customer confidence. The workflow will also make sure that the evaluation is homogenized across various categories and reduces the work to be done in manual moderation and assists the sellers to the correct policies of the platform.

Opportunities to improvement still exist. The system may be enhanced by adding more product types to the list, watermark recognition of complex patterns and fine-grained attributes matching including checking color or size. They could contain explainable-AI approaches to give even better feedback to sellers regarding reasons why they reject images. Human-in-the-loop review dashboard can also be applied with the goal of correcting model performance with actual cases.

On the whole, this project shows that image correctness verification can be automated and scalable based on practicality and that it is a good basis of advanced and sophisticated image moderation systems in e-commerce.

Future Recommendation

- **Deploy very pictorial product attribute detection.**

  Enhance the system to detect the minor features like precise color, material or size differences not addressed in the current.

- **Improve a more sophisticated watermark detection unit.**

  Combine the models with possibilities to identify the weak, semi-transparent, or stylized watermarks that cannot be spotted using conventional OCR-based methods.

- **Increasing and diversifying training sets.**

  Bring in more category-specific images that will create better model generalization on a broader sample of product types and photography styles.

- **Introduce explainable AI (XAI) in the transparency of the sellers.**

  Explain picture rejection in a more transparent way so that a seller will be able to understand and address the problem more efficiently.

- **Introduce a human-in-the-loop moderation dashboard**

  Allow manual reviewers to validate borderline cases and provide feedback that can be used to retrain and refine the system.

- **Optimize models for lower computation cost**

  Explore lightweight architectures or model compression techniques for faster and more energy-efficient real-time processing.

References

[1] A. Krizhevsky, I. Sutskever and G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, Advances in Neural Information Processing Systems (NIPS). pp. 1097–1105, 2012.

[2] S. Ren, K. R. Girshick and J. Sun together with him, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 2017.

[3] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement arXiv preprint, arXiv:1804.02767, 2018.

Bochkovskiy, C. Y. Wang and H. Y. M. Liao, YOLOv4: Optimal Speed and Accuracy of Object Detection Preprint on arXiv, arXiv:2004.10934, 2020.

[5] A. Radford et al., Learning Transferable Visual Models with Natural language Supervision, Proceedings of the 38th Annual International Conference on machine learning (ICML), 2021.

[6] B. Shi, X. Bai and C. Yao, “An End-to-End Trainable Neural Network to Recognize Image-Based sequences and apply to Scene Text Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 11, pp. 22982304, 2017.

[7] A. Mittal, A. K. Moorthy and A. C. Bovik, “No-Reference Image Quality Assessment in the Spatial Domain IEEE Signal Processing Letters, vol. 20, no. 5, pp. 409412, 2013.

[8] A. Zauner, Implementation and Benchmarking of Perceptual Image Hash Functions. Upper Austria University of Applied Sciences, 2010.

[9] Z. Zhong, L. Sun and Q. H. Wang, “A Survey on Deep Learning-Based Watermarking Methods and Techniques IEEE Access, vol. 11, pp. 35570–35593, 2023.

[10] F. Faghri, D. J. Fleet, J. R. Kiros and S. Fidler, VSE++: Improving Visual-Semantic Embeddings with Hard Negatives Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 73657373, 2017.

Base Paper

Base Paper Title:

Redmon, J., & Farhadi, A. (2018). YOLOv3: A small enhancement. arXiv preprint arXiv:1804.02767.

Overview of the Base Paper

Among the most powerful contributions to the field of real-time object detection, the chosen base paper, YOLOv3: An Incremental Improvement by Redmon and Farhadi features some of the most significant improvements in the field. YOLOv3 provides architectural changes that enhance detection speed and accuracy that makes it possible to recognize objects of different sizes and backgrounds efficiently. The relevance of this paper to the existing project is that the image of the product posted on the online marketplaces have to be analyzed rather fast to understand whether it represents the required object, shows several items, or irrelevant objects. The architecture of YOLOv3 enables fast processing yet excellent detection, and thus the system can be used in large-scale and real-time markets where thousands of images are uploaded per hour.

Key Features of the Base Paper

The YOLOv3 paper contributes several major technical improvements that directly support the goals of automated product image correctness validation:

1. Real-Time Object Detection

The article puts a strong focus on the high speed detection through one run of a neural network, a feature that is crucial to market places that have to handle a high number of images in real time. YOLOv3 demonstrates:

- Quick inference that can be applied to live pipelines which require uploading.
- Even on less powerful hardware, the performance is stable.
- Bulk image validation operations have high throughput.

2. Multi-Scale Detection

The ability to identify items in varying sizes is essential in product listing, in which images can be changed in zoom levels, surrounding clutter, or framing. YOLOv3 introduces:

- Feature pyramid networks
- Three detection layers of the different sizes of objects This option assists in detecting the products even when the sellers post images in a poor frame or irregular form.

3. Improved Feature Extraction

The bottom paper gives the Darknet-53 backbone, which has made a significant advancement in the quality of representation.

For this project, that means:

- Improved recognition of the key product.
- Minimized false alarms on the background details.
- Greater uniformity in different categories of products.

4. High Generalization Across Domains

YOLOv3 is trained on massive datasets (e.g., COCO), and thus can be fine-tuned to marketplace products categories without a significant amount of training.

This supports:

- Recognition of objects of everyday use.
- Learning of products with little training information.
- Strength in cases where the sellers post images on numerous sources.

Relevance to the “Image Correctness for a Product on the Marketplace” Project

The YOLOv3 paper aligns directly with the objectives of your project:

- As mentioned in the paper, there must be the need in object detection speed and precision, which is what your project achieves to be sure that only the product(s) actually relevant will be displayed in pictures uploaded by the sellers.
- Multi-object detection feature of YOLOv3 will allow your system to detect and label images that have more than one object, messy backgrounds, or objects not relevant to your business.
- Its great precision aids in the validation of product focus enabling the project to crop, isolate.
- Principles of real-time detection that are incorporated in the paper will enable your system to provide instant feedback to sellers prior to posting a product listing.
- The model is very powerful and can be used in the large markets which are dealing with thousands of new items per minute.

In that way, YOLOv3 is taken as the baseline paper, which predetermines the technical viability of the automated verification of image correctness with the help of computer vision.

Reason for Choosing This Paper

The YOLOv3 paper was selected as the base reference for several reasons:

- It offers an object-detection framework of the latest state of the art and proven to be applicable to product-image validation.
- It shows that it is fast, and this is a prerequisite of real-time marketplace systems.
- Major features: It has good performance on multi object, multi scale scenarios which is in line with the marketplace conditions.
- YOLO models are currently used by major companies in the industry (Amazon, Alibaba, and Flipkart) confirming its effectiveness in e-commerce settings.
- The article is referenced, reputable and technical in nature, which renders it viable as a scholarly basis in developing the project.

How Our Project Extends the Base Paper

Aspect	YOLOv3 Base Paper Findings	Image Correctness Project Enhancement
Core Function	Real-time object detection	Detection + relevance + quality + OCR + duplicate check
Scope	Detect objects in general images	Verify correctness of e-commerce product images
Multi-Object Handling	Identifies multiple objects	Flags multi-product violations for compliance
Processing	Fast single-shot detection	Real-time seller feedback pipeline
Limitations	No watermark or text validation	Added OCR/Tesseract for watermark detection

No duplicate detection	Added perceptual hashing + deep similarity	
No relevance Scoring	Added CLIP for image–text matching	

Conclusion

The technical basis of the system under Image Correctness on a Product on the Marketplace is the YOLOv3 paper. Although the underlying paper determines the possibility of the efficient implementation of the real-time detection of the object, your project takes the next step by converting the raw detection potential into a validation pipeline. The system has an added layer of checks such as text extraction, quality analysis, semantic relevance scoring, and duplicate identification functionalities, which are not in the base paper. Your project will be useful and able to scale to large numbers with project extensions of YOLOv3 into a full correctness verification tool to enhance confidence in the products listed to reduce returns and ensure the integrity of products across the conditions of businesses in the contemporary e-commerce marketplaces.

APPENDIX

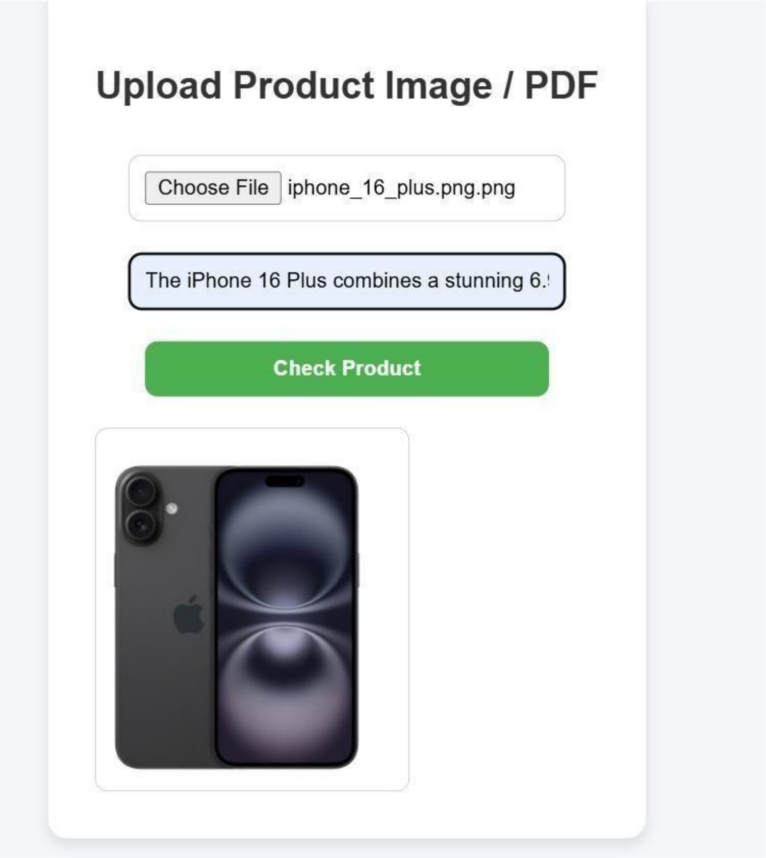
The appendix contains the support materials, data sets, visuals and technical details associated with the design and testing of the AI-based Image Correctness Verification System. Such materials will shed more light on the implementation, testing and validation of the project.

1. Data Sheets

Though it is mainly a software project, it relies on a number of external libraries and frameworks as well as ready-made models. As datasheet summaries, the following are contained:

1. **YOLO (You Only Look Once) Model Family**
   - Consists of model architecture specification, the input dimensions, feature extracting backbone, and detection layers.
   - The major specifications: inference speed, the amount of layers, FLOPs, and training parameters.
2. **CLIP (Contrastive Language–Image Pretraining)**
   - Incorporation of prompt dimensions, architecture of Transformers, tokenization procedure, and thresholds of cosine similarity.
3. **OCR Module (Tesseract / CRNN)**
   - prevailing languages, lowest threshold of images: resolution, accuracy wish, and text- extraction chain.
4. **Perceptual Hashing (pHash)**
   - Length of hash, similarity measure, bit by bit comparison and tolerance levels to detect duplicate.

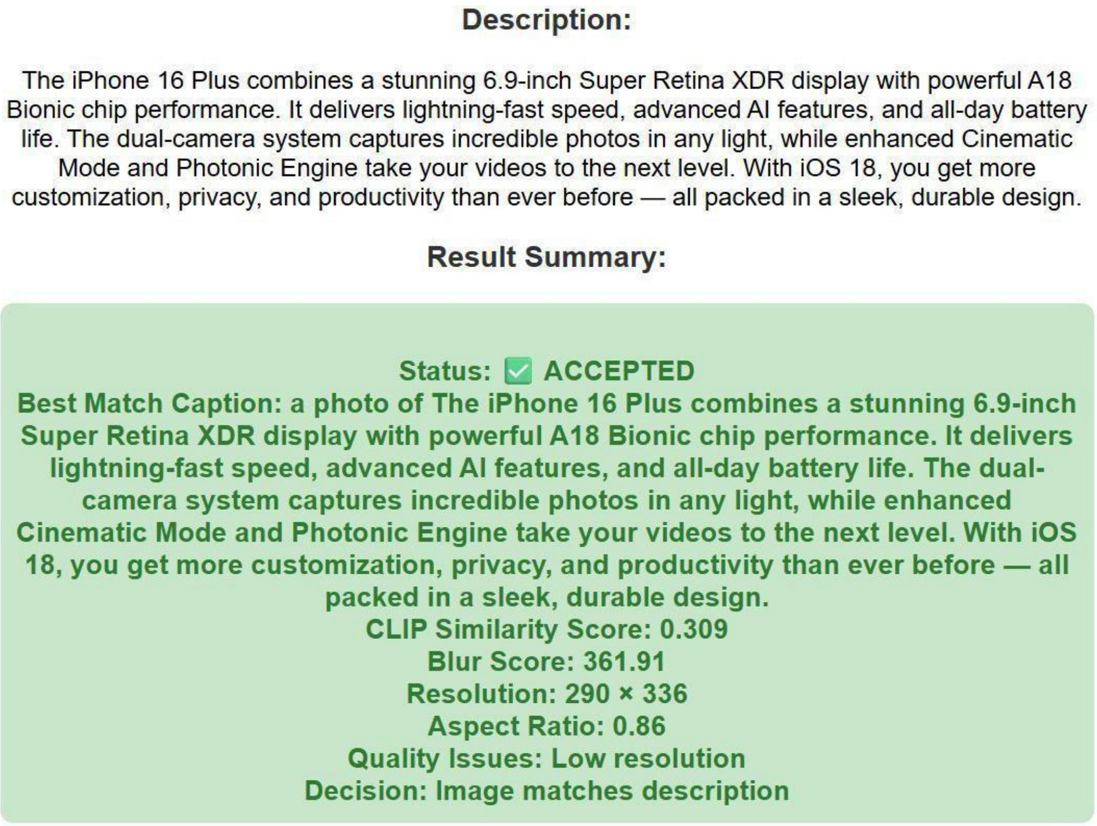
1. Project images:



Upload Product Image / PDF

iphone_16_plus.png.png

The iPhone 16 Plus combines a stunning 6.!

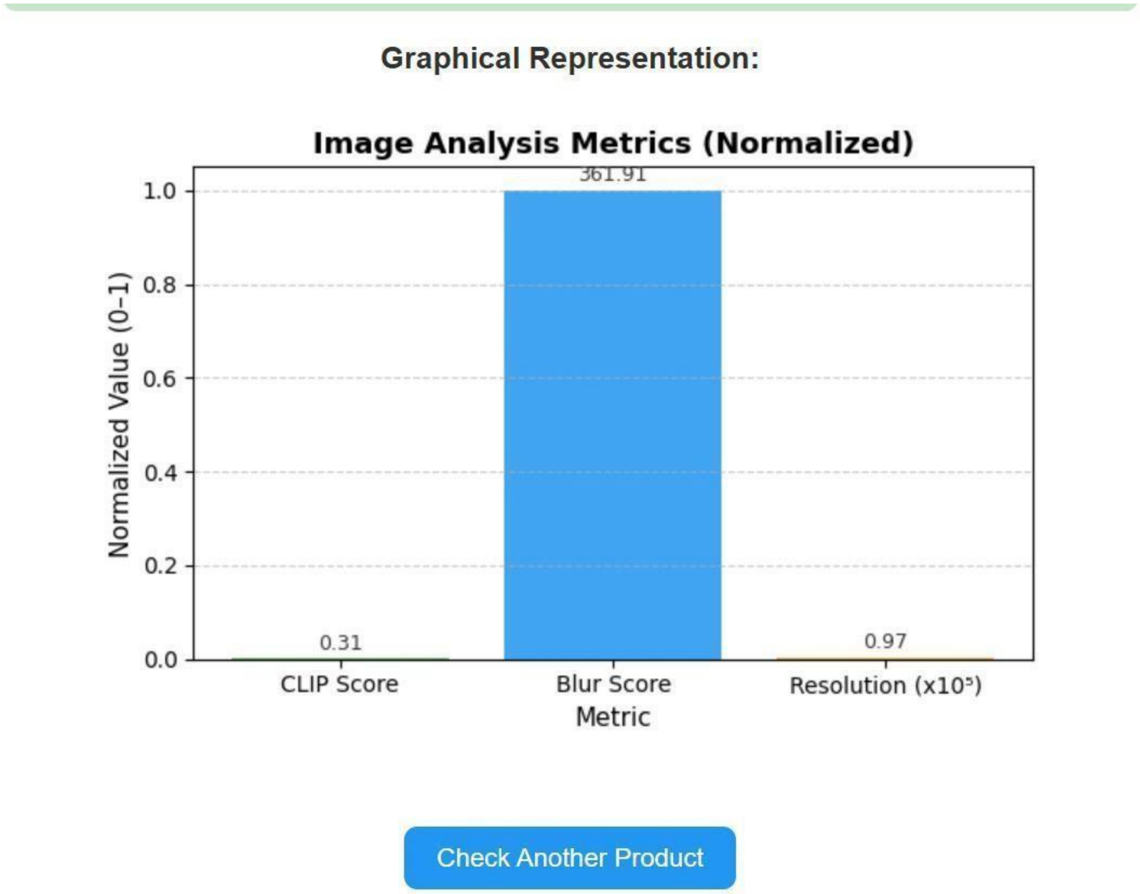


Description:

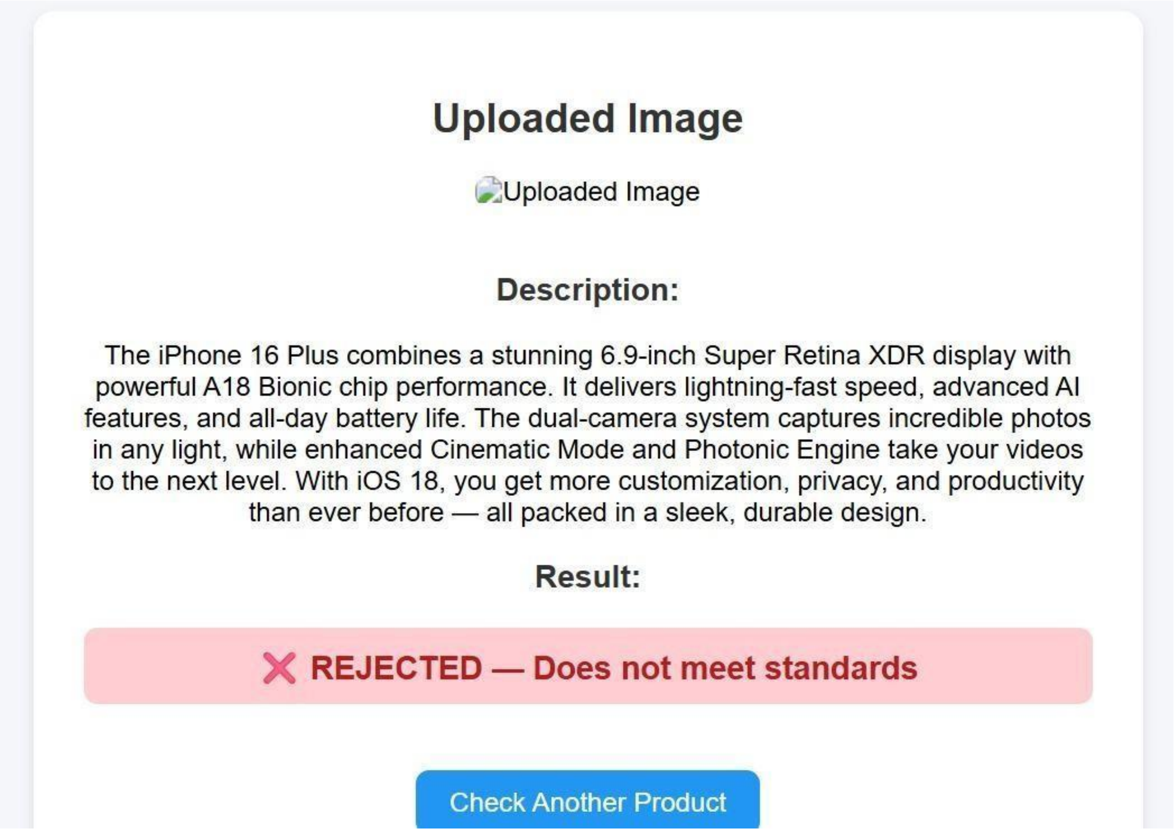
The iPhone 16 Plus combines a stunning 6.9-inch Super Retina XDR display with powerful A18 Bionic chip performance. It delivers lightning-fast speed, advanced AI features, and all-day battery life. The dual-camera system captures incredible photos in any light, while enhanced Cinematic Mode and Photonic Engine take your videos to the next level. With iOS 18, you get more customization, privacy, and productivity than ever before — all packed in a sleek, durable design.

Result Summary:

Status: ✅ ACCEPTED
Best Match Caption: a photo of The iPhone 16 Plus combines a stunning 6.9-inch Super Retina XDR display with powerful A18 Bionic chip performance. It delivers lightning-fast speed, advanced AI features, and all-day battery life. The dual-camera system captures incredible photos in any light, while enhanced Cinematic Mode and Photonic Engine take your videos to the next level. With iOS 18, you get more customization, privacy, and productivity than ever before — all packed in a sleek, durable design.
CLIP Similarity Score: 0.309
Blur Score: 361.91
Resolution: 290 × 336
Aspect Ratio: 0.86
Quality Issues: Low resolution
Decision: Image matches description



Graphical Representation:



Uploaded Image

Uploaded Image

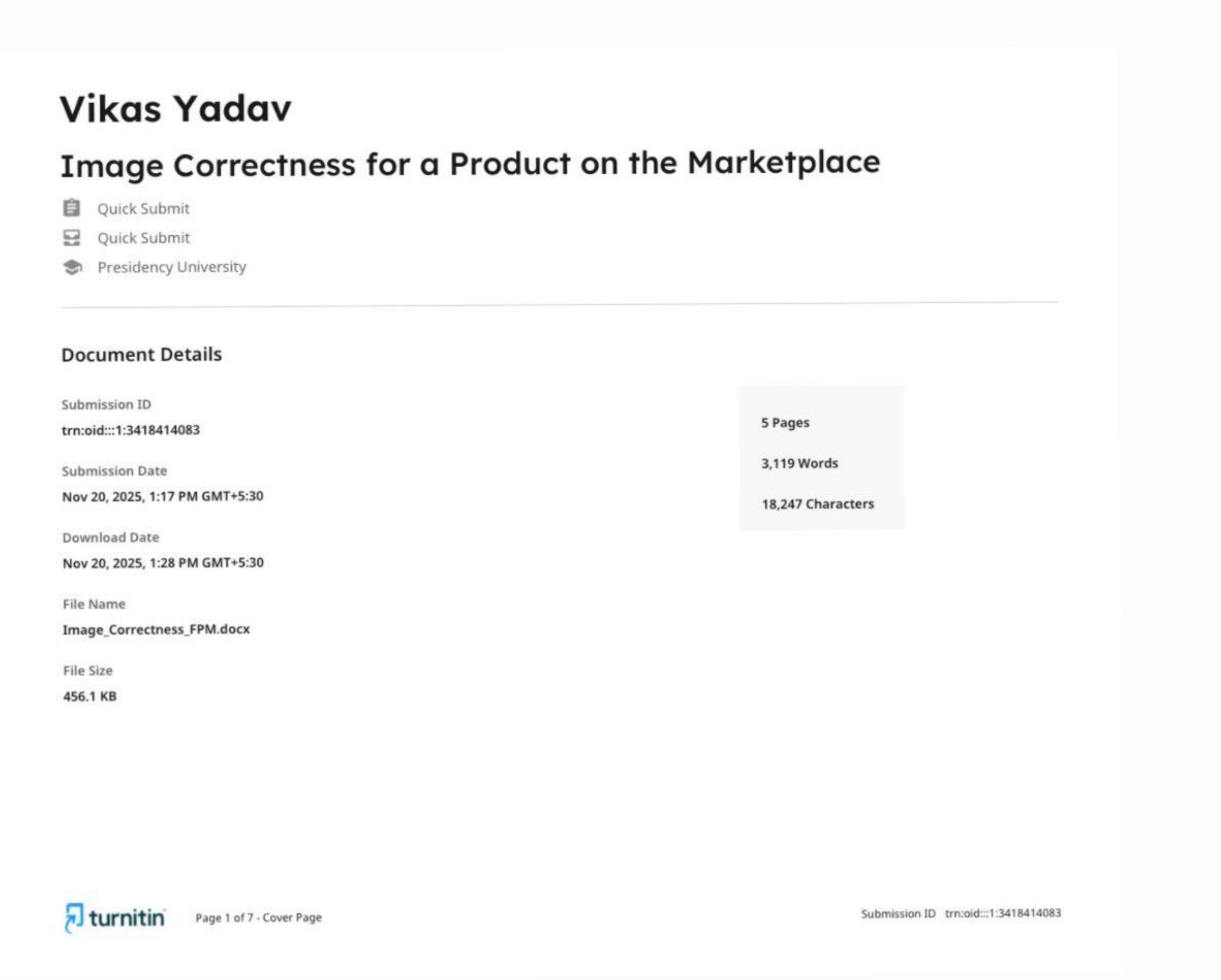
Description:

The iPhone 16 Plus combines a stunning 6.9-inch Super Retina XDR display with powerful A18 Bionic chip performance. It delivers lightning-fast speed, advanced AI features, and all-day battery life. The dual-camera system captures incredible photos in any light, while enhanced Cinematic Mode and Photonic Engine take your videos to the next level. With iOS 18, you get more customization, privacy, and productivity than ever before — all packed in a sleek, durable design.

Result:

❌ REJECTED — Does not meet standards

1. Project Report:



Vikas Yadav

Image Correctness for a Product on the Marketplace

Quick Submit

Quick Submit

Presidency University

Document Details

Submission ID

trn:oid:::1:3418414083

Submission Date

Nov 20, 2025, 1:17 PM GMT+5:30

Download Date

Nov 20, 2025, 1:28 PM GMT+5:30

File Name

Image_Correctness_FPM.docx

File Size

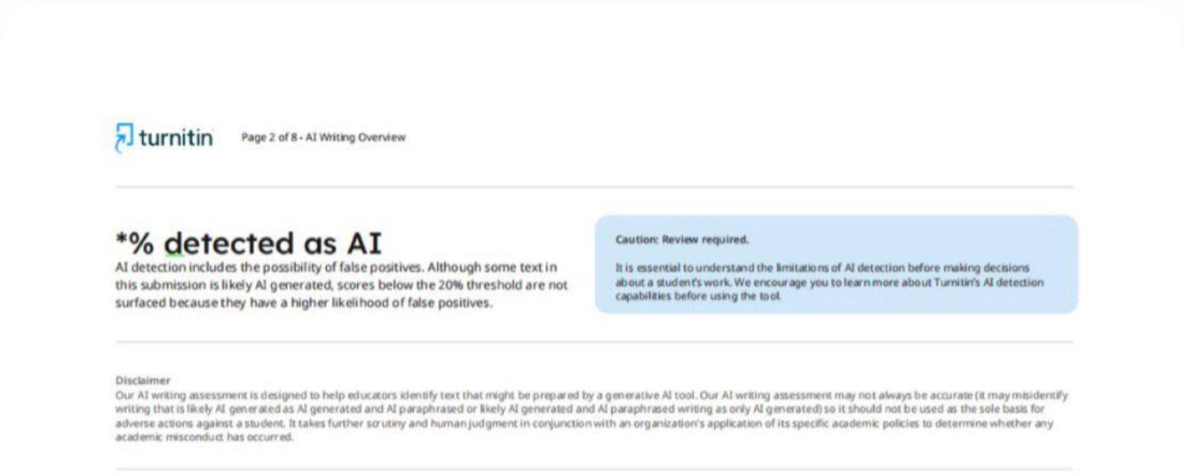
456.1 KB

5 Pages

3,119 Words

18,247 Characters

turnitin Page 1 of 7 - Cover Page Submission ID trn:oid:::1:3418414083



turnitin Page 2 of 8 - AI Writing Overview

*% detected as AI

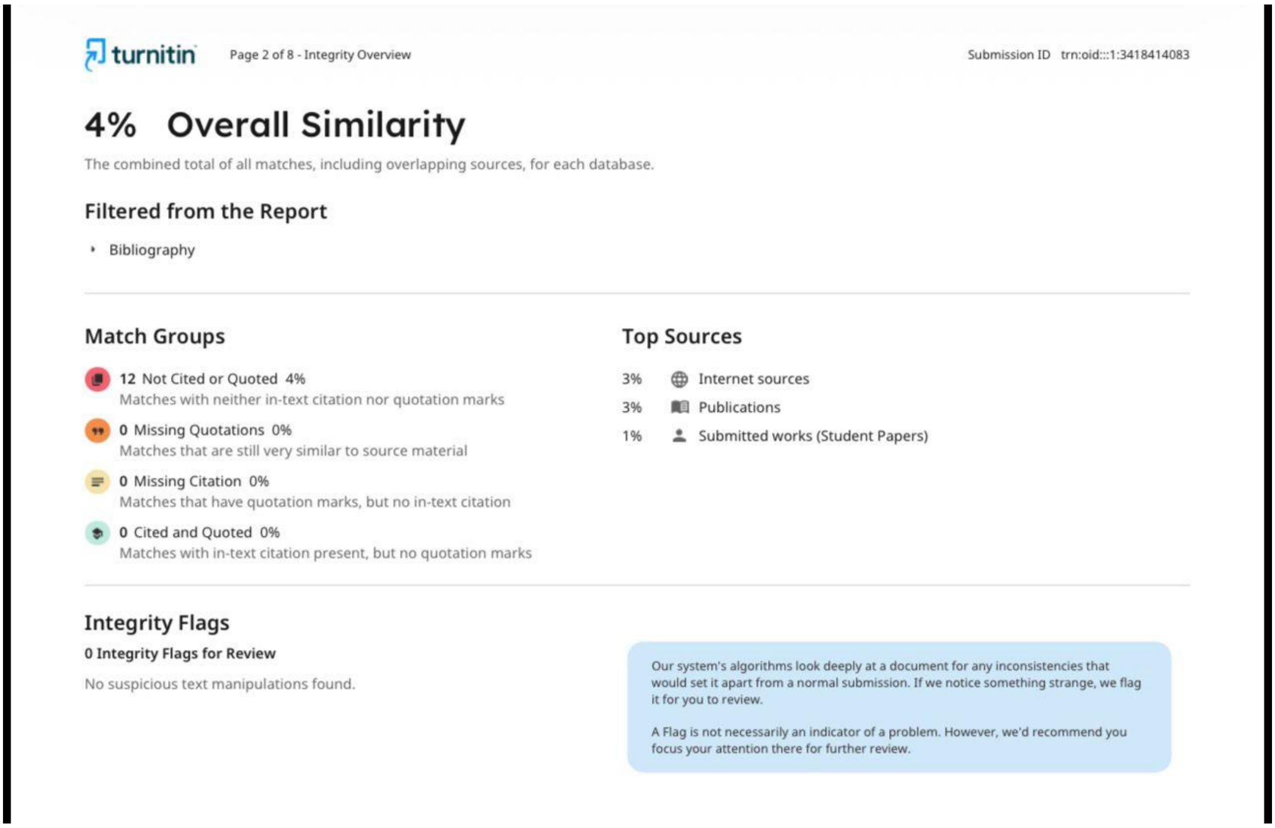
AI detection includes the possibility of false positives. Although some text in this submission is likely AI generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify writing that is likely AI generated as AI generated and AI paraphrased or likely AI generated and AI paraphrased writing as only AI generated) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.



Page 2 of 8 - Integrity Overview

Submission ID trn:oid:::1:3418414083

4% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Match Groups

12 Not Cited or Quoted 4%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

3%	Internet sources
3%	Publications
1%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Figure A.1.1 Project AI Detection Report.