# Image Detection Model for Cloud Clean-up and Cost Optimization.

DISSERTATION

Submitted in partial fulfillment of the requirements of the

Degree : MTech in AI and ML with deep learning

By

Gaurav Shukla

2022ac05280@wilp.bits-pilani.ac.in

Under the supervision of

Thejus Yerigeri Director Client Technology EY GDS

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE

Pilani (Rajasthan) INDIA

(April, 2025)

**BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI**

**FIRST SEMESTER 2024-25**

DSECLZG628T **/ AIMLCZG628T DISSERTATION**

Dissertation Title : Image Detection Model for Cloud Cleanup and Cost Optimization.

Name of Supervisor : Thejus Yerigeri

Name of Student : Gaurav Shukla

ID No. of Student : 2022ac05280

Courses Relevant for the Project & Corresponding Semester:

1. ML Ops
2. Deep learning
3. Advanced Deep learning
4. Deep Neural Network
5. Python fundamentals for data science
6. Artificial and computational intelligence

## Abstract

##### 

Cloud storage is essential for individuals and businesses, but unorganized and redundant data increases costs and inefficiency. This project introduces a machine learning-based Image Detection Model to identify and categorize unused, duplicate, and low-value images (e.g., screenshots, blurred photos, and redundant files). Using advanced image recognition, the system classifies and ranks images based on relevance, quality, and duplication, offering actionable suggestions to optimize storage, reduce costs, and streamline digital asset management.

### ****Image Detection System Workflow****

The system is designed to perform image detection tasks such as blur detection, screenshot identification, and duplicate image detection. It integrates multiple components to facilitate user interaction, model processing, and feedback collection for continuous improvement.

### ****User Interaction and UI****

The process begins with the **User**, who interacts with the **User Interface (UI)**. The UI serves as the front end for users to review data and verify results provided by the model. This will help to train the model.

A diagram of a software system

Description automatically generated

### ****Data Source: Google Drive or MS One drive or Local Folder****

Uploaded images are sourced from **Google Drive**, One Drive or a file folder serving as the primary repository. These images are transferred to the **Image Detection Service** for analysis.

### ****Image Detection Service****

The Image Detection Service is the core module responsible for handling image processing. It takes input data and performs various detection tasks. Specifically:

* Blur Detection identifies if an image is unclear or blurry.
* Screenshot Detection determines if an image is a screenshot.
* Duplicate Image Detection checks for duplicate images.

The service routes data to these detection modules as needed. Results from these modules are relayed back for further processing or user visibility.

### ****Model Prediction and Serving****

The Serving Model plays a crucial role in providing model predictions for the detection tasks. Fine-tuned parameters improve model accuracy. Processed images and prediction results are sent back to the Image Detection Service, ensuring the system operates with optimized outputs.

### ****Datastore and Feedback Loop****

Detection results and metadata are stored in the **Datastore**. This module collects user feedback (step 6), ensuring users can validate or flag incorrect results. The datastore facilitates two-way communication between the UI and system, improving transparency and user satisfaction.

#### ****Model Training****

Collected user feedback triggers the Training Model. The feedback is utilized to retrain and fine-tune the detection models, ensuring they evolve with time. Fine-tuned parameters are passed back to the Serving Model, creating a continuous learning loop.

#### ****Workflow Summary****

* User verifies the results outcome via the UI.
* Image is sourced from Google Drive to the Image Detection Service.
* The detection modules (Blur, Screenshot, Duplicate) process the images.
* The Serving Model predicts outcomes.
* Results are displayed to the user, and feedback is collected.
* Feedback is sent to the Training Model to improve future predictions.
* Fine-tuned parameters update the Serving Model.

This iterative workflow ensures consistent improvements in detection accuracy. The system architecture demonstrates a scalable and efficient image processing pipeline. It integrates user interaction, modular detection tasks, and a robust feedback loop to deliver reliable results while enhancing model performance over time.

### **Project Work Plan**

|  |
| --- |
| Detailed Plan of Work |
| |  |  |  |  |  | | --- | --- | --- | --- | --- | | Serial No. | Task/Phase | Start Date - End Date | Duration (Weeks) | Deliverable | | 1 | Literature Review | Week 1 - Week 2 | 2 | Problem definition and references | | 2 | Data Collection & Annotation | Week 3 - Week 5 | 3 | Labeled dataset | | 3 | Model Training & Fine-Tuning | Week 6 - Week 8 | 3 | Optimized image recognition model | | 4 | Deployment on Cloud Platform | Week 9 - Week 11 | 3 | Deployed application | | 5 | User Testing & Feedback | Week 12 - Week 13 | 2 | User feedback report | | 6 | Final Model Iterations | Week 14 - Week 15 | 2 | Improved system | | 7 | Documentation & Presentation | Week 16 | 1 | Final dissertation | |

#### ****Literature Review:****

* Review existing techniques in image detection, cloud storage organization, and cost-reduction strategies.
* Explore state-of-the-art models like ResNet, YOLO, or CLIP for image analysis.

#### ****Model Development:****

* Data Preparation: Collect and annotate a dataset of images categorized as high-importance, duplicates, and low-quality.
* Model Training: Train the image recognition model using transfer learning techniques.
* Fine-Tuning: Optimize the model for precision and recall based on the categories of interest.

#### ****Deployment: exam report****

* Develop a cloud-based application using Azure ML Studio for scalability.
* Utilize Kubernetes for containerized deployment, ensuring high availability.

#### ****Integration and Testing:****

* Integrate the model with cloud drive APIs (e.g., Google Drive, Dropbox).
* Conduct user testing to gather feedback and improve the system.

#### ****Learning from Usage:****

* Implement feedback loops for continuous model improvement.
* Use unsupervised learning techniques to adapt the model to new patterns of image storage and user behaviour.

Here’s an outline of the benefits and unique features of this proposition compared to Google’s duplicate detection and other similar features:

#### ****Comprehensive Image Classification****

* Google's Approach: Google Photos mainly identifies exact duplicates or near-identical images. It lacks a deeper analysis of image value or relevance.
* Our Approach: Beyond detecting duplicates, this proposition identifies low-value images such as screenshots, blurred photos, and redundant files. It classifies images based on quality, relevance, and duplication—offering a more refined and intelligent solution.

#### ****Advanced Ranking Mechanism****

* Google's Approach: Duplicate detection operates without ranking images based on their significance. Users manually decide which files to delete.
* Our Approach: The system ranks images based on relevance and quality, helping users prioritize and retain the most valuable content while cleaning up unnecessary clutter efficiently.

#### ****Actionable User Suggestions****

* Google's Approach: While Google Photos groups duplicates for deletion, it does not provide tailored recommendations to optimize storage.
* Our Approach: The system provides actionable and user-friendly suggestions, such as “Delete blurred images,” “Remove unnecessary screenshots,” or “Archive duplicates,” streamlining the cleanup process.

#### ****Context-Aware Detection****

* Google's Approach: Exact duplicate detection focuses on file-level analysis but lacks contextual understanding (e.g., distinguishing valuable photos from low-value ones).
* Our Approach: Using advanced image recognition techniques, the system distinguishes between screenshots, redundant content, and meaningful photos. It identifies blurry images or repetitive files, making it context-aware and intelligent.

#### ****Tailored to Cloud Storage Cleanup****

* Google's Approach: Google's features often focus solely on specific platforms (like Google Photos) and offer general storage management tools.
* Our Approach: The system is explicitly tailored for efficient cloud storage cleanup, ensuring users retain only essential files while optimizing space usage.

#### ****Key Words:**** Here are some relevant keywords:

* Cloud Storage Optimization
* Image Detection Model
* Machine Learning-Based Cleanup
* Duplicate Image Detection
* Low-Value Image Identification
* Advanced Image Recognition
* Blurred Photo Detection
* Screenshot Identification
* Redundant Data Removal
* Digital Asset Management
* Cloud Storage Efficiency
* Storage Cost Reduction
* Relevance-Based Image Ranking
* Actionable Cleanup Suggestions
* Unorganized Data Management
* Context-Aware Detection
* Automated Image Categorization
* User-Friendly Recommendations
* Optimized Cloud Space
* AI-Powered Storage Cleanup

**BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI**

**I SEMESTER 24-25**

DSECLZG628T **/ AIMLCZG628T DISSERTATION**

**Dissertation Outline (Abstract)**

**BITS ID No.** 2022ac05280 **Name of Student:** Gaurav Shukla

**Name of Supervisor:** Thejus Yerigeri

**Designation of Supervisor**: Director GDS EY

**Qualification and Experience:** BE, MBA(IIM-Bangalore)

**Official E- mail ID of Supervisor:** Thejus.Yerigeri@gds.ey.com

**Topic of Dissertation**: Image Detection Model for Cloud Cleanup and Cost Optimization.

(Signature of Student) (Signature of Supervisor)

Date:--8th December 2024--- Date:--8th December 2024---

**Discussion on the Chosen Topic**

1. **Purpose of the Work and Expected Outcome**  
   The primary goal of this work is to develop an image detection machine learning (ML) model to optimize cloud storage by identifying and categorizing images. The expected outcome includes a fully functional system that reduces redundant or unnecessary image storage, thereby minimizing cloud costs for individual users.
2. **Literature Review**  
   A comprehensive review of state-of-the-art image detection techniques was conducted, focusing on methodologies for image classification, duplicate detection, and quality assessment. Studies also highlighted the increasing costs associated with unmanaged cloud storage and the need for automated solutions.
3. **Existing Process and Limitations**  
   Currently, cloud storage users rely on manual deletion or basic inbuilt tools that are not tailored for identifying redundant or low-value images. These approaches are time-consuming, lack accuracy, and do not provide actionable recommendations for cost reduction.
4. **Justification for Methodology**  
   The chosen methodology leverages transfer learning with pre-trained models like ResNet and YOLO, ensuring a balance between development time and accuracy. This approach allows the reuse of robust feature extraction capabilities, making it suitable for large datasets and quick deployment.
5. **Project Work Methodology**  
   The work involves six phases:
   * Data collection and pre-processing to create a labelled dataset.
   * Model development using transfer learning techniques.
   * Model training and optimization for high precision and recall.
   * Deployment using cloud platforms like Azure or Google Cloud, integrating APIs for real-time processing.
   * User testing to gather feedback and refine the system.
   * Final iteration and documentation for delivery.
6. **Benefits Derivable from the Work**  
   This solution provides significant cost savings by automating image categorization and deletion, thereby reducing unnecessary cloud storage usage. It also enhances user experience by offering intelligent recommendations and improving storage organization.
7. **Other Supporting Details**  
   The project is scalable and can be integrated with multiple cloud platforms, making it applicable to a broader user base. Additionally, the system’s modular design ensures easy updates and adaptability to future requirements.

# Detailed Plan of Work

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Serial No. | Task/Phase | Start Date - End Date | Duration (Weeks) | Specific Deliverable |
| 1 | Literature Review | Week 1 - Week 2 | 2 | Problem definition and review of state-of-the-art image detection techniques |
| 2 | Data Collection & Annotation | Week 3 - Week 5 | 3 | Labelled dataset of images with categories: high-importance, duplicates, and low-quality |
| 3 | Model Development - Initial | Week 6 | 1 | Preliminary image detection model (baseline performance) |
| 4 | Model Training & Optimization | Week 7 - Week 8 | 2 | Optimized and fine-tuned model with improved precision and recall |
| 5 | Deployment Preparation | Week 9 | 1 | Cloud application setup and API integration framework |
| 6 | Deployment on Cloud Platform | Week 10 - Week 11 | 2 | Fully deployed and functioning application with UI integration |
| 7 | User Testing & Feedback | Week 12 - Week 13 | 2 | User feedback report highlighting usability and performance |
| 8 | Final Iterations & Refinement | Week 14 - Week 15 | 2 | Improved and finalized system addressing feedback and testing outcomes |
| 9 | Documentation & Presentation | Week 16 | 1 | Complete project report and presentation slides |

**Supervisor’s Rating of the Technical Quality of this Dissertation Outline**

EXCELLENT / GOOD / FAIR/ POOR (Please specify): \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Supervisor’s suggestions and remarks about the outline (if applicable).**

Date\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ (Signature of Supervisor)

Name of the supervisor: Thejus Yerigeri

Email Id of Supervisor: Thejus.Yerigeri@gds.ey.com

Mob # of supervisor: