

CampusFind: A Campus-based AI and Cloud-based Lost and Found Objects Management System

Dr. Vaishnavi Ganesh¹, Pratibha Waghale², Aman Sayyed³, Ruchika Wadhbudhe⁴, Achal Patil⁵ and Aditya Meshram⁶

¹Dept. of Computer Science and Engineering, Priyadarshini College of Engineering, Nagpur
Email: vaishnavi.ganesh8@gmail.com

²Dept. of Computer Science and Engineering, Yeshwantrao Chavan College of Engineering (YCCE), Nagpur
Email: pratibha201986@gmail.com

³⁻⁶Dept. of Computer Science and Engineering, Priyadarshini College of Engineering, Nagpur
Email: amansayyed156@gmail.com, wadbudheruchika04@gmail.com, achal.k.patil@gmail.com, adityameshram68687@gmail.com

Abstract— CampusFind is a smart campus, lost-and-found items retrieving and depositing, platform designed to help students securely report, search, and recover their belongings. Built using React, Vite, and TailwindCSS for a fast and responsive interface, and powered by Supabase for authentication, database, and storage, the system ensures secure user verification through college IDs. It features AI (Gemini) integration to automatically detect item details from images and match lost and found items using both image and text similarity. Users receive private notifications for potential matches and can manage their data and profile through a secure My Account portal..

Index Terms—Lost and found items, Supabase, AI(Gemini) integration, image and text similarity, potential matches, smart campus.

I. INTRODUCTION

People on campus can report lost and found things using Campus find, a clever matching based tool. Many faculty members and students on campus have lost their items and are turning to more conventional means, such as WhatsApp groups, to retrieve them. However, due to many WhatsApp groups from various campus branches, people frequently aren't able to retrieve their possessions. The goal of the Campus find project is to do away with the need for conventional procedures and make the process simple and effective.

II. LITERATURE SURVEY

Focus Track: Enhancing object detection and tracking for small and ambiguous objects[7]

Tracking small, hazy, and concealed objects in busy settings is a specialty of Focus Track, a powerful new one-stage Multi-Object Tracking (MOT) system. A refined YOLOv10 detector is used. The Adaptive Similarity Enhancement (ASE) module, which considers occlusion severity and motion changes to prevent identity switches, and the Modified Soft Buffered IoU (MS-BIOU), which improves bounding box association, are two of its primary inventions.

Additionally, the system includes a Size- 2 Occlusion Trajectory Smoother (SOTS) and Spatial-Temporal Confidence Enhancement (STCE).[7]

A. Find Mine: Find the Lost Items via Mobile App[13]

In order to assist users in promptly and remotely reclaiming or reporting lost and found things, this paper suggests "FindMine," an intuitive mobile application. Users can share information about goods they've found, such as the location and pertinent keywords, as part of the app's community-focused process. By choosing a region on a map, users who have misplaced an item can look for it. Additionally, it enables chat communication between the two parties to schedule delivery and verify ownership. PostgreSQL is used for the database, Spring Boot is used for the APIs, and Kotlin is used for the backend of the system. Utilizing geographic location for effective searching and posting is the primary objective. This eliminates the need for conventional, centralized lost and found services and facilitates the recovery process.[13]

A distributed image-retrieval method in multi-camera system of smart city based on cloud computing[14]

A distributed image retrieval technique for multi-camera systems in smart cities is presented in this research. It integrates a Distributed Fault-Tolerant Processing (DFP) method and makes use of a cloud computing platform. The three primary components of the approach, which focuses on ContentBased Image Retrieval (CBIR), are creating an inverted index, clustering features using the K-means algorithm through MapReduce, and producing feature vectors using the SURF algorithm. Applying DFP and fault tolerance techniques, including redundancy tactics like erasure, regenerative, and replication codes, is the primary contribution.[14]

III. METHODOLOGY

Basically, Campusfind works using the matching algorithm. Matching process consists of two techniques which is imageimage matching and text-text matching. In some cases AI automatically extracts the brand, color and other information from the image and then it continue with text-text matching. If the information is matching more than 70% we allow both users to securely chat with each other via sending them in-app notifications.

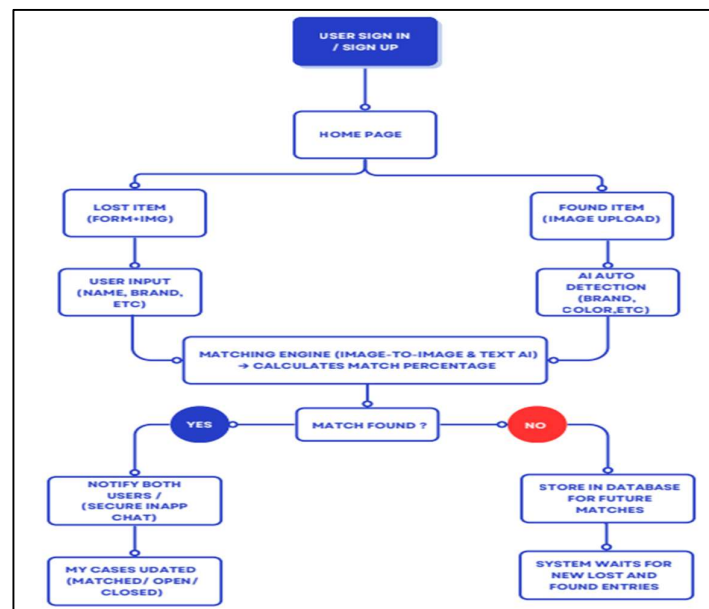


Figure 1 Flowchart showing the working of the project

A. Modules of the project

The project has two portals: one for lost items and another for found items. We gather information about the lost item, including its location, brand, color, description, and optional image, on the lost item portal. Additionally, in the found products portal, we enter the required image, description, brand, color, location, and other data, such as color and brand, which are automatically detected by AI.

B. User Authentication and Verification

Through the site, users can sign up and log in. When creating an account, users must upload their student ID card for verification in order to prevent outside scams. From the admin panel, verification will be carried out.

C. AI Integration(Gemini AI)

- Google's Gemini AI: This tool uses submitted photos to automatically identify and fill in data like item category, color, and brand.
- Allows users to rapidly identify possible matches by supporting image-to-image and text-to-image matching with a similarity percentage.

D. Notification and Case Management

Users who are matched with an accuracy of more than 75% receive the in-app notification. They have two choices in the notification: either claim the item if it is theirs, or reject it if it is not. Additionally, the case will be closed and the user will be able to talk securely if they claim the item. The user's cases will be managed under the "My Cases" tab with three corresponding statuses: open, closed, and matched. Each of the three will display the case's current state.

E. Security

The application offers secure conversation. Supabase, a cloud-based platform, will house all of the ID cards and other user information.

F. Tech Stack:

Backend & Database

Thus, SUPABASE, a cloud-based database management system, is employed. Additionally, Supabase operates in a backend that is managed with PostgreSQL. It facilitates user access and authorization and also permits cloud file storage.

G. Frontend:

- React (Vite) – For a quick, responsive, and modular user interface.
- TailwindCSS – For a simple, adaptable, and mobile-friendly user interface.
- JavaScript (ES6+) –To manage user interaction and dynamic functionality.

H. Deployment

- Netlify – For seamless frontend hosting and deployment.
- Environment Variables (.env) – Used to protect data and save API

IV. RESULT AND DISCUSSION

The first window of the project shows two modules, first tab showing 'report a lost item' and second tab showing 'found an item'.

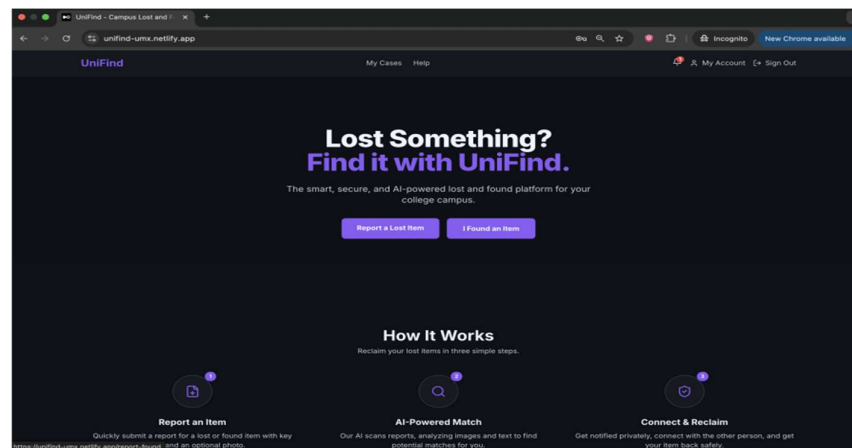


Figure 2. The two modules of the project

For a user who is reporting his lost item, he has to enter the following details about the lost item like its category, the item name, brand, color, last seen location and some additional hints , if the item has any, as shown in figure 3.

The screenshot shows a web browser window with the UniFind application. The page title is "Report a Lost Item". Below the title, there is a sub-header "Provide as much detail as possible to help us find your item." The form contains several input fields: "Category" (a dropdown menu), "Item Name" (a text field with the example "e.g., iPhone 14 Pro"), "Brand" (a text field with the example "e.g., Apple"), "Color" (a text field with the example "e.g., Space Black"), "Last Seen Location" (a text field with the example "e.g., Library, 2nd Floor"), and "Additional Hints" (a text area with the example "e.g., Has a small scratch on the corner, sticker on the back..."). There is also an "Upload an Image (Optional)" section with a note "A picture can significantly increase the chances of finding your item." and a button "Click to upload an image (PNG or JPG under 5MB)". At the bottom right of the form is a "Report Found" button.

Figure 3 Report a Lost Item Tab

In the tab ‘Report a found item’ , a user who found the item will upload its picture, add the found location and some additional hints about the product, as shown in figure 4.

The screenshot shows a web browser window with the UniFind application. The page title is "Report a Found Item". Below the title, there is a sub-header "Thank you for helping our community. Please provide details about the item you found." The form contains several input fields: "Upload an Image" (a text field with a note "An image is required. We'll use AI to analyze it and suggest details." and a button "Click to upload an image (PNG or JPG under 5MB)"), "Found Location" (a text field with the example "e.g., Library, 2nd Floor"), and "Additional Details" (a text area with the example "Any other relevant details, like where exactly you left it for safekeeping."). At the bottom right of the form is a "Report Found" button.

Figure 4 Report a Found Item tab

Whatever items are entered by different registered users who found lost items, they get stored in the tab ‘My cases’ as shown in figure 5.

The screenshot shows a web browser window with the UniFind application. The page title is "My Cases". Below the title, there is a sub-header "My Cases". The page displays a grid of three items: "Laptop Charger" (with a status of "Lost" and a date of "28/10/2023"), "iPhone 14 pro" (with a status of "Found" and a date of "18/10/2023"), and "Keys" (with a status of "Found" and a date of "18/10/2023"). Each item has a small image and a "Report" button. At the bottom of the page, there is a footer with the UniFind logo, a "Product" section (Report Lost, Report Found, Features), a "Company" section (About Us, Careers, Contact), and a "Support" section (Help Center, Terms of Service, Privacy Policy). The footer also includes the copyright notice "© 2025 UniFind. All rights reserved."

Figure 5 My Cases tab showing all found items

The AI integration part of the project automatically extracts the features of the product from the image posted in figure 4 and does image to text conversion and then finds the text to text match percentage between the information posted by the user who lost his item and the information posted by the user who found the item. The potential match is shown in figure 6.

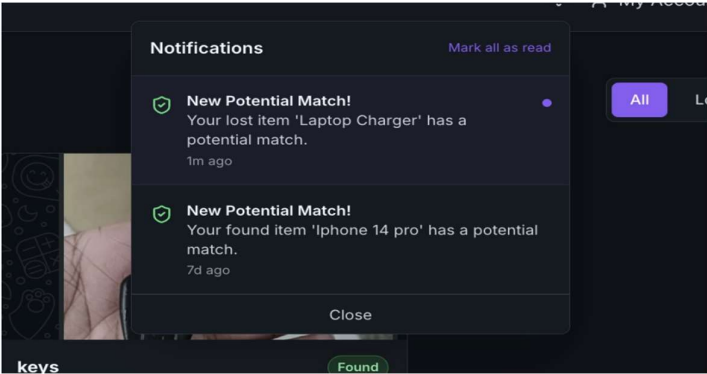


Figure 6 Showing Potential Match

If the potential match is above 70%, which is the threshold decided in this project,as shown in figure 7 then the person who lost his item is allowed to have a secured chat with the person who found the item, as shown in figure 8.

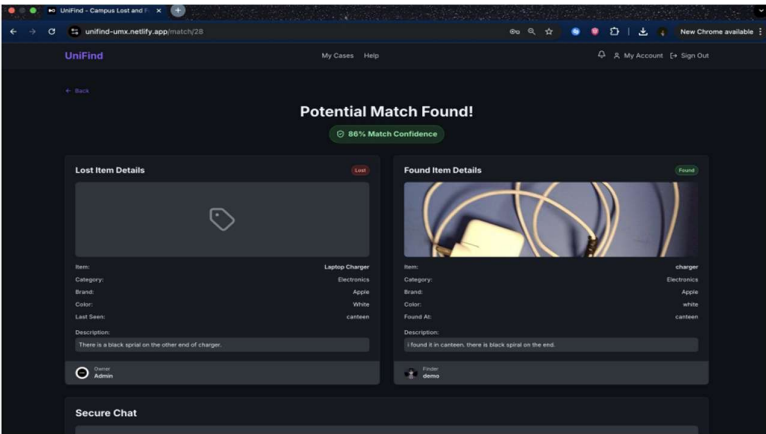


Figure 7 Potential match found to be more than 70%

The following bar graph shown in figure 9 shows that the proposed project outperforms all the existing research work with a best accuracy of 92%, best efficiency of 90% and best detection rate of 94%.

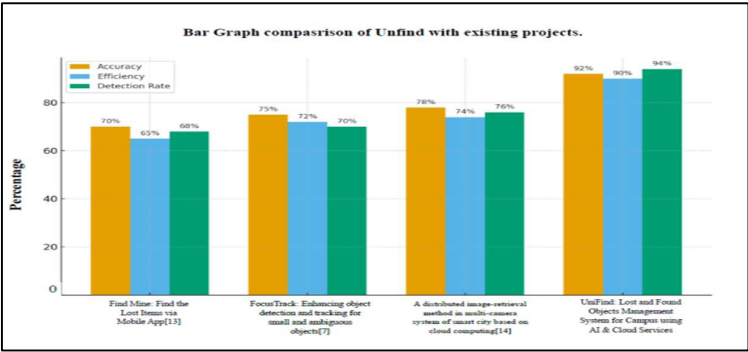


Figure 9. Bar graph showing comparison of Campusfind with existing projects

V. CONCLUSION AND FUTURE SCOPE

The project has two modules. In the first module, a person who lost his item can post its details and in the second module a registered user who found an item can post its details. Using an AI integrated approach and Supabase as a cloud integrated database, a potential match is found between the information posted by the above two parties. If the potential match is above a decided threshold of 70%, then immediately both the above parties are allowed for a secured chat, so that the person who lost his item can get it. The proposed research work outperforms all the existing work with a better accuracy of 92%, best efficiency of 90% and best detection rate of 94%.

The future scope lies in:

- Building a dedicated mobile app for students and college staff to easily report and search lost items.
- Implementation of qr codes of Campusfind in various location in campus like canteen, library, classrooms, etc., so that students can access lost or found portal in one scan and there is no need of sharing links and typing the website.
- Users get alerts when similar items appear through email/SMS /App notifications.

REFERENCES

- [1] S. Abba, A. M. Bizi, J.-A. Leeb, S. Bakouri, and M. L. Crespo, "Real-time object detection, tracking, and monitoring framework for security surveillance systems," *Heliyon*, vol. 10, Apr. 2024, Art. no. e34922, doi: 10.1016/j.heliyon.2024.e34922.
- [2] C.-W. Chou and Y.-T. Hsu, "Robust real-time object detection and counting system for casting foundries," *Appl. Soft Comput. J.*, vol. 176, Apr. 2025, Art. no. 113155, doi: 10.1016/j.asoc.2025.113155.
- [3] S. Jin, Z. Cao, and C. Yu, "Two-stage vision system: Application of multi-perspective object detection network and character recognition network in industrial product classification," *Eng. Appl. Artif. Intell.*, vol. 156, May 2025, Art. no. 111190, doi: 10.1016/j.engappai.2025.111190.
- [4] S. Kim, S. H. Hong, H. Kim, M. Lee, and S. Hwang, "Small object detection (SOD) system for comprehensive construction site safety monitoring," *Autom. Constr.*, vol. 156, Oct. 2023, Art. no. 105103, doi: 10.1016/j.autcon.2023.105103.
- [5] A. Kos, K. Majek, and D. Belter, "Enhanced lightweight detection of small and tiny objects in high-resolution images using object tracking-based region of interest proposal," *Eng. Appl. Artif. Intell.*, vol. 153, Apr. 2025, Art. no. 110852, doi: 10.1016/j.engappai.2025.110852.
- [6] Y. Wu, "Fusion-based modeling of an intelligent algorithm for enhanced object detection using a Deep Learning Approach on radar and camera data," *Inf. Fusion*, vol. 113, Aug. 2024, Art. no. 102647, doi: 10.1016/j.inffus.2024.102647.
- [7] S. B. J. Khan, C. Li, and P. Zhang, "FocusTrack: Enhancing object detection and tracking for small and ambiguous objects," *J. Vis. Commun. Image R.*, vol. 111, Aug. 2025, Art. no. 104549, doi: 10.1016/j.jvcir.2025.104549.
- [8] M. Nikouei et al., "Small object detection: A comprehensive survey on challenges, techniques and real-world applications," *Intell. Syst. Appl.*, vol. 27, Jul. 2025, Art. no. 200561, doi: 10.1016/j.iswa.2025.200561.
- [9] S. Fu, Q. Zhao, H. Liu, Q. Tao, and D. Liu, "Lowlight object detection via adaptive enhancement and dynamic feature fusion," *Alexandria Eng. J.*, vol. 126, Apr. 2025, pp. 60–69, doi: 10.1016/j.aej.2025.04.047.
- [10] Iqra and K. J. Giri, "SO-YOLOv8: A novel deep learning-based approach for small object detection with YOLO beyond COCO," *Expert Syst. Appl.*, vol. 280, Apr. 2025, Art. no. 127447, doi: 10.1016/j.eswa.2025.127447.
- [11] D. Nimma et al., "Object detection in real-time video surveillance using attention based transformer YOLOv8 model," *Alexandria Eng. J.*, vol. 118, Jan. 2025, pp. 482–495, doi: 10.1016/j.aej.2025.01.032.
- [12] J. W. Ma, T. Czerniawski, and F. Leite, "An application of metadata-based image retrieval system for facility management," *Future Generation Computer Systems*, vol. 8, pp. 265–288, Mar. 2020, doi: 10.1016/j.future.2017.11.015.
- [13] P. Choudhary, A. Singh, A. K. Choudhary, and A. P. Srivastava, "Find Mine: Find the Lost Items via Mobile App," in *2021 2nd International Conference on Intelligent Engineering and Management (ICIEM)*, 2021, pp. 491–495, doi: 10.1109/ICIEM51511.2021.9445379.
- [14] J. Yang, B. Jiang, and H. Song, "A distributed image-retrieval method in multi-camera system of smart city based on cloud computing," *Future Generation Computer Systems*, vol. 84, pp. 11–26, Jul. 2018, doi: 10.1016/j.future.2017.11.015.