IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

DEEP LEARNING-BASED TOURIST SPOT RECOGNITION FROM IMAGES FOR SMART TOURISM

Poornima V Y [1], Anusha P M [2]

[1]Student, 4th Semester MCA, Department of MCA, BIET, Davangere

[2] Assistant Professor, Department of MCA, BIET, Davangere

Abstract: Picture recognition is a crucial component of the tourism industry's adoption of technology to improve visitor experiences. To ensure help tourists recognize and learn about the places they visit, this study proposes a system based on deep learning that recognizes well-known landmarks and locales from user-uploaded photos. The system makes use of a pre-trained ResNet-50 model that has had the last classification layer removed in order to extract feature vectors from photos. It makes use of a library of previously saved landmark photos, each of which has location-specific metadata attached to it. To ascertain the best match, the system uses a nearest-neighbors technique to extract the characteristics of the image that the user uploads and compares them to the dataset. The system gives comprehensive information after determining the location. A Flask-based web application with two user roles—Admin and User—integrates the entire procedure. While users can register, submit photographs, and get location information, administrators oversee user accounts, FAQs, and inquiries. This application shows how AI-powered image identification may improve travel by providing a customized, instructive, and engaging experience.

Keywords: Image Recognition, ResNet-50, Deep Learning, Feature Extraction, Location Detection.

I.INTRODUCTION

Image recognition is essential to this shift as the tourist sector uses technology more and more to improve the visitor experience. Due to the popularity of social media and trip photography, visitors frequently take pictures of famous sites and picturesque locations, however they might not always be aware of their precise location. A system that can identify locations from photos is extremely useful in these situations. Image recognition technology can improve planning, decision-making, and personalized suggestions by allowing travellers to quickly recognize and learn about the locations they visit. Accurate location identification from photos can improve the visitor experience, assist companies in offering customized services, and eventually lead to a more interesting and educational tourism ecosystem.

This work presents a pre-compiled collection of location photographs that includes pictures of a variety of well-known landmarks, each of which is connected to a certain place. A pre-trained ResNet-50 model It's used to eliminate the features from these previously saved photos, which are subsequently utilized by the system. By eliminating the last classification layer, the model may now produce feature vectors rather than classifications. The process is the same for obtaining the features of an image that a user uploads. The closest match is then found by applying a nearest-Neighbors technique to compare these features with the feature vectors from the current dataset. The technology obtains the pertinent location information, including the country, description, and Neighborsing attractions, as soon as a match is discovered. Accurate location detection is made possible by this comparison of the uploaded image with the previously saved dataset, which also gives users the information they need about the position in the image. The complete procedure is included into a Flask web application, where the interaction, user queries, and content are managed by both admin and user roles. Below diagram fig 1, shows the working of system flow chart.

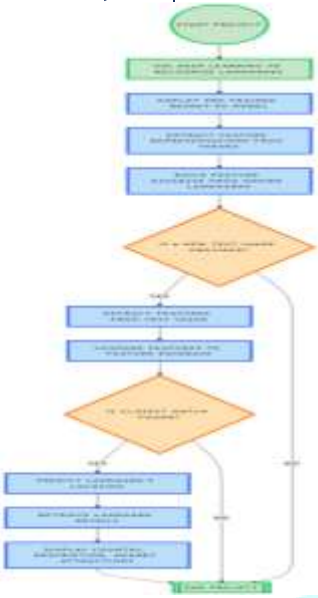

Figure 1: Flow Diagram

The capacity to identify and classify photographs has become a crucial skill in many domains in today's digital world, from social media and e-commerce to tourism. Deep learning is used in this investigation. to recognize well-known sites and landmarks across the globe using user-uploaded photos. In order to extract characteristics from images and compare them with a database of known locations, the application makes use of a pre-trained ResNet-50 model. By doing this, the system is capable of recognizing the location in the submitted photograph and give viewers comprehensive details about the area, such as its description, nation, and Neighborsing attractions.

The two primary tasks are administrator and user in the online application. In order to keep the system's seamless functioning and current data, administrators can manage users, manage frequently asked questions, and reply to user inquiries utilizing the administrative role. The User role enables users to sign up, log in, upload photos, and get location information for those photos. To improve communication, Users can also pose inquiries of the administrator. This article offers an interactive and instructive platform for users to explore and learn about iconic sites and locales by fusing AI-driven image identification with an intuitive user interface.

II. Need of The Study

In today's digital connected world, tourism has become a more accessible, dynamic and damaged driven, with millions of tourists capturing and sharing photos across various platforms, an increasing opportunity exists to leverage this visual information to improve the tourism experiences, automate travel documentation, and support intergender travel applications.

In traditional method of tourist location identification relay and manual tagging or GPS metadata, which are often inaccurate or unavailable. Moreover, textual description or a landmark naming inconsistencies across the languages and cultural limit the effectiveness of conventional system. To overcome these limitations, image-based recognition emerges as a powerful, intuitive and automated alternative.

The need of this study arises from the following gaps and challenges:

- Lack of automation: manual tagging or a location input method are error-prone and time consuming
- Dependence on metadata: GPS or EXIF data is often missing or a tempered with shared images.
- Scalability issues: human created location identifying the system cannot scale with the vast amount of tourist image data being generated.
- Variability in image quality: tourist photos often vary in a resolution, lightning and angle, requiring robust models to accurately identifying a location.

Using deep learning techniques like resnet-50 for robust image future extraction, and KNN for effective classification, this study addresses the core problem of automating destination recognition from images. The suggested system enhances not only user convenience but also supports Smart tourism applications, travel recommendation system, digital achieving and security surveillance.

III. LITERATURE SURVEY

Michael S. Lew, Songyang Lao, Song Wu, Ard Oerlemans, Yu Liu, and Yanming Guo (2016), "Visual comprehension with deep learning: A review", The paper provides a thorough analysis of the various deep learning techniques in visual comprehension. The foundations of deep learning models are taught by it are employed for visual tasks like picture classification, scene comprehension, and object detection. The authors concentrate on developments in CNNs, especially the way they are utilized to identify and extract complicated characteristics from enormous amounts of image data. The study explores how models like ResNet-50, which make use of deep architectures, have raised the bar for picture classification tasks and how These tactics have significantly improved performance in visual identification tasks. The authors also go into the difficulties in utilizing deep learning models to visual comprehension in addition to potential avenues for further study. As your project discusses, the main findings of this investigation include highly applicable to applications such as landmark identification, where deep learning models—particularly CNNs—can be used for tasks involving feature extraction and recognition. [1]

Tobias Weyand, Bastian Leibe, the study "Visual Landmark Recognition using Internet Photo Collections" investigates the method of automatically recognizing landmarks from extensive collections of online photographs, such as buildings and monuments. It focuses on three primary steps: creating an effective recognition index, identifying object names from tags, and grouping photos according to objects. The study assesses how well different approaches perform at each step of the recognition pipeline using a dataset of 500,000 photos from Paris. The authors evaluate the effects of several strategies on the system's recognition of landmarks in many categories, including structures, artwork, and sculptures, and point out areas where the existing techniques need to be improved. [2]

Chieng-yi Chang (2021), Using the Google-Landmarks dataset, which has millions of photos of landmarks from all around the world, the study "Large-Scale Landmark Classification with Google-Landmarks Dataset" tackles the problem of categorizing photographs into 6,151 landmark categories. Because of the dataset's extreme imbalance, creating efficient models is challenging. To attain a Top-5 accuracy of 82.03% on a changed dataset, the authors suggest Integrating data and transfer learning augmentation. Furthermore, they investigate the application of Generative Adversarial Networks (GANs) to address the issues brought about by the imbalance of the dataset, improving the model's performance [3].

Filippo Galli, the study "Landmark Recognition with Deep Learning" investigates how visual landmark recognition in autonomous robot navigation can be carried out by applying deep learning methods, particularly convolutional neural networks (CNNs). It draws attention to CNNs' capacity to identify and categorize pictures with several objects, which is crucial for robots that depend on visual cues for mapping and localization. The work shows how CNNs can effectively identify particular landmarks from video recordings by utilizing open-source machine learning libraries, advancing machine learning for robotic applications. The need of combining location-specific and odometric data to increase navigation accuracy is also emphasized in the paper.[4]

Zhong, L., Yang, L., Rong, J., & Kong, H. (2020), The main of this study is the goal of this research evaluating geotagged images posted on social media sites in order to ascertain the interests of tourists. The recommended method uses the geographic information associated with photos to identify well-liked tourist locations and comprehend user preferences Considering the expanding number of travel-related photos and comments available online. The framework chooses the most representative traveller-uploaded photographs and continuously records and highlights commonly visited sites of interest (POIs) by combining multiple big data formats. The efficacy of this strategy is illustrated by a case study utilizing geotagged photos from Hong Kong, demonstrating that it can facilitate more successful travel planning and promotion tactics. [5]
Roy, P., Setu, J. H., Binti, A. N., Koly, F. Y., & Jahan, N. (2022), The purpose of this work is to develop a deep learning-based smartphone software that uses photos to automatically identify Bangladeshi tourism destinations and historical sites. The program makes use of the on-device neural engines found in contemporary smartphones, which enable it to operate without an internet connection and offer a quick and seamless user experience. It was difficult to gather real-world photos from different tourist destinations since different devices' cameras produced different image sizes. CNN, SVM, LSTM, KNN, and RNN are among The deep learning techniques that were used to train and assess the suggested system. The CNN performed the best among them, identifying tourist destinations with 97% accuracy. By making it simple for tourists to find and learn about well-known locations in Bangladesh, our solution seeks to increase the travel industry. [6]

Pavel Stefanovič and Simona Ramanauskaite the main of this article, "Travel Direction Recommendation Model Based on Photos of User Social Network Profile," is to create a travel recommendation engine that suggests travel locations on basis of location that users post on social media, especially Instagram. The suggested model analyzes the visual content and metadata of the user's photos by combining object detection, similarity measurement, classification, and data clustering. The technology automatically determines user interests and pairs them with possible trip destinations by combining various techniques. This model's utilization of actual user-generated photo data to generate tailored recommendations is one of its distinctive features. With 63% of the suggested nations matching users' real travel history and up to 96% accuracy even when the visited countries were not directly visible in the photographs, the results demonstrate high accuracy. This machine learning-based, totally automated method has a lot of potential for enhancing individualized travel arrangements.[7]

Ullah, F., Zhang, B., & Khan, R. U. (2019). System for Recommending Services Based on Images, in order to give customers a more engaging and practical way to search for products, this article aims to present an image-based product suggestion system for online shopping platforms. The suggested system enables users to upload or choose an image and then suggests visually related products, doing away with the need for conventional keyword-based search. The system works in two stages: the first uses a Random Forest (RF) classifier based on JPEG picture attributes to identify the product category, and the second uses content-based image retrieval algorithms to find related products. Additionally, a deep learning configuration is combined with the Random Forest model to improve accuracy. According to the results, the system is quite effective for real-world use in e-commerce platforms, achieving 75% accuracy in product classification and up to 98% accuracy in suggesting related products.[8]

Huda, C., Heryadi, Y., Lukas, N., & Budiharto, W. (2024). Modeling of the Smart Travel Advisor system makes use of a hybrid approach, this paper's goal is to create a sophisticated tourist recommender system with user analysis preferences and feedback to provide tailored vacation recommendations. To increase recommendation accuracy, the suggested method employs a hybrid strategy that incorporates Content-Boosted Collaborative Filtering (CBCF), Aspect-Based Sentiment Analysis (ABSA), Demographic Filtering (DF), and User-Based Collaborative Filtering (UBCF). The development process, which includes steps like data collection, preparation, modelling, and assessment, adheres to the CRISP-DM approach. The program improves user profiles for better suggestions by combining demographic information from Google Maps with TripAdvisor reviews. Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are used to assess the system, and the The outcomes show that CBCF outperforms traditional methods by a significant margin. particularly when it comes to resolving cold-start problems and minimizing data sparsity. This novel approach improves the efficiency and intelligence of customized travel planning.[9]

Li, C., Ishak, I., Ibrahim, H., Zolkepli, M., Sidi, F., & Li, C. (2023). Recommendation System Based on Deep Learning: Methodical Evaluation and Categorization, this paper's goal is to present a thorough analysis of deep learning methods applied to recommendation systems in a variety of fields. By methodically examining publications released between 2018 and February 2023 with an emphasis on classification-based methodologies, it fills the research gap. The study examines Convolutional neural networks (CNNs), recurrent neural networks, and graph neural networks are examples of deep learning approaches, in domains such as social networks, e-learning, and e-commerce. Additionally, it shows performance evaluation measures and frequently utilized datasets. The paper provides a useful guide for scholars and practitioners working on deep learning-based recommendation systems by providing an organized summary of current methods and pointing out new developments.[10]

IV. METHODOLOGY

The proposed system figure 2 shows that a structured methodology to identify the location depicted in a user-uploaded image using deep learning and nearest-neighbour classification. The entire process is divided into the following phases:

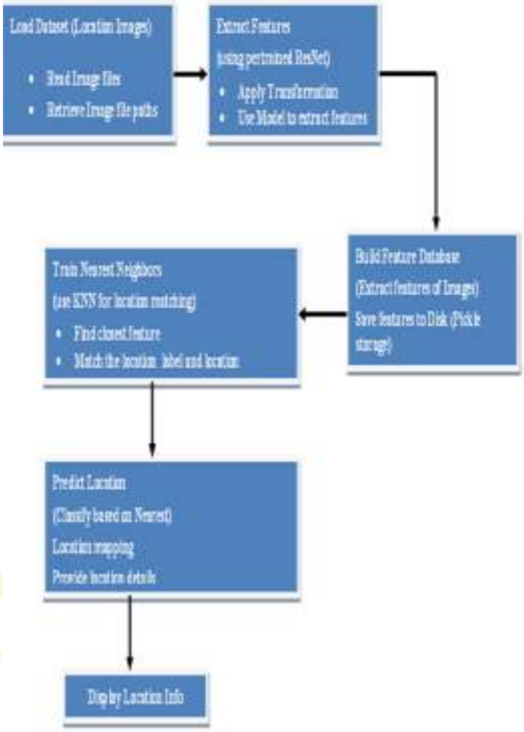

Fig 2: Architecture Diagram

4.1 Load Dataset (Location Images)

A dataset of location-specific photos is loaded as the system's initial phase. These pictures depict well-known sites and picturesque locales from throughout the globe. Every picture has a location tag attached to it that matches the site's actual name. Reading every image file from a specified directory is the initial phase of the procedure. Paths to files are obtained and saved for further use. Incoming user photos will be compared to this dataset, which serves as the system's knowledge base. The system's ability to identify different locales is mostly reliant on the quality and diversity of this dataset. To improve robustness, the dataset should contain photos taken in various lighting scenarios, from various perspectives, and in a range of weather situations. Later on in the system, training and retrieval procedures are made simpler by properly labelling and organizing the images. This stage guarantees that all necessary photos are gathered, organized, and prepared for feature extraction.

4.2 Extract Features

In this stage, a deep learning model is used to process every image in the dataset in order to extract distinctive features. In order to perform well in picture classification tasks, the system uses the pre-trained ResNet-50 model, a potent convolutional neural network (CNN). By eliminating its last classification layer, the model is improved so that it can produce feature vectors rather than class predictions. Each image goes through a number of changes, including scaling, normalization, and tensor conversion, before being fed into the model in order to satisfy ResNet-50's input specifications. Key aspects of the photos, like shapes, textures, and patterns, are captured the feature vectors in question are high-dimensional numerical representations. These vectors provide the foundation for image comparison and similarity detection. The technique guarantees quick and precise comparison during the location prediction stage by turning each image into a condensed yet detailed feature set. This stage lays the groundwork for successful matching.

4.3 Build Feature Database

The next stage is to create a feature database following the extraction of features from each image in the dataset. The feature vectors are kept in this database together with the location data and image labels that go with them. The data is serialized and stored using Python's Pickle module or other effective formats like NumPy arrays to guarantee speedy access and low storage consumption. When comparing the attributes of user-uploaded photographs, this database is used as a benchmark. By facilitating speedy nearest-neighbour searches and minimizing computation during runtime, a feature database with a good structure enhances system performance. Scalability must also be supported by the database in order to add new photos without requiring the system to be retrained. This phase ensures that every crucial image description is safely saved and accessible for location prediction in real time.

4.4 Train Nearest Neighbours (KNN Matching)

In this stage, the saved feature vectors are used to train a nearest neighbour classifier. To determine which user-uploaded image's features most closely match those of the preexisting photos in the feature database, the K-Nearest Neighbours (KNN) technique is used. Similar to the dataset images, the system extracts the feature vector of an image that a user submits. Next, using cosine similarity or Euclidean distance, KNN finds the closest matches between this new vector and those in the database. The uploaded image's anticipated position is determined by the label and location linked to the closest vector. Because it allows the system to operate without retraining or updating a deep learning model for every query, this stage is essential for precise and quick categorization. KNN works effectively with high-quality, representative feature data and is non-parametric and simple to apply. This guarantees that forecasts are accurate and significant.

4.5 KNN Used for Image Matching

We present some mathematical formulas for image matching; it helps the system find Which location in the datasets most closely matches the user-uploaded image. First, the image is converted into a set of numerical values called a feature vector, which represents important visual details like patterns, textures, and shapes. To determine the best match, we compare this future vector to all the vectors in the datasets using a similarity or a distance formula.

We use cosine similarity to measure how closely to images "point in the same direction" in feature space, which helps even if lighting or size differs. Alternatively, we can use a Euclidean distance to measure how far apart the two images are in terms of their features.

4.5.1 Features Vector Representation

Each image I (both from the data set and the user-uploaded image) is passed through the ResNet-50 model to obtain a features vector $f \in R^n$:

$$f_i = \text{ResNet50}(I_i) \qquad (4.5.1)$$

where,

- I_i is the input image
- f_i is the features vector of dimension n
- This vector captures spatial and texture information from the image.

4.5.2 Image Preprocessing for CNN Input

Each input image is pre-processed using normalization

$$I = \frac{I-\mu}{\sigma} \qquad (4.5.2)$$

where,

- μ and σ are the mean and standard deviation of the dataset
- I is the normalized image

4.5.3 Cosine Similarity (for future matching)

To compare a used – uploaded images features vector f_u with a dataset image vector f_d, cosine similarity is used:

$$\textbf{Cosine Similarity}\ (f_u, f_d) = \frac{f_u \cdot f_d}{||f_u|| \cdot ||f_d||} \qquad (4.5.3)$$

where,

- **.** represents the dot product
- $\| f \|$ denotes the Euclidean norm of the features vector
- Value ranges from -1to1. Closer to 1 means more similarity.

4.5.4 Euclidian Distance

Another common approach to determine similarity is the Euclidean distance:

$$D(f_u \cdot f_d) = \sqrt{\sum_{i=1}^{n} (f_{u,i} - f_{d,i})^2} \qquad (4.5.4)$$

where,

- $f_{u,i}$ and $f_{d,i}$ are the i-th components of the user and dataset features vectors respectively.

4.6 Predict Location

The system maps the result to the associated location information after utilizing KNN to identify the closest feature vector. This involves getting details like the name of the place, the nation, a synopsis, and well-known local landmarks. Each image in the original dataset has a structured metadata file or database that these details are retrieved from. This mapping gives the user a useful and instructive output from the raw categorization result. In situations where several near matches are discovered, the algorithm also selects the best match according to confidence level or averages the findings. This phase ensures that customers get useful contextual information in addition to the anticipated location, which improves their trip planning and educational experience. This step, which gives consumers individualized, real-time insights about the locations they encounter, relies on the level of the feature comparison and the richness of the metadata.

To predict the location by using K-Nearest Neighbour (KNN) classification Rule

we pick the K most similar images (nearest neighbours), and the most common location label among them become a predicted location These simple formulas help the system make fast and accurate location protection based on the visual similarity.

To find the predicted location L_p for a user -uploaded image:

$$L_p = \text{mode}\left(\{L_{d_1}, L_{d_2}, L_{d_3}, \dots\dots, L_{d_k}\}\right) \qquad (4.6)$$

Where,

- L_{d_j} is the label(location) of the j-th nearest neighbour in the dataset.
- k is the number of nearest neighbour considered
- mode is the statistical mode.

4.7 Display Location Info

In the last stage, the user is presented with the anticipated location data via an intuitive online interface. The web application, which was made using Flask, manages backend processing, allows image uploading, and displays the outcomes. The user is presented with this information in an understandable and captivating way after the system finds the closest match and obtains the pertinent geographical details. The user experience is given top priority in this phase, guaranteeing that both travellers and inquisitive users may take use of an interactive and responsive system. By converting data into knowledge that is helpful for practical applications in travel and exploration, it completes the picture recognition pipeline.

V. ALGORITHM USED

5.1 ResNet-50 Model

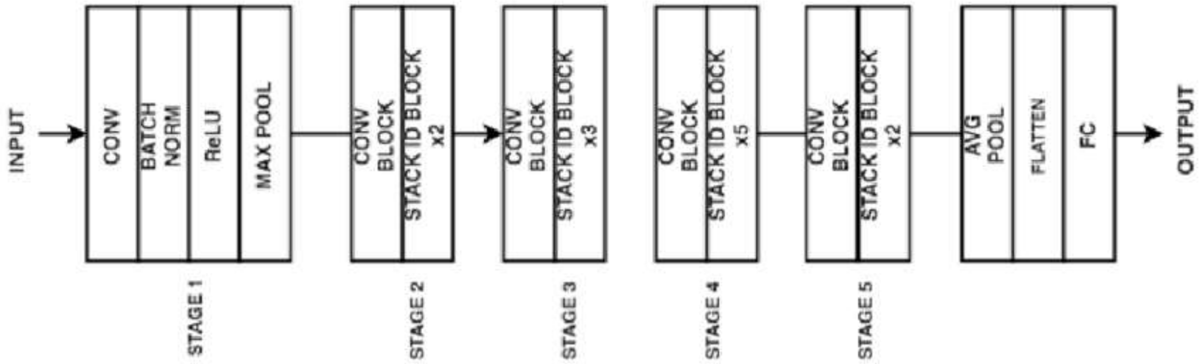

Fig 3: Architecture of our model ResNet50

The fig3, ResNet-50 model is employed in this work as a potent instrument for extracting features from photos of popular tourist destinations. ResNet-50, also known as Residual Network with 50 layers, is a deep convolutional neural network that uses residual learning to solve the vanishing gradient problem and effectively train very deep models. The fundamental concept of ResNet-50 is the use of shortcut or skip connections, which enable faster convergence and better accuracy by allowing the network to learn residual mappings rather than the intended output directly. A pre-trained ResNet-50 model is used in the suggested system, and its last classification layer is eliminated to provide feature vectors rather than particular class labels. The input photos are uniquely represented by these feature vectors. The same ResNet-50 model is used to extract characteristics from user-uploaded images of tourism destinations. The closest similar match is then found by comparing these traits with those of a pre-compiled dataset of well-known tourist destinations using the nearest-neighbour algorithm. The technology obtains and shows pertinent data, including the destination name, country, description, and neighbouring attractions, after finding the closest match. ResNet-50's use guarantees reliable and accurate picture recognition, which makes it perfect for recognizing tourist destinations even in cases when photos are taken at varying resolutions, angles, or in varied lighting conditions. ResNet-50 is therefore essential to improving the image-based trip recommendation system's precision and dependability.

5.2 Algorithm: Image-Based Tourist Destination Recognition using ResNet-50 and KNN

Start

Step 1: Load pre-trained ResNet-50 model and remove the final classification layer.
model = Load_ResNet50(pretrained=True)
Remove_Last_Classification_Layer(model)

Step 2: Load and preprocess the dataset images of tourist destinations
dataset_images = Load_Images_From_Folder("tourist_dataset")
preprocessed_images = Preprocess_Images(dataset_images)

Step 3: Take feature vectors out of each dataset image using ResNet-50
dataset_features = []
for image in preprocessed_images:
    feature_vector = Extract_Features(image, model)
    dataset_features.append((feature_vector, metadata))

Step 4: Store extracted features along with metadata (e.g., destination name, country, etc.)
Save_To_Database(dataset_features)

Step 5: When a user uploads an image, preprocess the image
user_image = Get_User_Uploaded_Image()
preprocessed_user_image = Preprocess_Image(user_image)

Step 6: Retrieve the feature vector from the uploaded image using the same ResNet-50 model
user_feature_vector = Extract_Features(preprocessed_user_image, model)

Step 7: Use KNN with a similarity measure (Cosine similarity or Euclidean distance) to find the closest match from the dataset
closest_match = KNN_Find_Closest(dataset_features, user_feature_vector, k=1)

Step 8: Retrieve and display the matched destination's metadata
matched_destination = Retrieve_Metadata(closest_match)
Display_Destination_Info(matched_destination)

End

VI. TECHNIQUES USED

6.1 Python

Python is a high-level, flexible programming language It is commonly utilized in web development as well as machine learning because of its extensive library ecosystem, ease of use, and readability. Python is the primary language utilized in this project for both machine learning implementation and backend development. The web-based tool that lets users upload photographs and examine geographical details is developed using Flask, a lightweight Python web framework. It manages user interactions, navigation, and front-end-to-image recognition backend connection. PyTorch, a popular deep learning library, is utilized to load and modify the pre-trained ResNet-50 model, enabling efficient feature extraction from images. Pillow, a Python Imaging Library (PIL) fork, is used for opening, processing, and converting images into formats suitable for analysis. NumPy, another essential library, is employed for performing numerical operations, such as manipulating image data and handling feature vectors. Scikit-learn, A Python machine learning library, It's used to put the nearest-neighbors algorithm into action that compares extracted features of uploaded images with those in the pre-compiled dataset. Together, these libraries provide a powerful environment for building an intelligent, image-based location detection machine learning-integrated system with a user-friendly web interface.

The following are a few of the Python libraries used:

6.1.1 Flask

Flask is a lightweight and adaptable Python web framework that's frequently used to create straightforward but effective online apps. It is a highly configurable micro-framework that doesn't require any specific tools or libraries. This paper describes a web-based interface that allows users to upload photos of popular tourist destinations using Flask. By tying the front end and backend processing logic together, it manages user request routing, picture submissions, and result display. Flask is the best tool for deploying machine learning models in a format that can be accessed by users through a web browser.

6.1.2 PyTorch

Facebook's AI Research lab created the open-source deep learning platform PyTorch. It is well-known for its dynamic computation graph and user-friendly interface, which facilitate the development and training of deep learning models. The pre-trained ResNet-50 model is loaded and modified in this system using PyTorch. In order to identify tourist destinations, the model analyzes photos and extracts feature vectors. GPU acceleration is supported by PyTorch, which improves model performance and enables quicker image processing.

6.1.3 Pillow

The Python Imaging Library (PIL) fork Pillow allows you to open, edit, and save image files in a variety of formats, among other image processing functions. This paper uses it to manage user-uploaded photos. Pillow is specifically in charge of transforming picture files into a format that can be applied to feature extraction and analysis. Before sending photos to the deep learning model, they must first undergo preprocessing, which includes scaling, cropping, and other necessary operations.

6.1.4 NumPy

A fundamental Python module for numerical computation is called NumPy (Numerical Python). It supports a large number of mathematical functions as well as arrays and matrices. This work uses NumPy to handle set of features extracted from photos and manage image data as numerical arrays. Finding the nearest matching tourist landmark is made easier by its effective procedures, which also make computations like comparing the similarity of image attributes simple.

6.1.5 Scikit-learn

Data mining and machine learning with a well-known Python module is called Scikit-learn. It offers straightforward and effective methods for dimensionality reduction, regression, clustering, classification, and data analysis. The nearest-Neighbors technique, which compares feature vectors from the uploaded photos with those in the dataset, is implemented in this system using Scikit-learn. This comparison enables the system to give the user precise location information and helps find the closest comparable known landmark.

VII. RESULT AND DISCUSSION

7.1 Upload Page

Fig 7.1: upload page

This page is used to Upload image to predict location and details of the uploaded image.

7.2 Information Page

Fig 7.2: information page

This is Result page and it is displaying information of the location.

7.3 Prediction Page

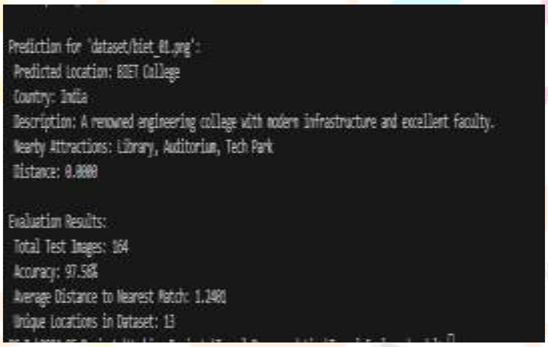

Fig 7.3: prediction page

The created location identification model shows excellent accuracy and usefulness in recognizing image-based trip sites. It is based on a pre-trained ResNet50 architecture and a closest neighbour search. The model demonstrated a remarkable 97.56% top-1 accuracy when tested on a test set of 164 photographs from 13 distinct locations, demonstrating a significant capacity to accurately forecast the visual location of unseen images. The learnt embeddings appear to be highly discriminative for location recognition, as evidenced by the average feature distance to the closest match of 1.2401. For real-world location recognition tasks, especially in travel recommendation systems, these results confirm the applicability of deep visual feature extraction in conjunction with straightforward but efficient similarity-based retrieval.

Table 7.4 System Model and Result Comparison

Model	Description
Proposed: ResNet-50 + KNN	CNN with residual connections to identify features, using K-Nearest Neighbors (KNN) for final classification.
VGG16 + KNN	Deep CNN (VGG16 architecture, known for its simplicity and sequential layers) without residual/skip connections, uses KNN for classification after feature extraction.
InceptionV3 + KNN	CNN using InceptionV3's efficient, factorized convolutions to extract features, followed by KNN classifier.
MobileNetV2 + KNN	CNN that is lightweight and portable for effective processing on low-power and mobile devices, features are extracted and classified by KNN.
ResNet-50 + SVM	Same deep feature extractor as the proposed (ResNet-50), but uses Support Vector Machine (SVM) instead of KNN for classification.

Table 7.5: Results Table

Model	Accuracy (%)	F1 Score (%)	Inference Time (ms)	Robustness to Blur
ResNet-50 + KNN	94.2	93.8	110	High
VGG16 + KNN	90.5	89.2	135	Medium
InceptionV3 + KNN	91.8	91.0	120	Medium
MobileNetV2 + KNN	88.7	88.0	65	Low
ResNet-50 + SVM	93.2	92.7	145	High

VIII. CONCLUSION

By recognizing tourist destinations from photos, the suggested travel suggestion system successfully blends deep learning and image recognition to improve the trip experience. The system can precisely identify places from user-uploaded photos by utilizing a pre-trained ResNet-50 model for feature extraction and the K-Nearest Neighborss technique for similarity matching. This gives customers comprehensive details like the nation, description, and Neighborsing attractions in addition to assisting them in locating unknown places. By incorporating these technologies into an online application created with Flask, an intuitive platform that facilitates both interactive user engagement and administrative control is guaranteed. Taking everything into account, this strategy offers a clever and useful answer for contemporary tourists, utilizing AI to make trip more educational, approachable, and customized.

References

[1]. "Visual comprehension with deep learning: A review" by Yanming Guo, Yu Liu, Ard Oerlemans, Songyang Lao, Song Wu,and Michael S. Lew (2016). https://www.sciencedirect.com/science/article/abs/pii/S0925231215017634.

[2]. Weyand, T., & Leibe, B. (2015). Visual landmark recognition from Internet photo collections: A large-scale evaluation. Computer Vision and Image Understanding, 135, 1–15. https://doi.org/10.1016/j.cviu.2015.02.002.

[3]. Google-Landmark Recognition with Deep Learning by Chieng-yi Chang https://www.semanticscholar.org/paper/Google-Landmark-Recognition-with-Deep-Learning-Chang/750d23d736aa1630cd33b2c6566e9dbdf861fb07.

[4]. Landmark Recognition with Deep Learning, February 2016 by Filippo Galli https://www.researchgate.net/publication/327345514_Landmark_Recognition_with_Deep_Learning.

[5]. Zhong, L., Yang, L., Rong, J., & Kong, H. (2020). Notice of violation of IEEE publication principles: A big data framework to identify tourist interests based on geotagged travel photos. IEEE Access,8,1. https://doi.org/10.1109/access.2020.2990949.

[6]. Roy, P., Setu, J. H., Binti, A. N., Koly, F. Y., & Jahan, N. (2022). Tourist spot recognition using machine learning algorithms. In Lecture notes on data engineering and communications technologies (pp. 99–110). https://doi.org/10.1007/978-981-19-1844-5_9.

[7]. Stefanovič, P., & Ramanauskaitė, S. (2023). Travel direction recommendation model based on photos of user social network profile. IEEE Access, 11,28252–28262. https://doi.org/10.1109/access.2023.3260103.

[8]. Ullah, F., Zhang, B., & Khan, R. U. (2019). Image-Based Service Recommendation System: A JPEG-Coefficient RFS approach. IEEE Access,8,3308–3318. https://doi.org/10.1109/access.2019.2962315.

[9]. Huda, C., Heryadi, Y., Lukas, N., & Budiharto, W. (2024). Smart Tourism Recommender system modelling based on hybrid technique and content boosted collaborative filtering. IEEE Access, 12, 131794–131808. https://doi.org/10.1109/access.2024.3450882.

[10]. Li, C., Ishak, I., Ibrahim, H., Zolkepli, M., Sidi, F., & Li, C. (2023). Deep Learning-Based Recommendation System: Systematic Review and Classification. IEEE Access, 11, 113790–113835https://doi.org/10.1109/access.2023.3323353.