VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“Jnana Sangama”, Belagavi , Karnataka, INDIA

A Project Report
on

“LANDMARK RECOGNITION USING CONVOLUTIONAL NEURAL NETWORKS”

Submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Engineering
in
Computer Science and Engineering

Submitted By

C.P YASHWANTH	**1GA17CS035**
MADDALI SOWMYA	**1GA17CS080**
RAKSHITHA MURTHY	**1GA17CS118**
SWARAJ PARIDA	**1GA17CS163**

Under the Guidance of
Mr. KAMLESHWAR KUMAR YADAV
Assistant Professor

Department of Computer Science and Engineering
Accredited by NBA(2019-2022)
GLOBAL ACADEMY OF TECHNOLOGY
Rajarajeshwarinagar, Bengaluru - 560 098

2020 – 2021

GLOBAL ACADEMY OF TECHNOLOGY

Department of Computer Science and Engineering

Accredited by NBA(2019-2022)

CERTIFICATE

Certified that the Project Entitled **"Landmark Recognition using Convolutional Neural Networks"** carried out by **C.P Yashwanth**, bearing **USN 1GA17CS035, Maddali Sowmya**, bearing **USN 1GA17CS080, Rakshitha Murthy**, bearing **USN 1GA17CS118, Swaraj Parida**, bearing **USN 1GA17CS163,** bonafide students of Global Academy of Technology, in partial fulfillment for the award of the **BACHELOR OF ENGINEERING** in **Computer Science and Engineering** from Visvesvaraya Technological University, Belagavi during the year 2020-2021. It is certified that all the corrections/suggestions indicated for Internal Assessment have been incorporated in the Report submitted to the department. The Partial Project report has been approved as it satisfies the academic requirements in respect of the project work prescribed for the said Degree.

Mr. Kamleshwar K Yadav Assistant Professor Dept. of CSE GAT, Bengaluru.	Dr. Bhagyashri R Hanji Assoc. Professor & HOD Dept. of CSE GAT, Bengaluru.	Dr. N. Rana Pratap Reddy Principal GAT, Bengaluru.

External Viva

Name of the Examiners Signature with date

1

2.

GLOBAL ACADEMY OF TECHNOLOGY

Rajarajeshwarinagar, Bengaluru – 560 098

DECLARATION

We, **C.P Yashwanth**, bearing **USN 1GA17CS035, Maddali Sowmya**, bearing **USN 1GA17CS080, Rakshitha Murthy**, bearing **USN 1GA17CS118, Swaraj Parida**, bearing **USN 1GA17CS163,** students of Eighth Semester B.E, Department of Computer Science and Engineering, Global Academy of Technology, Rajarajeshwarinagar Bengaluru, declare that the Project Work entitled **"Landmark Recognition using Convolutional Neural Networks"** has been carried out by us and submitted in partial fulfillment of the course requirements for the award of degree in **Bachelor of Engineering** in **Computer Science and Engineering** from **Visvesvaraya Technological University, Belagavi** during the academic year **2020-2021**.

C.P YASHWANTH	1GA17CS035
MADDALI SOWMYA	1GA17CS080
RAKSHITHA MURTHY	1GA17CS118
SWARAJ PARIDA	1GA17CS163

Place: Bengaluru

Date: 02/08/2021

ABSTRACT

The world of Computer Vision is vast and complex. This world can bring about problems that is resilient to scale along with maintaining the accuracy. Identifying landmarks in large-scale is problematic. Number of landmarks in a dataset can be sky-high and can be still scalable, so identifying one landmark out of a bulk will cost us high computational resources and time. Thus, the approach of this paper is to do a comparative study between two different models to ensure how they individually and significantly impact on the results. Existing datasets are either enormous or lack variety of landmarks. Consequently, we are introducing a dataset which is coherent amongst different categories and is not computationally heavy to use.

Recognition is done using Deep Neural Network with CNN layers and trying to extract features of an image. For retrieving information from a large-scale dataset, we are comparing feature vector of query image with all the nearest neighbor feature vectors from the dataset. This is done with respect to Deep Local Feature (DeLF) module, which is available on TensorFlow Hub, can be used to substitute alternative keypoint detectors and descriptors for image retrieval. It uses feature descriptors, which are 40-dimensional vectors that describe each notable point in an image.

ACKNOWLEDGEMENT

The satisfaction and the euphoria that accompany the successful completion of any task would be incomplete without the mention of the people who made it possible. The constant guidance of these persons and encouragement provide, crowned our efforts with success and glory. Although it is not possible to thank all the members who helped for the completion of the phase - 2 of project individually, we take this opportunity to express our gratitude to one and all.

We are grateful to management and our institute **GLOBAL ACADEMY OF TECHNOLOGY** with its very ideals and inspiration for having provided us with the facilities, which made this, phase - 1 project a success.

We express our sincere gratitude to **Dr. N. Rana Pratap Reddy,** Principal, Global Academy of Technology for the support and encouragement.

We wish to place on record, our grateful thanks to **Dr. Bhagyashri R Hanji**, HOD, Department of CSE, Global Academy of Technology, for the constant encouragement provided to us.

We are indebted with a deep sense of gratitude for the constant inspiration, encouragement, timely guidance and valid suggestion given to us by our guide **Mr. Kamleshwar Kumar Yadav,** Assistant Professor, Department of CSE, Global Academy of Technology.

We are thankful to all the staff members of the department for providing relevant information and helped in different capacities in carrying out this phase -2 project.

Last, but not least, we owe our debts to our parents, friends and also those who directly or indirectly have helped us to make the phase - 2 project work a success.

C.P YASHWANTH	**1GA17CS035**
MADDALI SOWMYA	**1GA17CS080**
RAKSHITHA MURTHY	**1GA17CS118**
SWARAJ PARIDA	**1GA17CS163**

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

CNN	Convolutional Neural Network
DeLF	Deep Local Features
DFD	Data Flow Diagram
MAP	Mean Average Precision
RANSAC	Random Sample Consensus
R-CNN	Region Based Convolutional Neural Networks
SVM	Support Vector Machines
UML	Unified Modeling Language
URL	Universal Resource Locator
VGG	Visual Graphics Geometry

CHAPTER 1

INTRODUCTION

1.1 Introduction to Project

Image retrieval and instance recognition are fundamental research topics which have been studied for decades. The task of image retrieval is to rank images in an index set with respect to their relevance to a query image. The task of instance recognition is to identify which specific instance of an object class is shown in the query image.

As techniques for both tasks have evolved, approaches have become more robust and scalable and are starting to “solve” early datasets. Moreover, while increasingly large-scale classification datasets like ImageNet, COCO and OpenImages have established themselves as standard benchmarks, image retrieval is still commonly evaluated on very small datasets.

Present datasets are tremendous which makes it difficult to be scaled, therefore require high computational resources and time whereas the custom dataset performs well while having vast variety and keeping all the other factors to a minimum.

1.2 Problem Definition

The world of Computer Vision is vast and complex. This world can bring about problems that is resilient to scale along with maintaining the accuracy. Identifying landmarks in large-scale is problematic. Number of landmarks in a dataset can be sky-high and can be still scalable, so identifying one landmark out of a bulk will cost us high computational resources and time. Thus, the approach of this paper is to do a comparative study between two different models to ensure how they individually and significantly impact on the results. Existing datasets are either enormous or lack variety of landmarks. Consequently, we are introducing a dataset which is coherent amongst different categories and is not computationally heavy to use.

1.3 Existing System

Large-scale recognition is the main concern and our project deals with the variable database size and can highly scalable. Also, the objects that we are detecting are at an instance level so, identifying features that defines the instances is the key. Present datasets are tremendous which makes it difficult to be scaled, therefore require high computational resources and time whereas our dataset performs well while having vast variety and keeping all the other factors to a minimum.

Simple CNN architectures are not built explicitly for capturing important features of landmarks and they produce extremely limited accuracy in reality. So, we test the dataset on a simple approach of identifying feature vectors of the images and using it to perform a similarity search to fasten the retrieving process. The keypoint matching is used to find inliers between query image and similar images.

Deep Local Feature (DeLF) module, which is available on TensorFlow Hub, can be used to substitute alternative keypoint detectors and descriptors for image retrieval. It uses feature descriptors, which are 40-dimensional vectors that describe each notable point in an image.

1.4 Proposed System

In this proposed system users query image is taken, it is resized and cropped to standardized size then it is fed to the DeLF module. The DeLF module returns a list descriptors and locations. Descriptors are nothing but features in the form of vectors and the locations are the location in the image where features are present. According to the descriptors K-nearest neighbour search is performed on the images present in the dataset. Now the similar images are chosen, then RANSAC geometric verification is performed on each of these tentative images and the users query image. Once the verification is done each image will have a certain number of inliers in common with the user’s query image. The image with the highest number of inliers is retrieved with its information as the output.

1.5 Objective of the Project Work

The objective of the project work is to make a model which recognizes landmarks from an image using different algorithms such as Visual Geometry Group (VGG) and Deep Local Feature (DeLF)

1.6 Scope of the Project Work

Landmark recognition, the problem comes from a famous Kaggle competition, the Google Landmark Recognition Challenge. Training set contains over 1.2 million images spread across 14,951 classes of landmarks, varying from one to thousands of images per class. This problem of extreme classification is something that is very prevalent in the data science community today with the advancement of deep learning.

1.7 Project Report Outline

Chapter 1: In this chapter an introduction to the project, with the problem definition, existing system, proposed system, objectives of the project work and the scope of the project work is mentioned.

Chapter 2: In this chapter thorough System Study of the Literature Survey is conducted.

Chapter 3: In this chapter System requirements like functional and non-functional, software and hardware requirements are mentioned.

Chapter 4: In this chapter System Design and Architecture is discussed. Data Flow Diagrams and Use Case Diagram is also shown. Finally, the Modules present in the project are also mentioned with their names, functionality, input and output.

Chapter 5: In this chapter the project implementation is shown with the procedure and the different algorithms used in the project implementation.

Chapter 6: In this chapter the project testing is done in various testing environments and is recorded according to different testing modules.

Chapter 7: In this chapter the results, along with their snapshots are discussed.

Chapter 8: In this chapter providing the conclusion, and also including the major contributions to the project and the future enhancements are also mentioned.

CHAPTER 2

LITERATURE SURVEY

2.1 System Study

Our world consists of breath-taking landmarks A landmark is a recognizable natural or artificial feature used for navigation, a feature that stands out from its near environment and is often visible from long distances. Landmarks are usually classified as either natural landmarks or man-made landmarks, both are originally used to support navigation on finding directions. A variant is seamark or daymark, a structure usually built intentionally to aid sailors navigating featureless coasts. Natural landmarks can be characteristic features, such as mountains or plateaus. Examples of natural landmarks are Table Mountain in South Africa, Mount Ararat in Turkey, Uluru in Australia, Mount Fuji in Japan and Grand Canyon in the United States. Trees might also serve as local landmarks, such as jubilee oaks or conifers In modern sense, landmarks are usually referred to as monuments or prominent distinctive buildings, used as the symbol of a certain area, city, or nation. Some examples include the Statue of Unity in Narmada, the White House in Washington, D.C., the Statue of Liberty in New York City, the Eiffel Tower are often very tall and visible from many miles around, thus often serve as built landmarks. Also town hall towers and verifies often have a landmark character.

Many researchers used various supervised and unsupervised leaning algorithms like nearest neighbors, neural networks image processing, open cv for recognition of these landmarks. But recognizing these landmarks still improving.
Our proposed project uses different techniques like VGG.16, DeLF models by setting and tuning the hyper-parameters to get good accuracy in classifying and predicting the recognition of the landmarks.

2.2 Review of Literature

[2014] [**Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation tech report**] [Jeff Donahue, Jitendra Malik, Ross Girshick, Trevor Darrell]

The intent of this paper is that the system takes an input image, extracts approximately 2000 bottom-up region proposals and computes features for each proposal using a large Convolutional Neural Network (CNN), after it classifies each region using class-specific linear Support Vector Machines (SVMs). The advantage of the paper is it provides higher accuracy than CNNs (R-CNN achieves a Mean Average Precision (MAP) of 53.7% on PASCAL VOC 201 for comparison, reports 35.1% MAP)

[2015] [**Fast R-CNN**] [Ross Girshick]

The intent of this paper is that the image is processed with several convolutional and max pooling then region of interest has been extracted from feature, map each feature vector is fed into a fully connected layers that finally branches into two outputs. The advantages of this paper are training is single-stage, using a multi-task loss, training can update all network layers, no disk storage is required for feature caching.

[2017] [**A Large-Scale Image Retrieval with Attentive Deep Local Features**] [Andr´e Araujo, Bohyung, Hyeonwoo Noh, Jack Sim, Tobias Weyand]

The intent of this paper is to extract dense features from an image by applying a fully convolutional network, using RANSAC and employ the number of inliers as the score of retrieved images. The following are the advantages of this paper, Deep Local Features (DeLF) clearly outperforms all other techniques significantly, DeLF has higher recall rate, and attention helps more than fine-tuning.

[2020] [**1st place Solution to Google Landmark Retrieval**] [SeungKee Jeon]

The intent of this paper is to use metric learning to classify numerous landmark classes and uses transfer learning with two train datasets. Efficient pooling extract features from images and a deep neural network is followed to squeeze features into smaller dimensions. After which, cosine softmax is used to classify a number of classes. The data is trained is trained in four steps where the size of the landmark classes is increased in each step which in turn increases the leaderboard score of the model. Ensemble method is used to increase the core further. The advantages of this paper are as follows: Google Landmark dataset version 2 is the biggest landmark dataset containing human made and natural landmarks. The model presents a cleaner version of Google Landmark dataset version 2 using automatic data cleaning system. By using the ensemble method, the score of private leaderboard increases to the best value which increases the accuracy of the result.

[2020] [**Google Landmarks Dataset v2 A Large-Scale Benchmark for Instance-Level Recognition and Retrieval**] [Tobias Weyand, Andr´e Araujo, Bingyi Cao, Jack Sim]
This paper uses Google landmark dataset for large scale, fine-grained instance recognition and image retrieval in the domain of human made and natural landmarks. The dataset used is consists of over 5M images and 200k distinct instance labels. Its test set consists of 118k images with ground truth annotations for both the retrieval and recognition tasks this paper uses Instance-level recognition refers to a very fine-grained identification problem, where the goal is to visually recognize a single (or indistinguishable) occurrence of an entity.

[2020] [**Supporting large-scale image recognition with out-of-domain samples**] [Christof Henkel, Philipp Singer]
The intent of this paper is the test set of the first google landmark recognition competition, which was released together with its ground truth labels, as a validation set is considered. A crucial property of this dataset is the presence of around 98% of out-of-domain images, which we will call non-landmarks. In a first step, the paper has embedded images in a high dimensional feature space using convolutional neural networks trained with an additive angular margin loss and classify images using visual similarity. Then efficiently re-ranking predictions and filter noise utilizing similarity to out-of-domain images. The advantages are: The solution utilizes global features as extracted from several different backbones, fitted with an arc margin head and arcface loss. The approach does not use local features which reduces possibility of matching with non-landmark features in the images.

Chapter 3

SYSTEM REQUIREMENTS SPECIFICATION

3.1 Functional Requirements

The functional requirements for a system describe what the system should do. These requirements depend on the type of software being developed; the general approach taken by the organization when writing requirements. The functional system requirements describe the system function in detail, its inputs and outputs, exceptions and so on.

Functional requirements are as follows:

- This project is required to execute the given model and project the expected outputs. The ultimate goal of this project is to make the model to detect the landmark present in the image and output the correct name of the landmark.
- In the training of the model the model should report the accuracy of the output.
- The tool should be enabling the user to upload the image from various formats.
- The tool should be enabling the user to upload the image of any size.
- The tool should be providing the user with the feature where the user can run the software thus expecting the best result all the time.

.

3.2 Non-Functional Requirements

Non functional requirements, as the name suggests, are requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability, response time and store occupancy. Alternatively, they may define constraints on the system such as capabilities of I/O devices and the data representations used in system interfaces.

The non-functional requirements are as follows:

- The developed model is scalable, the model is able to detect the landmark present in the image given the features of the landmark.
- The user should satisfy all the hardware and software requirements, only then the model is going to support your actions.
- The model is expected to produce the results within no time (sometimes a few seconds)

- The tool should never fail in the middle of the operation.
- The response time of the tool should be acceptable.

3.3 Hardware Requirements

- Intel i3 6^{th} gen and above
- GB ram (min)
- GB graphics card
- 100 GB HDD (min)

3.4 Software Requirements

- Python 3.0 and above
- Google colaboratory or Jupyter notebook
- TensorFlow 2.1 and above
- Anaconda 3
- Any web browser : Chrome, and Firefox
- Operating System : Windows or Linux

Chapter 4

SYSTEM DESIGN

4.1 Design Overview

System Design is the process of defining architecture, components, and modules Interfaces and data for a system to satisfy requirements. System design could see it as the application of system theory to product development. There is some overlap with the description of system analysis, system architecture and system engineering.

If the broader topic of "product development" blends the perspective of marketing, design, and manufacture into a single approach to product development. System design is therefore the process of defining and developing systems to satisfy specified requirements of the user object-oriented analysis and design methods are becoming the most widely used methods for computer system design. The UML has become the standard language in object-oriented analysis and design. It is widely used for modelling software systems and is increasingly used for high designing non-software systems and organizations.

System design is one of the most important phases of software development process. The purpose of the design is to plan the solution of the problem specified by the requirement documentation. In other words, the first step in the solution to the problem is the design of the project.

The design will continue the specification of all the modules, their interaction with other modules and the desired output from each module. The output of the design process is a description of the software architecture as shown in Figure 4.1.

4.2 System Architecture

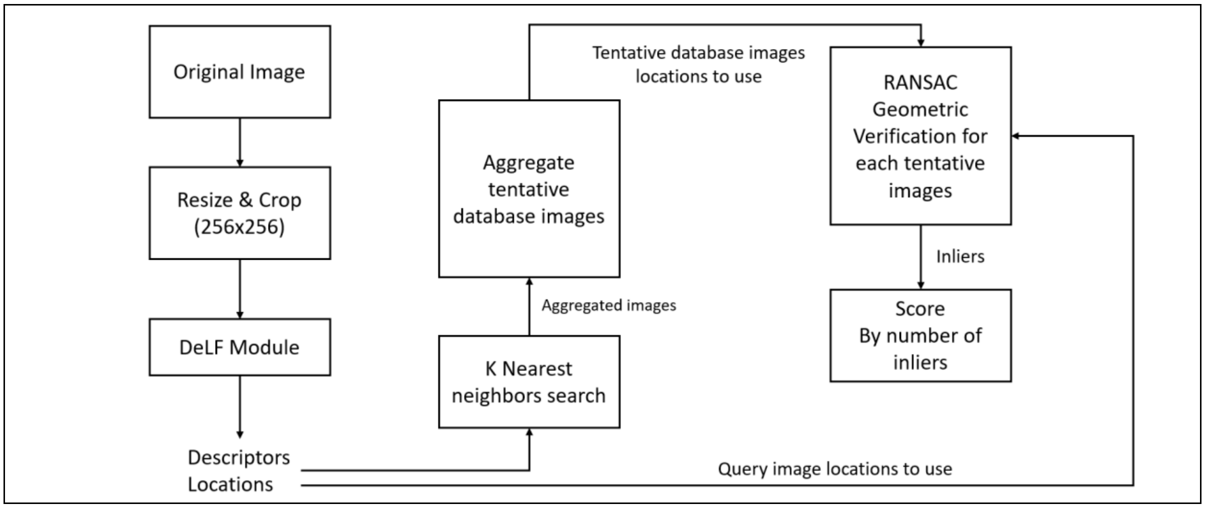

Figure 4.1: System Architecture

The system architecture represented in the figure 4.1, uses user's query image, which is resized and cropped to standardized size and then is fed to the DeLF module. The DeLF module returns a list descriptors and locations. Descriptors are nothing but features in the form of vectors and the locations are the location in the image where features are present. According to the descriptors K-nearest neighbor search is performed on the images present in the dataset. Now the similar images are chosen, then RANSAC geometric verification is performed on each of these tentative images and the users query image. Once the verification is done each image will have a certain number of inliers in common with the user's query image. The image with the highest number of inliers is retrieved with its information as the output.

4.3 Data Flow Diagrams

A data flow diagram is the graphical representation of the flow of data through an information system. Data flow diagram is very useful in understanding a system and can be efficiently used during analysis.

A data flow diagram shows the flow of data through the system. It views a system as a function that transforms the inputs into desired outputs. Any complex systems will not perform this transformation in a single step and data will typically undergo a series of transformations before it becomes the output as shown in Figure 4.2, 4.3, and 4.4.

With a data flow diagram, users are able to visualize how the system will operate that the system will accomplish and how the system will be implemented. Old system data flow diagrams can be drawn up and compared with a new data flow diagram to draw comparisons to implement a more efficient system.

DFDs can be used to provide the end user with a physical idea of where the data they input, ultimately as an effect upon the structure of the whole system.

4.3.1 Data Flow Diagram - Level 0

Figure 4.2 provides the following information: User provides the query image which is fed to the model. The model compares the query image with the already existing images in the Image Folder. It performs necessary functions and outputs a prediction back to the user.

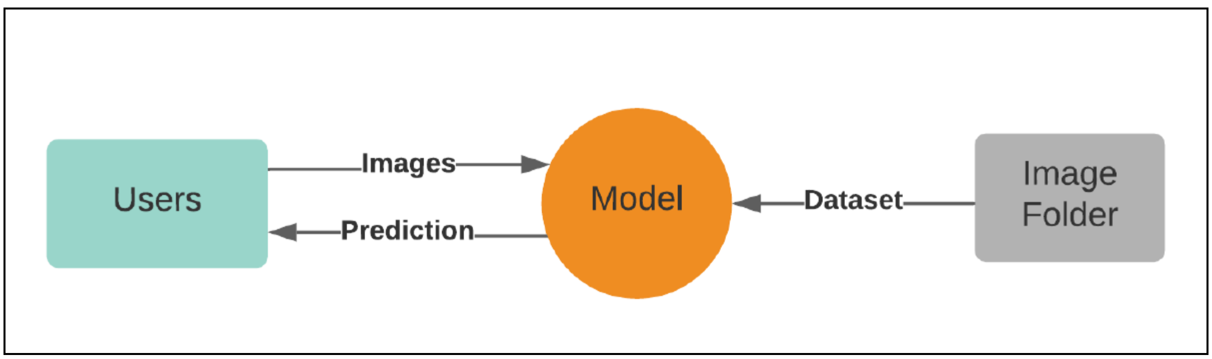

Figure 4.2 – Data Flow Diagram Level 0

4.3.2 Data Flow Diagram - Level 1

Figure 4.3 provides the following information: The user provides the query image; it is re-sized and converted into NumPy arrays. The model obtains the location and feature vectors from the NumPy arrays. The feature vectors are used to verify the query image with the database image. Finally, the geometric verification retrieves the most similar image.

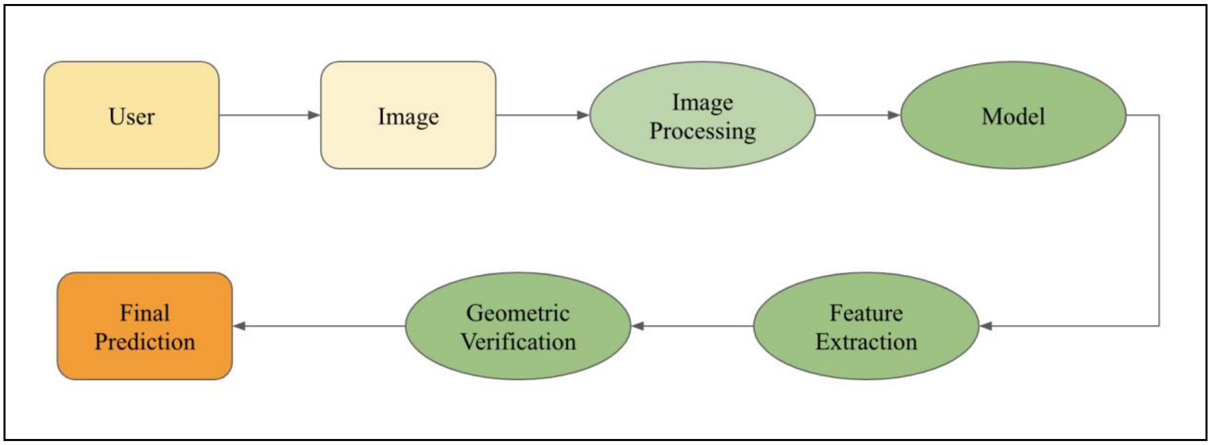

Figure 4.3 – Data Flow Diagram Level 1

4.3.3 Data Flow Diagram - Level 2

Figure 4.4 provides the following information: Image Processing is one of the modules being done which includes resizing the image, noise reduction in the image, and image augmentation will be done. The next module is Building a Model which includes Localized Feature Extraction, Dimensionality Reduction and RANSAC Geometric Verification.

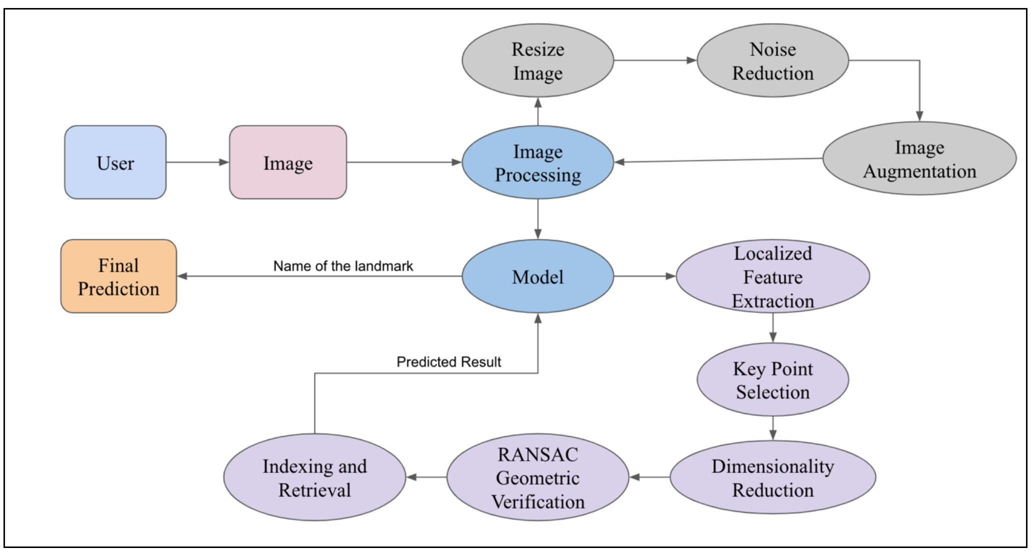

Figure 4.4 – Data Flow Diagram Level 2

4.4 Use Case Diagrams

The external objects that interact directly with the systems are called the **actors.** Actors include humans, external devices and other software systems. The important thing about actors is that they are not under control of the application. In this project, user of the system is the actor.

To find use cases, for each actor, list the fundamentally different ways in which the actor uses the system. Each of these ways is a use case as shown in Figure 4.5.

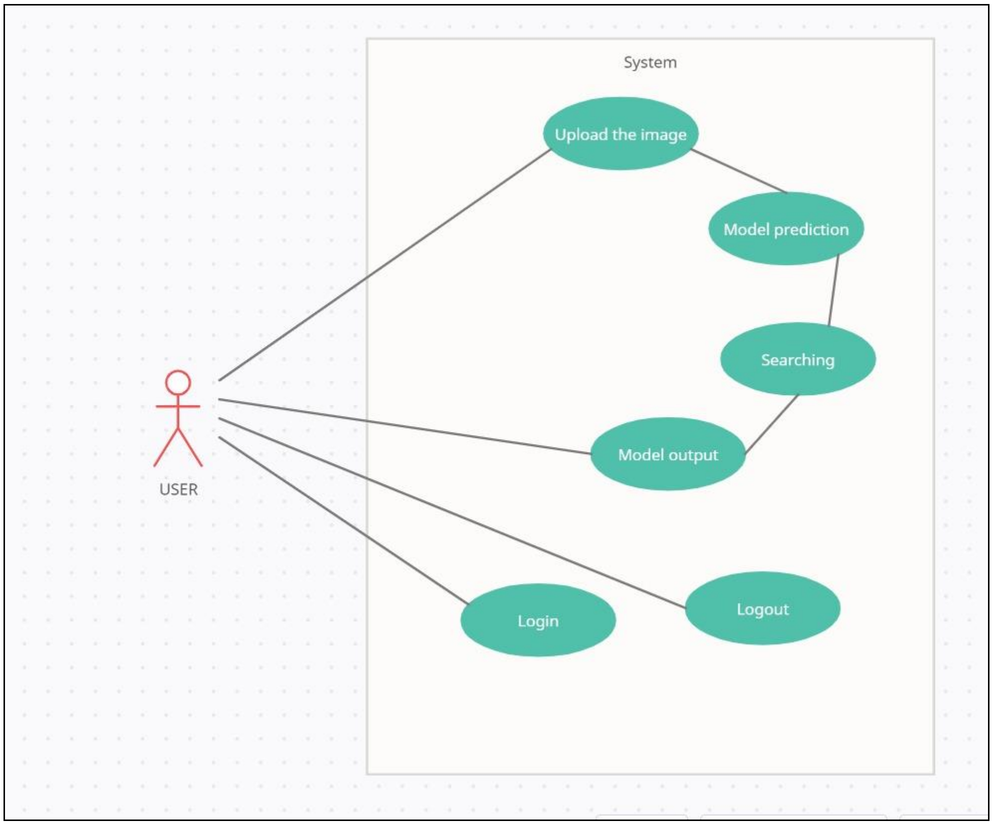

Figure 4.5 – Use Case Diagram

4.5 Modules of the Project

4.5.1 Image Acquisition

4.5.2 Image Pre-Processing

4.5.3 Feature Extraction

4.5.4 Geometric Verification

4.5.5 Final Predictions

4.5.1 Module 1

Module Name:- Image Acquisition

Functionality:- Fetching images from the given Universal Resource Locators (URLs)

Input:- Universal Resource Locators (URLs)

Output:- .Customized images

4.5.2 Module 2

Module Name:- Image Pre-Processing

Functionality:- Re-sizing images and converting images into NumPy arrays

Input:- Images of different sizes

Output: - Uniform sized images

4.5.3 Module 3

Module Name:- Feature Extraction

Functionality:- Obtaining the location and feature vectors

Input:- NumPy array

Output: - Array of location and feature vectors

4.5.4 Module 4

Module Name:- Geometric Verification

Functionality:- Verifying query image with database image

Input:- Query image and database images

Output:- Number of inliers among the matched images

4.5.5 Module 5

Module Name:- Final Prediction

Functionality:- To retrieve the most similar image

Input:- Number of inliers

Output:- Image with the highest number of inliers

Chapter 5

IMPLEMENTATION

Implementation is the process of converting a new system design into an operational one. It is the key stage in achieving a successful new system. It must therefore be carefully planned and controlled. The implementation of a system is done after the development effort is completed.

5.1 Steps for Implementation

- Installation of Hardware and Software utilities.
- Testing the developed system with sample data.
- Detection and correction of errors.
- Data updating

5.2 Implementation Issues

The implementation phase of software development is concerned with translating design specifications into source code. The primary goal of implementation is to write source code and internal documentation so that conformance of the code to its specifications can be easily verified and so that debugging testing and modification are eased. This goal can be achieved by making the source code as clear and straightforward as possible. Simplicity clarity and elegance are the hallmarks of good programs and these characteristics have been implemented in each program module.

The goals of implementation are as follows.

- Minimize the memory required.
- Maximize output readability.
- Maximize source text readability.
- Minimize the number of source statements.
- Minimize development time.

5.3 Algorithms

5.3.1 Algorithm 1 – KDTree Nearest Neighbor Search

```
def NN(Point Q, kdTree T, int cd, Rect BB):
    // if this bounding box is too far, do nothing
    if T == NULL or distance(Q, BB) > best_dist: return
    // if this point is better than the best:
    dist = distance(Q, T.data)
    if dist < best_dist:
        best = T.data
        best_dist = dist
    // visit subtrees is most promising order:
    if Q[cd] < T.data[cd]:
        NN(Q, T.left, next_cd, BB.trimLeft(cd, t.data))
        NN(Q, T.right, next_cd, BB.trimRight(cd, t.data))
    else:
        NN(Q, T.right, next_cd, BB.trimRight(cd, t.data))
        NN(Q, T.left, next_cd, BB.trimLeft(cd, t.data))
```

Chapter 6

TESTING

This chapter gives the outline of all testing methods that are carried out to get a bug free system. Quality can be achieved by testing the product using different techniques at different phases of the project development. The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components sub assemblies and/or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

6.1 Test Process

Testing is an integral part of software development. Testing process certifies whether the product that is developed compiles with the standards that it was designed to. Testing process involves building of test cases against which the product has to be tested.

The main testing process in our project is the identification of correct landmark in the query image. To test the accuracy, we give the model 5 edge case images of each landmark. An edge image can contain a lot noise like people, low lighting conditions, partial occlusion. We have noted the number of inliers found between the query image and the matched image.

6.2 Testing Objectives

The main objectives of testing are as follows:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has high probability of finding undiscovered error.
- A successful test is one that uncovers the undiscovered error.

6.3 Levels of Testing

Different levels of testing process are used in the testing process. Each level of testing aims to test different aspects of the system. The basic levels are unit testing. functional testing, system testing, performance testing, integration testing, and acceptance testing.

6.3.1 Unit Testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly and that program input produces valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application it is done after the completion of an individual unit before integration. This is a structural testing that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.3.2 Functional Testing

Functional testing provides systematic demonstrations that functions tested are available as specified by the business and technical requirements system documentation and user manuals.

Functional testing is centered on the following items.

Valid Input	Identified classes of valid input must be accepted.
Invalid Input	Identified classes of invalid input must be rejected.
Functions	Identified functions must be exercised.
Output	Identified classes of application outputs must be exercised.
System/Procedures	Interfacing systems or procedures must be invoked

6.3.3 System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results.

6.3.4 Performance Testing

The performance testing ensures that the output can be produced within the time limits and the time taken by the system for compiling giving response to the users and request being sent to the system for it to retrieve the results.

6.3.5 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

6.3.6 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

6.4 Test Cases

The test cases provided here test the most important features of the project.

6.4.1 Test cases for the project

Table 6.1 Test cases

Sl. No	Test Data	Expected Results	Observed Results	Remarks
1	Image input “Mount Fuji”	Outputs the name Mount Fuji and place Japan	Mount Fuji is recognized from the query image. Outputs the name Mount Fuji and place Japan	Pass
2	Image input “Eiffel Tower”	Outputs the name Eiffel Tower and place Paris	Eiffel Tower is recognized from the query image. Outputs the name Eiffel Tower and place Paris	Pass
3	Image input “Louvre Museum”	Outputs the name Louvre Museum and place Paris	Louvre Museum is recognized from the query image. Outputs the name Louvre Museum and place Pairs	Pass
4	Blurred Image of “Arc de Triomphe”	Outputs the name Arc de Triomphe and place Paris	India Gate is recognized from the query image. Outputs the name India Gate and place India.	Fail

Chapter 7

RESULTS

This section describes the screens of the “Landmark Recognition using Convolutional Neural Networks”. The snapshots are shown below.

Snapshot 1: Comparison between query image of Mount Fuji and dataset image of Mount Fuji

In the figure 4.6 landmark considered is Mount Fuji and feature matching is done between the same landmark which results in 24 inliers. The number of inliers is higher than other matches and is accurate for practical use.

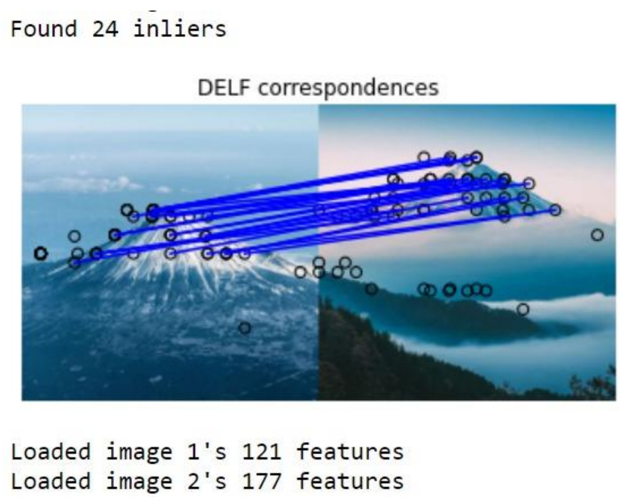

Figure 4.6: Snapshot of Mount Fuji being compared to itself

Snapshot 2: Comparison between query image of Eiffel Tower and dataset image of Eiffel Tower

In the figure 4.7 landmark considered is Eiffel Tower and feature matching is done between the same landmark which results in 38 inliers. The number of inliers is higher than other matches and is accurate for practical use.

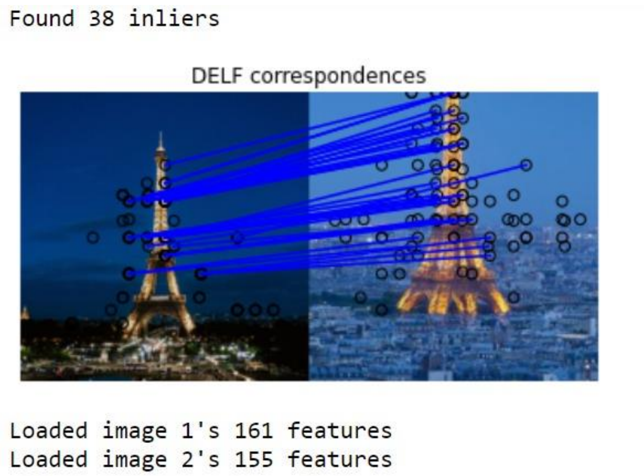

Figure 4.7: Snapshot of Eiffel Tower being compared to itself

Snapshot 3: Comparison between query image of Louvre Museum and dataset image of Louvre Museum

In the figure 4.8 landmark considered is Louvre Museum and feature matching is done between the same landmark which results in 65 inliers. The number of inliers is higher than other matches and is accurate for practical use.

Figure 4.8: Snapshot of Louvre Museum being compared to itself

Chapter 8

CONCLUSION

8.1 Major contributions

Large-scale recognition is the main concern and our project deals with the variable database size and can be highly scalable. Also, the objects that we are detecting are at an instance level so, identifying features that defines the instances is the key. Present datasets are huge which makes it difficult to be scaled, therefore require high computational resources and time whereas the custom-made dataset performs well while having vast variety and keeping all the other factors to a minimum. Simple CNN architectures are not built explicitly for capturing important features of landmarks and they produce very limited accuracy in reality. So, we test the dataset on a simple approach of identifying feature vectors of the images and using it to perform a similarity search to fasten the retrieving process. The result is obtained from a two-phase process, recognition and retrieval. Recognition is done using Deep Neural Network with CNN layers and trying to extract features of an image. For retrieving information from a large-scale dataset, we are comparing feature vector of query image with all the nearest neighbor feature vectors from the dataset.

8.2 Future Enhancements

The project has a very vast scope in future. The project can be implemented in any local system which runs fast and can be used in offline mode. Project can be updated in near future as and when requirement for more landmark arises, as it is very flexible in terms of scalability. This project can be implemented as an image search system with a larger dataset. It can be deployed as an application where any curious user can upload an image with an unknown landmark and get the information regarding it, which is much faster and more user-friendly.

BIBLIOGRAPHY

[1] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, Bohyung Han. “A Large-Scale Image Retrieval With Attentive Deep Local Features”. IEEE, 2017, pp. 3456-3465

[2] Ross Girshick. “Fast R-CNN”. IEE, 2015, pp. 1440 - 1448

[3] Tobias Weyand, André Araujo, Bingyi Cao, Jack Sim. “A Large-Scale Benchmark for Instance-Level Recognition and Retrieval”. IEEE, 2020, pp. 2575-2584

[4] Christof Henkel, Philipp Singer. “Supporting Large-scale Image Recognition with out-of-domain Samples”. eprint arXiv:2010.01650, 2020

[5] SeungKee Jeon. “1st Place Solution to Google Landmark Retrieval 2020”. eprint arXiv:2009.05132, 2020

APPENDIX

<u>INNOCENTRIC – 2021, VIRTUAL INTER-COLLEGE PROJECT COMPETITION</u>

Innocentric 2021

Global Academy of Technology

(An Autonomous Institute under VTU, Belagavi, NAAC Accredited with 'A' Grade)
Ideal Homes Township, Rajarajeshwari Nagar, Bengaluru-98

Department of Computer Science and Engineering
Accredited by NBA 2019-2022

CERTIFICATE OF PARTICIPATION

This is to certify that **Mr. C P Yashwanth** of **Global academy of technology** has participated and presented a Project titled **Landmark recognition using Convolutional neural networks** in **INNOCENTRIC 2021, Virtual Inter-Collegiate Project Competition** organized by the Department of Computer Science and Engineering at Global Academy of Technology, Bengaluru on 23rd July 2021.

Dr. Kavitha K S	**Dr. Bhagyashri R Hanji**	**Dr. N Ranapratap Reddy**
Convener	Professor & Head, CSE	Principal

Certificate ID : GAT-PRJ-000010
23-7-2021

Innocentric 2021

Global Academy of Technology

(An Autonomous Institute under VTU, Belagavi, NAAC Accredited with 'A' Grade)
Ideal Homes Township, Rajarajeshwari Nagar, Bengaluru-98

Department of Computer Science and Engineering
Accredited by NBA 2019-2022

CERTIFICATE OF PARTICIPATION

This is to certify that **Ms. Maddali Sowmya** of **Global Academy of Technology** has participated and presented a Project titled **Landmark Recognition using Convolutional Neural Networks** in **INNOCENTRIC 2021, Virtual Inter-Collegiate Project Competition** organized by the Department of Computer Science and Engineering at Global Academy of Technology, Bengaluru on 23rd July 2021.

Dr. Kavitha K S	**Dr. Bhagyashri R Hanji**	**Dr. N Ranapratap Reddy**
Convener	Professor & Head, CSE	Principal

Certificate ID : GAT-PRJ-000005
23-7-2021

Innocentric 2021

Global Academy of Technology

(An Autonomous Institute under VTU, Belagavi, NAAC Accredited with 'A' Grade)
Ideal Homes Township, Rajarajeshwari Nagar, Bengaluru-98

Department of Computer Science and Engineering
Accredited by NBA 2019-2022

CERTIFICATE OF PARTICIPATION

This is to certify that **Ms. Rakshitha Murthy** of **Global Academy of Technology** has participated and presented a Project titled **Landmark Recognition using conventional neural network.** in **INNOCENTRIC 2021, Virtual Inter-Collegiate Project Competition** organized by the Department of Computer Science and Engineering at Global Academy of Technology, Bengaluru on 23rd July 2021.

Dr. Kavitha K S Convener	**Dr. Bhagyashri R Hanji** Professor & Head, CSE	**Dr. N Ranapratap Reddy** Principal

Certificate ID : GAT-PRJ-000003
23-7-2021

Innocentric 2021

Global Academy of Technology

(An Autonomous Institute under VTU, Belagavi, NAAC Accredited with 'A' Grade)
Ideal Homes Township, Rajarajeshwari Nagar, Bengaluru-98

Department of Computer Science and Engineering
Accredited by NBA 2019-2022

CERTIFICATE OF PARTICIPATION

This is to certify that **Mr. Swaraj Parida** of **Global Academy of Technology** has participated and presented a Project titled **Landmark Recognition using Convolutional Neural Networks** in **INNOCENTRIC 2021, Virtual Inter-Collegiate Project Competition** organized by the Department of Computer Science and Engineering at Global Academy of Technology, Bengaluru on 23rd July 2021.

Dr. Kavitha K S Convener	**Dr. Bhagyashri R Hanji** Professor & Head, CSE	**Dr. N Ranapratap Reddy** Principal

Certificate ID : GAT-PRJ-000055
23-7-2021