Lost and Found Image Retrieval

(Project type 2)

Submitted in partial fulfilment of the requirements for the degree of

Bachelor of Technology

In

Computer Science

by

Nikhil Kumar Parashar (19BCE0076)

Anmol Bansal (19BCE0630)

Under the guidance of

Prof. Naveen Kumar N

SCOPE VIT University, Vellore

June 2021

<u>DECLARATION</u>

I hereby declare that the thesis entitled **"Lost and Found Image Retrieval"** submitted by me, for the award of the degree of Bachelor of Technology in Computer Science to VIT is a record of bonafide work carried out by me under the supervision of **Prof Naveen Kumar N**.

I further declare that the work reported in this thesis has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Vellore
Date: 7 June 2021

Signature of the Candidate

CERTIFICATE

This is to certify that the thesis entitled **"Lost and Found Image Retrieval"** submitted by **Anmol Bansal, 19BCE0630 and Nikhil Kumar Parashar, 19BCE0076, SCOPE, VIT**, for the award of the degree of Bachelor of Technology in Computer Science, is a record of bonafide work carried out by him under my supervision during the Winter 2020-21 semester, as per the VIT code of academic and research ethics.

The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university. The thesis fulfils the requirements and regulations of the University and in my opinion, meets the necessary standards for submission.

Place: Vellore
Date: 7 June 2021

Signature of the Guide

<u>ABSTRACT</u>

Suppose you and your family went on a vacation and stayed at a hotel. You had a great time but when you returned you realized you left behind a belonging in the room. You go back but you don't find it because they didn't have a lost and found section, or if they had you had to go through too much hassle looking for your item in that mismanaged mess with no guarantee of recovering it. So much more time spent and energy put, but with zero output.

Hotel management never takes Lost and Found management seriously. But people who lost it might know the importance of that object. The loss of an important belonging can take a heavy toll on your mind. Sometimes people don't even go back if they have lost an item because they don't want that much hassle. The question that arises here is - Is there anything we can do to solve this problem? So, people can easily retrieve their lost belongings?

Our project aims to solve the same problem. So, you don't have to go through so much trouble to retrieve the lost item. Our project uses Image Retrieval in a way that hasn't been implemented yet, bringing in a new application in the field of Content-Based Image and Video Retrieval.

CONTENTS

1. INTRODUCTION

1.1 Objective

Losing possession in hotel or resort premises is a serious concern for many. The management also has to take care of these items and ensure that it is retrieved by their actual owner only. Our project aims to solve this hassle and pressure a person deals with to retrieve a lost object. The object can be lost anywhere, whether it be a hotel, hospital, restaurant, school, etc. For this project, our application focuses on the retrieval of lost objects from hotels premises. Our main motive is for better management of these lost items and to ease the process of finding a lost item.

1.2 Motivation

The expense of things or the items one possesses is increasing over the years. Mobile phones, watches, bags are really expensive in the present times. Losing these expensive possessions can be disheartening and even result in privacy breach in some particular cases. Hotels and Resorts are a place where a person gets comfortable as if at home and it is most likely that an individual may lose his/her belongings there.

1.3 Background

Different hotel administrations have different ways of keeping track of the lost items and a common way was to protect those items in a storeroom until the owner comes asking for the item. Creating an online database through images and letting the users access the database from a similar image of the lost item is a great way of easing the process. Image retrieval was the main point of focus here. Choosing the best image retrieval method that is widely used and gives accurate result was necessary. After some thorough research, we found that VGG16 is a model that will help us give the most accurate results. Also, the web was a great medium to bring this project to life and create an interface for the users and hotel management. Thus, we made a portal to access the model with ease.

For this project we have used a model VGG 16 provided by Keras, it is 16 layers deep convolutional neural network. The model achieves 92.7% top 5 test accuracy in ImageNet which is a dataset of over 14 million images belonging to several 1000 classes. The default input size of the image for the VGG16 model is 224x 224 px with

3 channels for RGB Image.

Since images are unstructured data and we can't directly train our model over those images. We converted these images into structured data and extracted their features. For that, we converted images into NumPy arrays and performed Deep Feature Synthesis. Deep feature synthesis stacks multiple transformation and aggregation operations to create features from data spread across many tables. Like most ideas in machine learning, it's a complex method built on a foundation of simple concepts.

We made use of various modules for the making of our project like Keras and TensorFlow for building our deep learning models, Matplotlib and pillow for image preprocessing and importing them, NumPy for all the calculations and preprocessing and Flask to connect the model to our web application.

2. Related Works

There hasn't been much development for Lost and Found Management in the field of Computer Science. Existing systems of Hotel Management barely include any mention of Lost and Found Management. If it does, then it is to maintain a register with handwritten details. Moreover, some of the existing systems in the world, focus mainly on personal lost items in a city and a multiplatform mobile application for reporting and retrieval of lost objects. The main concern over here is the wide database and unorganized manner of reporting lost objects which can lead to fake values in the model.

3. Proposed System

3.1 Drawbacks of the Existing System

Existing methods are subjected to carelessness and vagueness. Any objects found are kept in the very mismanaged storeroom so-called "Lost and Found" with no promise of actually finding the lost item. Also, there is no process for the verification if the item belongs to the retriever. With such weak management, how can one easily retrieve his/her lost belongings?

3.2 Our Solution

Current Hotel Management ignores the Lost and Found management. Even if they have

a lost and found section it is not managed well enough. So, when you lose your belonging, it gets tough to get them back. Every hotel can manage the lost item easily. All they have to do is upload the image of the retrieved lost item to their database through our web app and when the person returns looking for their belonging, they just have to provide an image of the item and they can easily find the item. The model searches for similar images in the database and shows them to the user. The location where that item is stored by the hotel management is also shown for quick retrieval.

3.3 Sample Code

SaveFeatures.py

```python
from PIL import Image
from DeepFeatures import DeepFeatures
from pathlib import Path
import numpy as np

if __name__ == '__main__':
    df = DeepFeatures()

    for ipath in sorted(Path("./static/image").glob("*.jpg")):
        feature = df.extract(img = Image.open(ipath))
        fpath = Path("./static/feature")/(ipath.stem + ".npy")
        np.save(fpath, feature)
```

DeepFeatures.py

```python
from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.vgg16 import VGG16, preprocess_input
from tensorflow.keras.models import Model
import numpy as np

class DeepFeatures:
    def __init__(self):
        base = VGG16(weights = 'imagenet')
        self.model = Model(inputs = base.input, outputs =
base.get_layer('fc1').output)

    def extract(self, img):
        img = img.resize((224,224))
        img = img.convert('RGB')
        x = image.img_to_array(img)
        x = np.expand_dims(x, axis = 0)
        x = preprocess_input(x)
```

```
        feature = self.model.predict(x)[0]
        return feature/np.linalg.norm(feature)
```

Server.py

```python
import numpy as np
from PIL import Image
from DeepFeatures import DeepFeatures
from datetime import datetime
from flask import Flask, request, render_template
from pathlib import Path

app = Flask(__name__)

df = DeepFeatures()

@app.route('/', methods=['GET', 'POST'])
def index():
    if request.method == 'POST':
        features = []
        img_paths = []
        for feature_path in Path("./static/feature").glob("*.npy"):
            features.append(np.load(feature_path))
            img_paths.append(Path("./static/image") /
(feature_path.stem + ".jpg"))
        features = np.array(features)
        file = request.files['query_img']
        img = Image.open(file.stream)
        uploaded_img_path = "static/uploaded/" +
datetime.now().isoformat().replace(":", ".") + "_" + file.filename
        img.save(uploaded_img_path)
        query = df.extract(img)
        dists = np.linalg.norm(features-query, axis=1)
        ids = np.argsort(dists)[:5]
        scores = [(img_paths[id].stem, img_paths[id]) for id in ids]

        return render_template('index.html',
query_path=uploaded_img_path, scores=scores)
    else:
        return render_template('index.html')

@app.route('/upload', methods=['GET','POST'])
def upload():
    if request.method == 'POST':
        file = request.files['query_img']
        img = Image.open(file.stream)
        room = request.form['room']
        store = request.form['store']
        rack = request.form['rack']
        name = room + '-' + store + '-' + rack
```

```
        uploaded_img_path = "static/image/" + name + ".jpg"
        img.save(uploaded_img_path)

        feature = df.extract(img)
        fpath = Path("./static/feature")/(name + ".npy")
        np.save(fpath, feature)

        return render_template('upload.html')
    else:
        return render_template('upload.html')

if __name__=="__main__":
    app.run(host = "127.0.0.1", port = 8000)
```

4. Result Analysis and Screenshots

We used a custom dataset of objects over 10 classes that are most likely to be lost. Every time we searched for an object the model found the object and viewed it as the most relevant image in the top -5 search. On average, the top - 3 images retrieved were found to be relevant to the object being found.
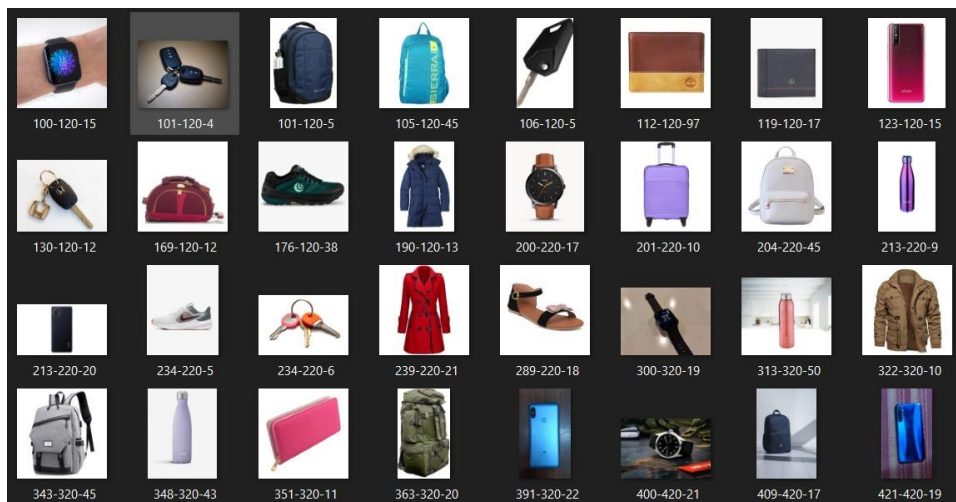
Figure 1 Custom Dataset
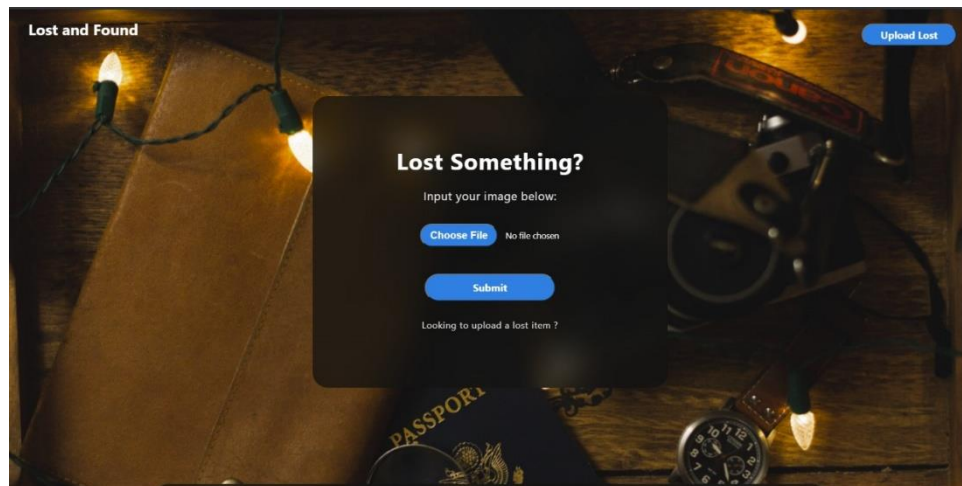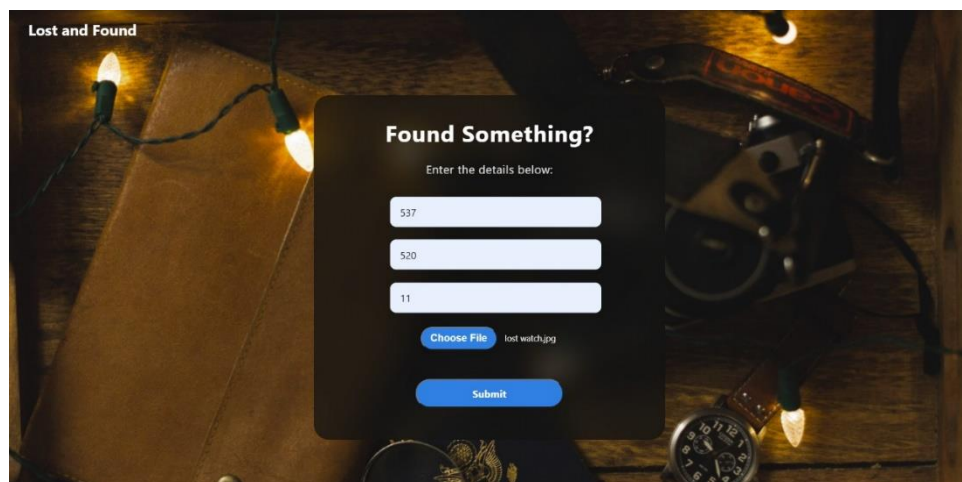
Figure 2 Query Image Search
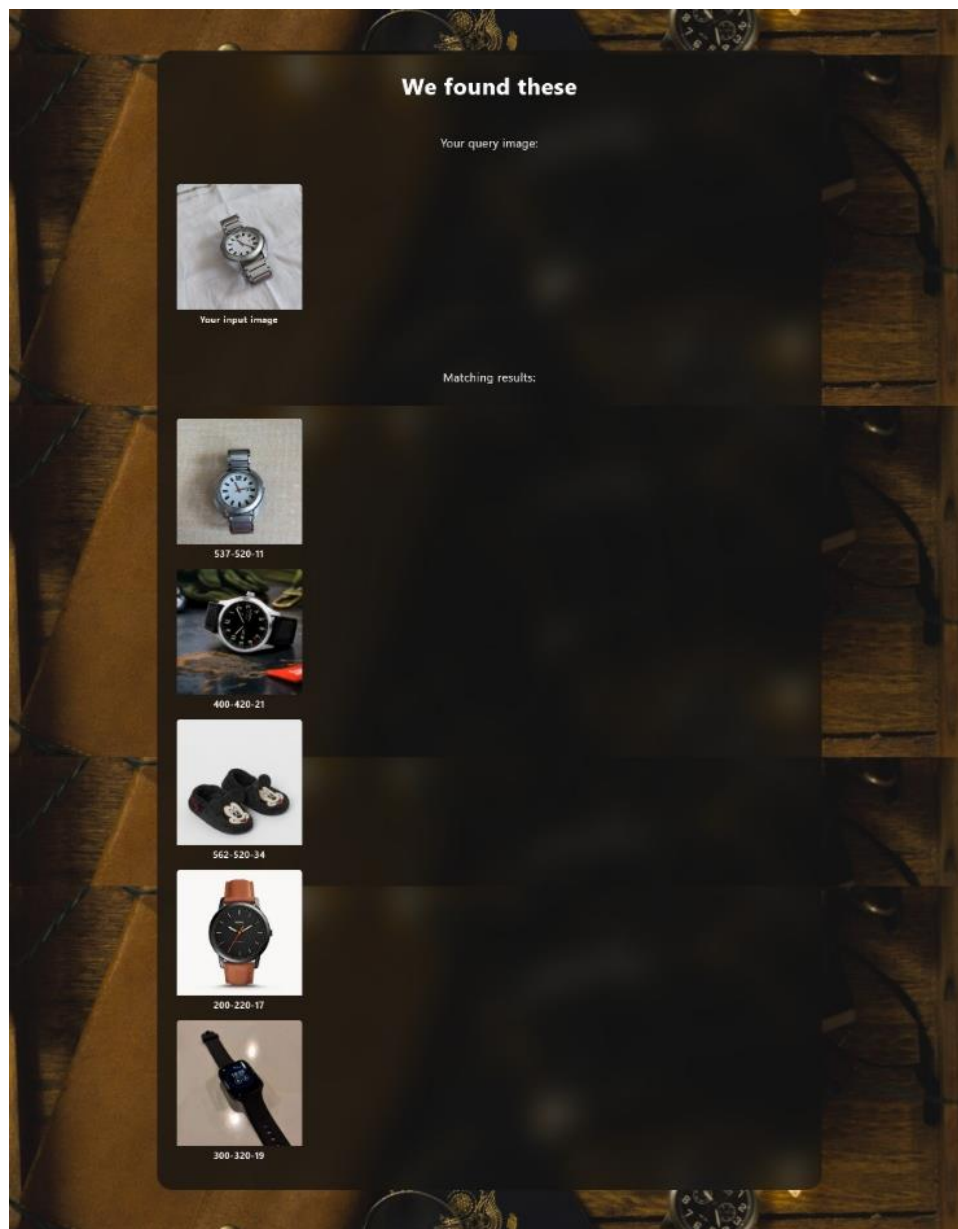

Figure 3 Upload Lost Item
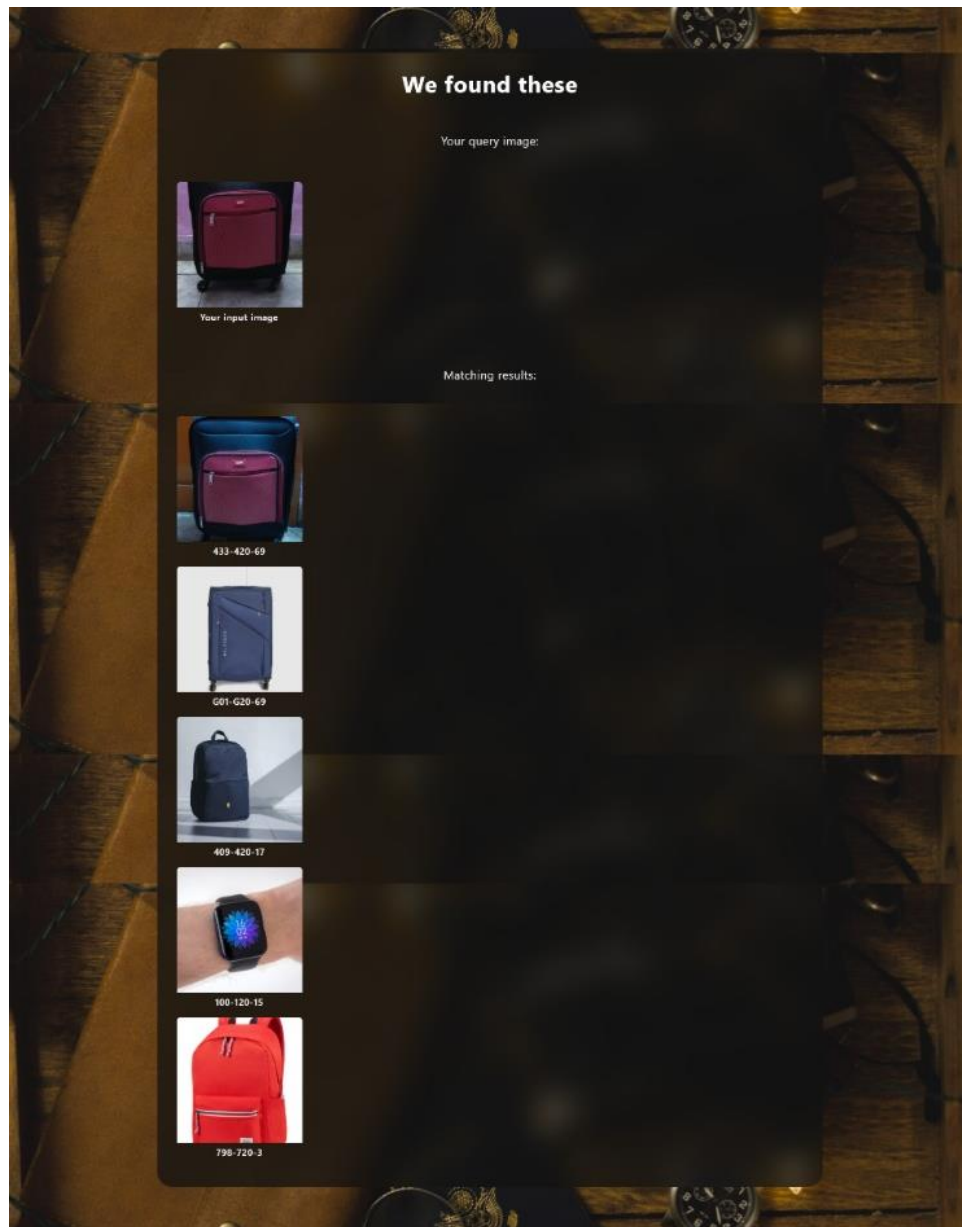
Figure 4 Search Result 1

Figure 5 Search Result 2

5. Conclusion

Given the results and analysis, we conclude that the implementation of our web app into hotels can have a really strong impact on the betterment of lost and found management. People can easily find the lost items, and hotel management won't have to go through the hassle of finding the object from a mess that used to be. Any item, however small it may be, can be found.

6. References

Online References:

1. https://flask.palletsprojects.com/en/2.0.x/
2. https://neurohive.io/en/popular-networks/vgg16/
3. https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c
4. https://www.geeksforgeeks.org/vgg-16-cnn-model/
5. https://matplotlib.org/stable/contents.html
6. https://pypi.org/project/numpy/

Journal References:

1. https://www.researchgate.net/publication/330162417_Search_by_Image_New_Search_Engine_Service_Model
2. https://arxiv.org/abs/1409.1556
3. https://arxiv.org/pdf/1409.1556.pdf,2015.pdf
4. https://www.researchgate.net/publication/337105858_Transfer_learning_using_VGG-16_with_Deep_Convolutional_Neural_Network_for_Classifying_Images
5. https://www.engpaper.com/image-retrieval-2018.htm
6. https://www.researchgate.net/publication/257706512_Content-Based_Image_Retrieval_Research
7. https://www.researchgate.net/publication/274024782_Image_Query_Based_Search_Engine_Using_Image_Content_Retrieval