Robust Landmark Recognition with Out-of-Distribution Detection using Deep Features and K-Nearest Neighbors

Lucas F. Pereira[1], Pedro H. N. Castro[2], Valéria de C. Santos[1], André L. C. Ottoni[1],
Gladston J. P. Moreira[1] and Eduardo J. S. Luz[1]
[1]*Computing Department, Federal University of Ouro Preto*, Ouro Preto, Brazil, 35402-136
[2]*Postgraduate Program in Computer Science, Federal University of Ouro Preto*, Ouro Preto, Brazil, 35402-136

***Abstract*—Recognizing landmarks from diverse, real-world images is challenging due to variations in viewpoint, illumination, and occlusion. A critical, often overlooked, aspect is the ability to reject images that do not depict any known landmark (Out-of-Distribution, OOD, samples). This paper proposes a robust system for landmark recognition that integrates a deep convolutional neural network (CNN) for feature extraction and classification with a k-Nearest Neighbors (KNN) based approach for OOD detection. We leverage transfer learning with a DenseNet-201 architecture, fine-tuned on a diverse landmark dataset. The proposed system achieves 97.5% classification accuracy on In-Distribution (ID) landmark images on the Visual Chine Dataset. Our KNN-based OOD detection method, using features from the trained DenseNet, achieves a 97.0% True Positive Rate for ID samples at a 95.0% recall threshold (TPR@95TPR), effectively measuring precision at high ID recall against OOD samples, demonstrating its efficacy in distinguishing landmarks from irrelevant scenes. This combined approach offers a practical solution for real-world tourist applications.**

***Index Terms*—Landmark Recognition, Deep Learning, Convolutional Neural Networks, Out-of-Distribution Detection, k-Nearest Neighbors, Transfer Learning, Tourism Applications.**

I. Introduction

The global tourism industry is a significant economic driver, with a projected market volume expected to reach $1.114 trillion by 2029 [1]. Advances in Deep Learning (DL) have spurred innovations to enhance tourist experiences. For instance, DL models have been used for tourist demand forecasting [2] and analyzing factors influencing visitor volume [3]. This work focuses on developing a system that identifies tourist attractions from images captured by smartphones, enabling users to quickly obtain information about a landmark by simply photographing it.

Recognizing landmarks is non-trivial; images can vary significantly due to lighting conditions, viewing angles, partial occlusions, and seasonal changes [4]. A crucial challenge arises when a system must not only classify known landmarks but also identify and reject images that do not belong to any pre-defined landmark category, a problem known as Out-of-Distribution (OOD) detection. Modern Convolutional Neural Networks (CNNs) achieves impressive performance on In-Distribution (ID) data (i.e., data similar to the training distribution) [5], [6]. However, they often produce confident but incorrect predictions for OOD inputs [7]–[10]. This is problematic for real-world applications where input control is limited [8]. While some prior work has addressed landmark recognition [4], robust OOD detection has often been overlooked, and the potential of deeper, pre-trained CNNs has not always been fully exploited.

To address these challenges, this paper proposes a two-stage system:

1) A powerful landmark classifier based on the DenseNet-201 architecture [11], leveraging transfer learning from ImageNet [12] and fine-tuning.
2) Along with an effective OOD detection mechanism using the k-Nearest Neighbors (KNN) algorithm operating on the deep features extracted by the classifier. This combination aims to provide both accurate landmark identification and reliable rejection of irrelevant scenes.

Our primary research questions are:

1) (RQ1) Can a fine-tuned DenseNet-201 architecture achieve high accuracy in classifying diverse landmark images?
2) (RQ2) Can a KNN-based OOD detector, utilizing deep features from the trained classifier, effectively distinguish between ID landmark images and OOD non-landmark images?
3) (RQ3) How do different KNN parameters (number of neighbors k, distance metric) affect OOD detection performance in this context?

We conduct experiments using the ”Visual China” landmark dataset [13] for ID classification and the Places365 dataset [14] as a source of OOD samples. Our system demonstrates strong performance, achieving 97.5% accuracy on the landmark classification task and 97.0% precision for ID classification (at 95.0% True Positive Rate (TPR) for ID samples) in the OOD detection task, validating our approach.

The main contributions of this work are:

- A robust landmark recognition system integrating a fine-tuned DenseNet-201 classifier with a KNN-based OOD detector.

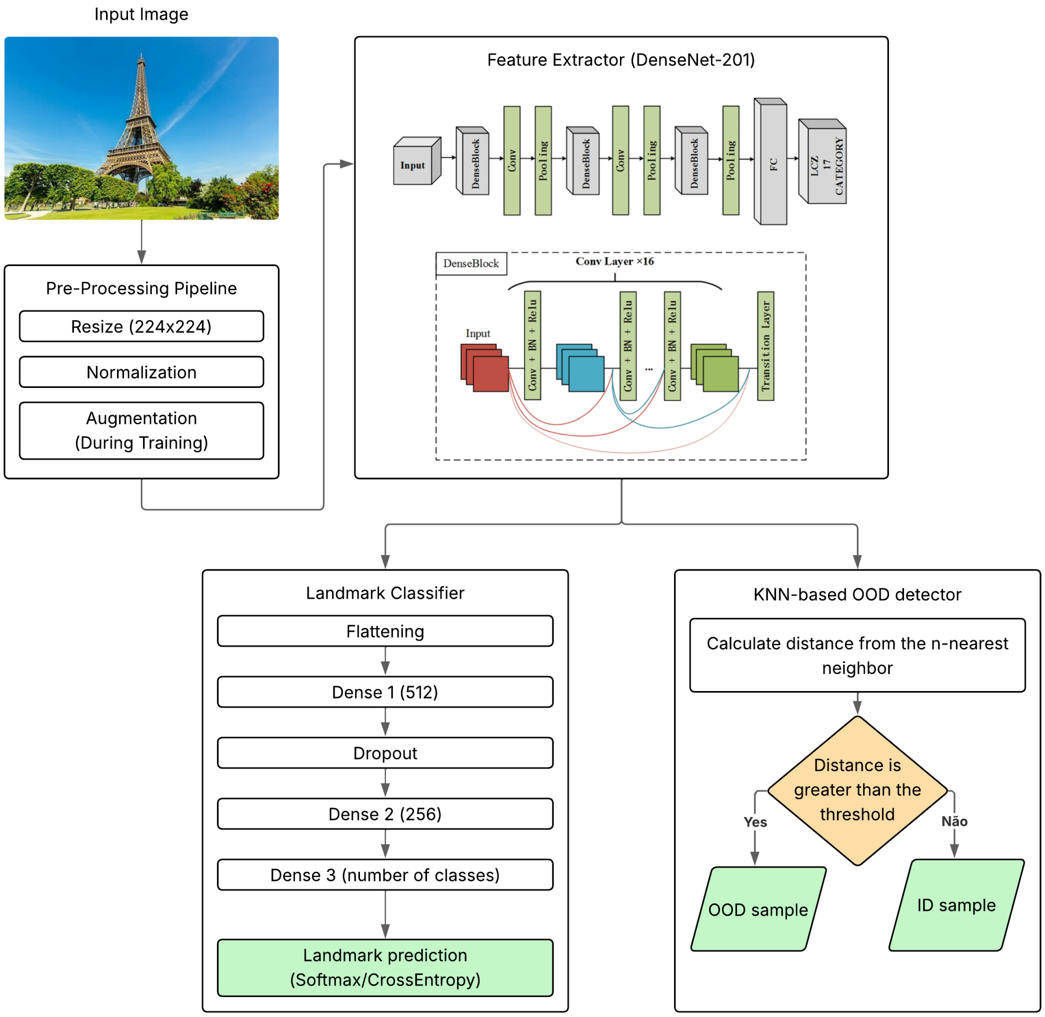

Fig. 1: Overview of the proposed landmark recognition and OOD detection system. An input image is processed by a pre-processing pipeline (resize, normalization, and data augmentation during training). The core is a DenseNet-201 based feature extractor. These features are used by a classifier for landmark identification and by a KNN-based module for OOD detection to reject non-landmark images.

- A comprehensive evaluation of the KNN-based OOD detection method on deep features for the landmark recognition task, including an analysis of different k values and distance metrics (Euclidean vs. Cosine).
- Demonstration of high classification accuracy and effective OOD rejection on a challenging real-world landmark dataset and a diverse OOD set.
- An open problem setting (Landmark recognition with OOD) and a baseline for future research.

The remainder of this paper is organized as follows: Section II discusses related literature. Section III introduces the necessary background concepts and formal problem setting. Section IV details our proposed methodology. Section V describes the experimental setup and Section VI presents and discusses the results. Finally, Section VII concludes the paper and outlines future work.

II. Related Work

This work builds upon research in landmark recognition, DL for image classification, and OOD detection.

A. Landmark Recognition

Landmark recognition, also known as tourist spot recognition, consists of identifying landmarks from input data that encodes their respective information. In this case, the aim is to perform this identification using images that refer to a set of known landmarks.

In the work of 15, a set of Machine Learning (ML) algorithms was applied to recognize images of tourist attractions in the city of Bangladesh. For this purpose, 846 images photographed by tourists' smartphones were collected, comprising

5 famous places in the city. Then, the number of images was increased with modified copies through methods such as rotation, approximation, zooming out and Gaussian noise. Having this new database, a comparison was made between the use of CNN, *Support Vector Machine*, *Long Short-Term Memory* (LSTM), *K-Nearest Neighbor* (KNN) and *Recurrent Neural Network* (RNN). After a series of experiments, the authors noted that the use of a CNN enabled a better result, whose accuracy reached 97.0%.

In [16], in turn, the authors used an object detection technique based on deep learning to detect tourist attractions in real time through cell phone cameras. The authors proposed the use of the YOLO (You only look once) network, a deep CNN capable of detecting multiple objects in a scene and their respective locations. After carrying out experiments with 28 tourist attractions in the city of Hsinchu, Taiwan, it was possible to conclude that YOLO excels in accuracy and speed when compared to the *Faster region-convolutional neural networks* and *Single-Shot Multibox Detector* methods for object detection.

B. Out-of-Distribution Detection

OOD detection aims to decide whether a sample belongs to a distribution of interest. To this end, several methods have been proposed. Among them, the following methods stand out.

1) classifier-based uncertainty methods: This strategy aims to identify OOD samples based on the degree of uncertainty in the predictive model. The authors of 8, for example, developed Out-of-DIstribution detector for Neural networks (ODIN): an OOD detection method focused on images whose principle is the use of temperature scaling on the softmax function and perturbations in the input data. With this approach, the authors demonstrated that the distribution of the outputs of the softmax function differs for data inside and outside the training distribution in multiclass classification tasks. Thus, the method consists of generating a perturbation in the input image, propagating the image through the neural network with a softmax activation function with Temperature Scaling, and comparing the maximum probability with a threshold. In this work, the threshold is chosen so that the Recall metric for the database within the distribution is 95.0%. Despite the satisfactory results of the experiments, in order to regulate the disturbance factor ϵ and temperature T, it is necessary to use a database outside the distribution. As discussed in [17], defining a representative OOD database is a challenging task, since a model adjusted from it may not generalize to other examples outside the distribution. In fact, an OOD sample is any image that deviates from the standard classes used in the model training, which encompasses a vast amount of possible features.

2) distance-based methods: A distance-based method for OOD detection relies on computing a distance metric (e.g., Euclidean, Mahalanobis) between the target sample and the ID feature representations (such as class means or training embeddings). If the distance exceeds a predefined threshold, the sample is considered OOD. These methods assume that OOD inputs deviate significantly from the feature space learned from ID data. In 18, the author presents a demonstration of the effectiveness of KNN in OOD detection. According to the author, one of the advantages of the method is that it is nonparametric, that is, it does not require an initial assumption about the data distribution, which makes it generic and flexible. The OOD technique used consists of first extracting the feature map of each image in the training set, using a pre-trained neural network, in this case with ImageNet-1k weights. Then, the same extractor is used to obtain the feature map of the target sample in order to calculate the Euclidean distance in relation to the ID samples. The distance to be evaluated is that referring to the k-th closest neighbor. The metric used to define the threshold considers the value that correctly classifies at least 95.0% of the training data in order to evaluate the TPR95 metric. The use of distance-based methods (e.g., KNN), unlike ODIN, has the advantage of being simple and not requiring an OOD database for training. However, a real application could have high memory costs, since it would be necessary to store the embedding of each training example for each of the possible image classes. Considering a problem involving few classes combined with the current computational power of storage and processing, the KNN method was explored in this work due to its effectiveness and simplicity.

3) reconstruction-based methods: A reconstruction-based method for OOD detection leverages a model (typically an autoencoder or similar architecture) trained to reconstruct ID data. At test time, the model attempts to reconstruct the input, and the reconstruction error (e.g., mean squared error between input and output) is measured. If the error is significantly high, the sample is considered OOD. The underlying assumption is that the model will reconstruct ID samples well but fail to accurately reconstruct OOD samples due to their unfamiliar patterns. Regarding this strategy, 19 created a framework for solving OOD detection problems using reconstruction techniques. According to the authors, this approach enables the effective solution of general OOD problems, as it allows the model to learn intrinsic information about the data distribution. Furthermore, they justify that classifying neural networks often does not learn the essential characteristics of each class in the training set, since learning occurs based on the patterns of distinction between these classes. In reconstruction techniques, the model is, in turn, forced to extract the content of the images in the most representative way possible. Therefore, the work consisted of using masked image modeling as a pretext task.

III. Preliminaries and Problem Setting

This section provides the necessary background on the core techniques and formally defines the problem.

A. Convolutional Neural Networks

CNNs are a class of DL models that have revolutionized computer vision, a subarea of computing that studies how a computer can "visualize" and interpret visual data such as images and videos. These are a family of neural networks that use convolutional layers. These layers use sliding filters that favor the extraction of features from grid-structured data, especially images, without depending entirely on the spatial location of the elements of interest [20]. Due to this particularity, CNNs are widely used in computer vision tasks.

B. DenseNet

In this work, the DenseNet-201 network will be used, a CNN architecture proposed in 11. Instead of connecting each layer to the previous one, this architecture connects each layer to the previous ones through a concatenation of output deep features. According to 11, previous work has already demonstrated that shorter connections between layers close to the input and output of the network make training more efficient while allowing for an increase in its depth and accuracy, an observation that motivated the creation of this neural network structure. As in the network proposed in [6], DenseNet also uses a sequencing of blocks and creates short connections between them so that the output of a block is used to feed subsequent blocks, which encourages feature reuse and gradient flow. During the study, accuracy comparisons were made between DenseNet and other state-of-the-art architectures in image classification tasks. Thus, the authors concluded that the proposed architecture is promising and has superior results to ResNet within the ImageNet database, for example.

C. Transfer Learning

Transfer learning is a technique where a model pre-trained on a large dataset (e.g., ImageNet [12] for image classification) is adapted for a new, often smaller, target dataset [21]. According to 22, this strategy seeks to improve the results of learning in a given task in a specific problem domain by transferring the learning obtained in tasks in a different, but related, domain. First, a model pre-trained in a general-level task is used. Then, this model is adjusted to perform more specific tasks, starting the training process from the previously adjusted parameters.

D. Problem Setting: Landmark Classification and OOD Detection

Let $\mathcal{X}$ be the input image space and $\mathcal{Y}_{ID} = \{c_1, c_2, \ldots, c_M\}$ be the set of M known ID landmark classes. The landmark classification task is to learn a mapping $f : \mathcal{X} \to \mathcal{Y}_{ID}$ that assigns an input image $I \in \mathcal{X}$ to its correct landmark class $y \in \mathcal{Y}_{ID}$. We train a classifier f_θ parameterized by θ.

The OOD detection task is to determine if an input image I comes from the ID training distribution $P_{ID}(I)$ or an OOD distribution $P_{OOD}(I)$, where $P_{OOD}(I)$ represents images of scenes not belonging to any of the M known landmark classes. Let $\phi : \mathcal{X} \to \mathcal{Z}$ be a feature extractor that maps an input image I to a feature representation (embedding) $\mathbf{z} = \phi(I) \in \mathcal{Z} \subseteq \mathbb{R}^D$. The OOD detector is a function $g : \mathcal{Z} \to \{\text{ID, OOD}\}$ that, given an embedding $\mathbf{z}$, decides if the original image I is ID or OOD. Our goal is to develop a system that performs both f and g effectively.

E. K-Nearest Neighbors for OOD Detection

The KNN algorithm is a non-parametric method used for classification and regression [23]. For OOD detection, as proposed by the authors in [18], we leverage KNN in the feature space $\mathcal{Z}$. Given a gallery of feature vectors $\{\mathbf{z}_{ID}^{(i)}\}_{i=1}^{N_{ID}}$ extracted from ID training images, and a test image's feature vector $\mathbf{z}_{test}$, the OOD score $s(\mathbf{z}_{test})$ can be defined as the distance to its k-th nearest neighbor in the gallery:

$$s(\mathbf{z}_{test}) = \min_{k}\{\text{dist}(\mathbf{z}_{test}, \mathbf{z}_{ID}^{(j)}) \mid j \in \text{TopK}(\mathbf{z}_{test}, \{\mathbf{z}_{ID}^{(i)}\})\} \tag{1}$$

where $\text{dist}(\cdot, \cdot)$ is a distance metric (e.g., Euclidean or Cosine distance), and $\text{TopK}(\mathbf{z}_{test}, \{\mathbf{z}_{ID}^{(i)}\})$ returns the indices of the k nearest neighbors of $\mathbf{z}_{test}$ from the ID gallery. A larger distance $s(\mathbf{z}_{test})$ suggests that $\mathbf{z}_{test}$ is more likely to be OOD. A threshold τ is then applied to $s(\mathbf{z}_{test})$ to make the ID/OOD decision.

IV. Proposed Method

Our proposed system for landmark recognition with OOD detection consists of three main stages: data preprocessing, a CNN-based landmark classifier, and a KNN-based OOD detection module. An overview is shown in Figure 1.

A. Data Preprocessing and Augmentation

Input images are first resized to 224×224 pixels, the standard input size for DenseNet-201 [11]. Pixel values are normalized to the range $[0, 1]$. Labels are one-hot encoded. To increase dataset variability and prevent overfitting during training, we apply data augmentation techniques. Specifically, random horizontal flipping and random rotation (by a factor of up to 0.2, i.e., $\pm 36°$) are applied to the training images on-the-fly. This considers that photos taken by smartphones generally have slight rotations. According to 24, simple data augmentation techniques can significantly improve the generalization capacity of models, especially in scenarios with limited data sets, by simulating real variations that occur in images captured in natural environments.

B. Landmark Classifier Architecture

We employ the DenseNet-201 architecture, pre-trained on ImageNet, as our base feature extractor. The original classification head of DenseNet-201 is removed. We then append a new classification head consisting of:

1) A flattening layer applied to the output of the DenseNet-201 base.
2) A fully connected (Dense) layer with Rectified Linear Unit (ReLU) activation.
3) A Dropout layer with a rate of 20.0% to mitigate overfitting.
4) Another Dense layer with ReLU activation.
5) A final Dense output layer with M neurons (where M is the number of landmark classes) and Softmax activation to produce class probabilities.

The resulting model has approximately 66.6 million parameters, of which about 48.3 million (primarily in the later blocks of

DenseNet-201 and the new head) are made trainable during fine-tuning, while earlier layers are frozen. This specific architecture (number of dense layers and neurons) was determined through experimentation.

C. Classifier Training

The landmark classifier is trained using transfer learning. We initialize the DenseNet-201 base with weights pre-trained on ImageNet. We then fine-tune the last 12 layers of the DenseNet-201 base and the newly added classification head on our target landmark dataset. Earlier layers of DenseNet are kept frozen, as they typically learn more general, low-level features. Training is performed using the Adaptive Moment Estimation (ADAM) optimizer [25] with a learning rate of 0.001. We use a batch size of 64. The model is trained for a maximum of 100 epochs, with an early stopping criterion: training halts if the validation accuracy does not improve for 6 consecutive epochs. This typically led to convergence in approximately 30 epochs. The loss function is categorical cross-entropy.

D. OOD Detection using KNN on Deep Features

For OOD detection, we adopt the KNN-based approach described by [18].

1) **Feature Extraction:** We use the trained landmark classifier as a feature extractor ϕ. Specifically, we take the output of the flattening layer (after the final layer of DenseNet-201) as the feature representation $\mathbf{z} = \phi(I)$ for an input image I.
2) **ID Gallery Creation:** A gallery of feature vectors, $\mathcal{G}_{ID} = \{\mathbf{z}_{train}^{(i)} \mid I_{train}^{(i)}$ is an ID training image$\}$, is created by processing all images from the ID training set through ϕ.
3) **Distance Calculation:** For a test image I_{test} with feature vector $\mathbf{z}_{test}$, we calculate its distance to all feature vectors in $\mathcal{G}_{ID}$. We experiment with two distance metrics:
   - Euclidean distance: $d_E(\mathbf{a},\mathbf{b}) = \sqrt{\sum (a_j - b_j)^2}$
   - Cosine distance: $d_C(\mathbf{a},\mathbf{b}) = 1 - \frac{\mathbf{a}\cdot\mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|}$
4) **K-th Nearest Neighbor Distance:** The OOD score $s_k(\mathbf{z}_{test})$ is the distance to the k-th nearest neighbor of $\mathbf{z}_{test}$ in $\mathcal{G}_{ID}$. We evaluate $k \in [1, 15]$.
5) **Threshold Determination for TPR@95TPR Metric:** To evaluate performance using the TPR@95TPR metric (Precision at 95.0% TPR for ID samples), we first need to determine a distance threshold τ_k for each k and each distance metric. τ_k is chosen such that 95.0% of the ID validation images are correctly classified as ID (i.e., their $s_k(\mathbf{z})$ is less than or equal to τ_k). This establishes the 95.0% TPR operating point.
   a) For each ID validation sample, calculate its k-th nearest neighbor distance $s_k(\mathbf{z}_{val\_id})$ to the gallery $\mathcal{G}_{ID}$.
   b) Collect all such distances for the ID validation set.
   c) Sort these distances in ascending order.
   d) τ_k is the distance at the 95th percentile of these sorted distances.
6) **OOD Classification:** A test sample $\mathbf{z}_{test}$ is classified as ID if $s_k(\mathbf{z}_{test}) \leq \tau_k$, and OOD otherwise.

V. Experimental Setup

This section outlines the datasets, implementation specifics, and evaluation metrics used to assess our proposed system.

A. Datasets

1) In-Distribution (ID) Dataset: We use the ”Visual China” dataset [13] for training and evaluating the landmark classifier and as the ID set for OOD detection experiments. It contains 8412 images across 9 famous tourist landmarks. The classes are: Eiffel Tower, Great Wall, Temple of Heaven (Heaven), Huangguoshu Waterfall, Leifeng Pagoda, Potala Palace, Pyramids, Mount Qomolangma (Everest), and Three Pagodas. The dataset is reasonably balanced, with each class having between 873 and 975 images. Table I shows the per-class image count. We split this dataset into training (70.0%), validation (10.0%), and test (20.0%) sets. Example images are shown in Figure 2.

TABLE I: Class Distribution in the Visual China (ID) Dataset.

Class	**Total Images**
Eiffel	972
GreatWall	967
Heaven	931
Huangguoshu	900
Leifeng	948
Potala	975
Pyramid	935
Qomolangma	873
ThreePagoda	911
Total	**8412**

(a) Eiffel
(b) Great Wall
(c) Heaven
(d) Huangguoshu
(e) Leifeng
(f) Pyramid

Fig. 2: Example images from the Visual China (ID) dataset. Source: [13]

2) Out-of-Distribution (OOD) Dataset: For OOD detection evaluation, we use a subset of the Places365 dataset [14]. Specifically, we use its validation split, comprising 7298 images of diverse scenes and locations (e.g., bedrooms, forests, streets) that are distinct from the landmark categories in our ID dataset. This dataset serves as a challenging and realistic source of OOD samples. Example images are shown in Figure 3.

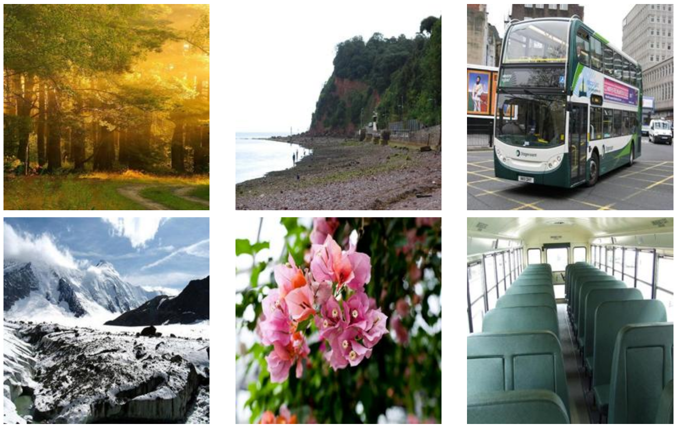

Fig. 3: Example images from the Places365 (OOD) dataset. Source: [14]

B. Implementation Details

Experiments were conducted on a machine equipped with an AMD Ryzen Threadripper 3960X CPU, 128GB RAM, and an NVIDIA RTX 3090 GPU (24GB GDDR6X). The software stack includes Python, with TensorFlow [26] (Keras API [27]) for deep learning model development and training, and Scikit-learn [28] for KNN implementation and metric calculations.

C. Evaluation Metrics

1) Landmark Classification: We evaluate the landmark classifier using standard metrics on the ID test set:

- **Accuracy:** Overall percentage of correctly classified images.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2}$$

- **Precision (per-class):**

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

- **Recall (per-class):**

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

- **Confusion Matrix:** To visualize class-wise performance and misclassifications.

2) OOD Detection: For OOD detection, we evaluate the ability to distinguish ID test images from OOD (Places365) images. We use a metric termed TPR95 to evaluate. This calculates the precision of classifying samples as ID when the threshold τ_k (from Section IV-D) is set to achieve a 95.0% TPR (Recall) for ID samples from the ID validation set. Effectively, this means:

1) Set threshold τ_k to correctly identify 95.0% of ID validation images as ID.
2) Using this τ_k, classify a mixed test set of (ID test images + OOD images).
3) The ”TPR95” metric reported is the Precision for the ID class on this mixed test set:

$$\text{Precision}_{\text{ID}} = \frac{\text{True ID images classified as ID}}{\text{All images classified as ID}} \tag{5}$$

This metric assesses how well the system can identify ID samples while maintaining a high ID detection rate. We refer to this as P@95TPR henceforth.

VI. Results and Discussion

This section presents the performance of our proposed system for landmark classification and OOD detection.

A. Landmark Classification Performance (RQ1)

The fine-tuned DenseNet-201 classifier achieved an overall accuracy of **97.5%** on the ID test set. Table II details the per-class precision and recall. The confusion matrix is shown in Figure 4.

TABLE II: Per-Class Precision and Recall on the ID Test Set.

Class	Precision	Recall
Qomolangma	0.995	0.980
GreatWall	0.954	0.956
Leifeng	0.974	0.937
Huangguoshu	1.000	1.000
Pyramid	0.985	0.978
ThreePagoda	0.984	1.000
Heaven	0.971	0.975
Potala	0.964	0.974
Eiffel	0.949	0.974

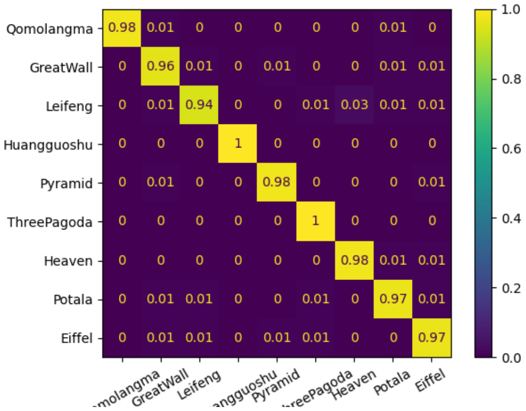

Fig. 4: Confusion matrix for landmark classification on the ID test set.

From the confusion matrix, it was noted that the model has greater ease in classifying some classes, such as Huangguoshu and ThreePagoda. This result can be understood from the lower variability of the examples in these images. The class Huangguoshu, referring to the largest waterfall in Asia, for example, has similar instances because they share common and recurring characteristics, such as the white color of the moving water, the blue and green tones of the river and the texture of the trees. These images lack significant variation in shapes and elements within their surroundings, as the angles for photographing the location are limited and the elements around the waterfall are almost always consistent, except for minor variations in lighting and tone related to the weather. This causes the model to extract the most important, recurring, and representative characteristics of what defines this tourist attraction.

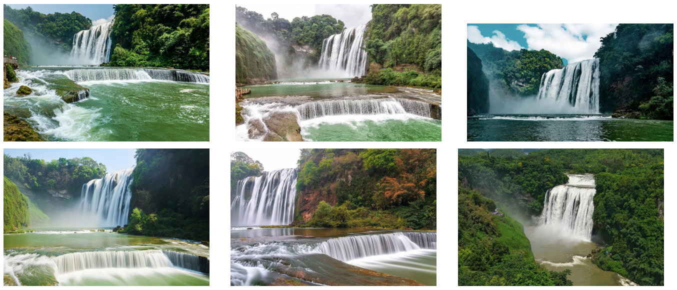

Fig. 5: Huangguoushu class samples

Although its images have representative characteristics for classification, the *Leifeng* class obtained inferior results. Built on the banks of a lake and surrounded by typical East Asian trees, the images referring to the *Leifeng* tower have greater variations, the main ones being: tones related to weather and time, angle, and visible portion. The variations in angle, in particular, add many extra elements to the images, such as buildings and other constructions. In addition, it is possible to notice that some photos were taken from a distance, thereby inserting even more spatial information into the database about the elements surrounding the tower. Such variations add complexity to the model, enabling it to extract representative and generic properties about this monument, which can lead it to consider these characteristics as determinants for classification during training.

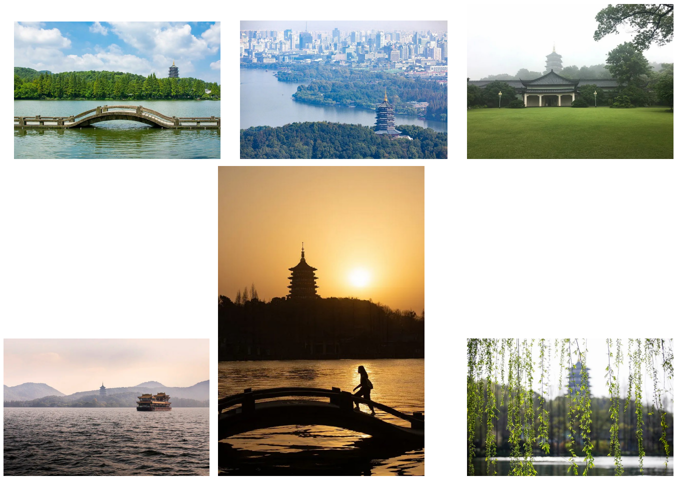

Fig. 6: Leifeng class samples

B. OOD Detection Performance (RQ2, RQ3)

We evaluated the KNN-based OOD detection module using the P@95TPR metric, varying the number of neighbors k and the distance metric (Euclidean vs. Cosine). The results are presented in Table III.

TABLE III: P@95TPR Results for OOD Detection using KNN with Euclidean and Cosine Distances for different values of k.

k	**P@95TPR**	
	Euclidean	**Cosine**
1	0.79	**0.97**
2	0.76	**0.97**
3	0.75	**0.97**
4	0.75	0.96
5	0.74	0.96
6	0.74	0.96
7	0.74	0.96
8	0.73	0.96
9	0.73	0.96
10	0.73	0.96
11	0.73	0.96
12	0.73	0.96
13	0.72	0.96
14	0.72	0.96
15	0.72	0.96

The Cosine distance consistently outperformed Euclidean distance across all values of k. The best P@95TPR of 97.0% was achieved with Cosine distance for $k = 1, 2,$ and 3. Unlike the Euclidean distance, the Cosine distance prioritizes the angle between vectors instead of their magnitude, that is, it focuses on the correlation between the values of the feature vector, which can intrinsically represent shapes, colors, and textures. Thus, the method is less susceptible to errors caused by the variation in the magnitude of the vectors. Such variation can easily occur when propagating ID images with features that are not relevant for classification, such as lighting and angle of the object of interest, through the network. Finally, it can be stated that the technique is effective and can be implemented as an OOD detection module in a classification system for landmarks.

VII. Conclusion

This paper presented a robust system for landmark recognition coupled with OOD detection, a critical component for real-world applicability. Our approach leverages a fine-tuned DenseNet-201 architecture for accurate landmark classification and a KNN-based method operating on its deep features for effective OOD rejection. Experimental results demonstrated high performance: 97.5% accuracy for landmark classification on the Visual China dataset, and a P@95TPR of 97.0% for OOD detection (distinguishing Visual China from Places365 samples) using Cosine distance with $k = 1, 2,$ or 3. These findings confirm that: (1) fine-tuned DenseNet-201 is highly effective for landmark classification; (2) KNN on deep features is a potent strategy for OOD detection in this domain; and (3) Cosine distance is superior to Euclidean distance for this task, particularly with small k. This combined approach provides a practical and effective solution for enhancing tourist experiences through automated landmark identification.

VIII. Future Work

In order to improve landmark classification, a possible continuation of this work would be the implementation of a classifier based on *Vision Transformers* (ViT). This neural network architecture has shown promising results in computer vision tasks, offering an innovative approach to image processing. Instead of using convolutions, *Vision Transformers* divide the image into smaller parts and process them as sequences, similar to text processing in natural language models. This feature allows the model to capture long-range dependency relationships in the image, which can be particularly useful for identifying landmarks with complex variations in their appearance.

In addition, the use of Vision Transformers can lead to an improvement in classification accuracy, especially in cases where landmarks present variations in their appearance due to different lighting conditions, weather or capture angle. Another possible continuation would be the implementation of an image search system capable of finding similar landmarks from an input image. Such a system could be useful for tourists who want to explore landmarks similar to a location they have visited, or to help identify landmarks from low-quality images.

Despite the promising results, a real system would require rigorous testing to assess the relationship between the quality of predictions and the number of image classes. A larger number of classes may require a more complex model or a set of models with divided responsibilities. In addition, it may increase the computational costs of storing the galleries and executing the KNN algorithm. A possible solution would be to investigate strategies to reduce the size of the ID dataset, such as using clustering techniques to identify the most representative points.

Acknowledgment

This work was supported by CNPq (grants 308400/2022-4, 307151/2022-0), CAPES (grant 001), FAPEMIG (grant APQ-01647-22). We also thank the UFOP/PPGCC for their support.

References

[1] statista web site, "Travel & tourism - worldwide," 2024. [Online]. Available: https://www.statista.com/outlook/mmo/travel-tourism/worldwide#:~:text=By%202024%2C%20revenue%20is%20estimated, US%241%2C114.00bn%20by%202029.

[2] J.-W. Bi, H. Li, and Z.-P. Fan, "Tourism demand forecasting with time series imaging: A deep learning model," *Annals of Tourism Research*, vol. 90, p. 103255, 2021.

[3] R. Law, G. Li, D. K. C. Fong, and X. Han, "Tourism demand forecasting: A deep learning approach," *Annals of Tourism Research*, vol. 75, pp. 410–423, 2019.

[4] P. Roy, J. H. Setu, A. N. Binti, F. Y. Koly, and N. Jahan, "Tourist spot recognition using machine learning algorithms," in *Intelligent Communication Technologies and Virtual Mobile Networks*, G. Rajakumar, K.-L. Du, C. Vuppalapati, and G. N. Beligiannis, Eds. Singapore: Springer Nature Singapore, 2023, pp. 99–110.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, vol. 25, 2012, pp. 1097–1105.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[7] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," in *5th International Conference on Learning Representations, ICLR*, 2017. [Online]. Available: https://openreview.net/forum?id=Hkg4TI9xl

[8] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," in *6th International Conference on Learning Representations, ICLR*, 2018. [Online]. Available: https://openreview.net/forum?id=H1VGkIxRZ

[9] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 86–94.

[10] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus, "Intriguing properties of neural networks," 2014. [Online]. Available: http://arxiv.org/abs/1312.6199

[11] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," pp. 2261–2269, 2017.

[12] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[13] V. C. Group. (2024) Visual china. Recurso eletrônico. [Online]. Available: https://www.kaggle.com/datasets/protectoryao/visual-china

[14] P. Kumar. (2022) Places365. Recurso eletrônico. [Online]. Available: https://www.kaggle.com/datasets/pankajkumar2002/places365

[15] P. Roy, J. Setu, A. Binti, F. Koly, and N. Jahan, "Tourist spot recognition using machine learning algorithms," pp. 99–110, 01 2023.

[16] Y.-C. Chen, K.-M. Yu, T.-H. Kao, and H.-L. Hsieh, "Deep learning based real-time tourist spots detection and recognition mechanism," *Science Progress*, vol. 104, no. 3_suppl, p. 00368504211044228, 2021.

[17] Y.-C. Hsu, Y. Shen, H. Jin, and Z. Kira, "Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 948–10 957.

[18] Y. Sun, M. Yifei, Z. Xiaojin, and L. Yixuan, "Out-of-distribution detection with deep nearest neighbors," in *Proceedings of the 39th International Conference on Machine Learning*, vol. 162, 2022. [Online]. Available: https://proceedings.mlr.press/v162/sun22d/sun22d.pdf

[19] J. Li, P. Chen, Z. He, S. Yu, S. Liu, and J. Jia, " Rethinking Out-of-distribution (OOD) Detection: Masked Image Modeling is All You Need ," in *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 11 578–11 589.

[20] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, *Dive into Deep Learning*. Cambridge University Press, 2023.

[21] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[22] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2021.

[23] IBM. What is the k-nearest neighbors (knn) algorithm? [Online]. Available: https://www.ibm.com/think/topics/knn#:~:text=The%20k%2Dnearest%20neighbors%20(KNN)%20algorithm%20is%20a%20non,of%20an%20individual%20data%20point.

[24] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017. [Online]. Available: https://arxiv.org/abs/1712.04621

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR*, 2015.

[26] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: a system for large-scale machine learning," in *Proceedings of the 12th USENIX Conference on Operating Systems Design and Implementation*. USENIX Association, 2016, p. 265–283.

[27] F. Chollet *et al.*, "keras," 2015.

[28] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.