Travel Me Now - Convolutional Neural Networks in Augmented Reality Exploration of Paris

Miruna-Elena Iosub
"Alexandru Ioan Cuza" University of Iasi, Faculty of Computer Science
St. General Henri Mathias Berthelot 16, Iași 700483
mirunaiosub24@gmail.com

Adrian Iftene
"Alexandru Ioan Cuza" University of Iasi, Faculty of Computer Science
St. General Henri Mathias Berthelot 16, Iași 700483
adiftene@gmail.com

ABSTRACT

Augmented reality (AR) is being increasingly utilized across many diverse fields. Its applications span medicine, education, gaming, interior and exterior design, attracting a wide range of age groups, particularly young people excited by technological innovations. Today, several AR applications are available to help tourists navigate cities and uncover interesting information interactively. In this paper, we propose an application that can recognize certain tourist attractions from Paris, France more exactly- the Eiffel Tower, the Louvre Pyramids, the Arc de Triomphe, the Notre-Dame de Paris, and the Orsay Museum with the help of AR and with the help of convolutional neural networks (CNNs). Using AR, once the user has arrived at the tourist destination and classified it with the help of CNN, they will have an immersive experience in which they will learn various curiosities and historical details about the place and the visiting schedule.

Author Keywords

Augmented reality; convolutional neural networks; mobile application

ACM Classification Keywords

H.5.2. Information interfaces and presentation (e.g., HCI): User Interfaces. H.3.2. Information Storage and Retrieval: Information Storage.

General Terms

Human Factors; Design; Measurement.

DOI: 10.37789/icusi.2024.4

INTRODUCTION

Over the past decade, augmented reality (AR) has revolutionized various applications, adapting them to leverage this emerging technology. Although AR is commonly associated with leisure and entertainment [1, 2], its applications extend into critical sectors such as medicine [3, 4] and education [5, 6]. This wide-ranging utility highlights AR's unique ability to overlay informative and interactive content onto the real world, tailored to the application's purpose. The primary goal of the application proposed in this paper is to facilitate the discovery of key points of interest in Paris, France, through an interactive AR approach similar to [7]. The first component is a convolutional neural network (CNN) designed to recognize tourist attractions in Paris. To build this image classifier, we collected data by photographing various tourist sites and sourcing images from the Internet. We trained the CNN with different models and hyperparameters, ultimately selecting the model that achieved the best balance of accuracy and prediction speed for integration into the AR mobile application. The AR application includes two main features: real-time detection of tourist sites using the classifier, providing users with information such as operating hours, interesting facts, and interior views superimposed on the real world. The second feature is a text recognition and translation component that helps foreign tourists understand information on signs and boards at various attractions. Additionally, we developed a Web API application to retrieve detailed information about places in Paris by category, utilizing the Google Maps API. This application also communicates with a database that stores data on points of interest recognized by the image classifier. The final part of the project is a mobile application that integrates all these components. It offers an interactive map showing various tourist attractions by category, provides routing to these sites, and includes an AR feature for discovering points of interest.

RELATED WORK

Review of Image Classification Models

In this section, we review several image classification models that have been foundational in the field of computer vision and are relevant to our work.

ResNet

ResNet is highly beneficial because of its innovative residual connections, which effectively combat the vanishing gradient problem, allowing for the training of much deeper networks. ResNet50, with its 50 layers, strikes an excellent balance between depth and computational cost, offering high accuracy without becoming overly complex. This makes it particularly well-suited for a wide range of image classification tasks where high accuracy is essential.

VGG19

VGG19 is advantageous due to its simplicity and effectiveness. Despite its depth (19 layers), it uses small 3x3 filters that are easy to implement and well-understood, making it a popular choice in the research community. Its strong performance in competitions like ILSVRC14

highlights its robustness, especially in tasks requiring precision in image classification. The model's simplicity also makes it easier to fine-tune and apply in various domains.

DenseNet

DenseNet, introduced by Gao Huang et al., represents a significant advancement in efficient neural network architecture. It features connections between layers, where each layer receives input from all preceding layers. This design not only mitigates the vanishing gradient problem but also reduces the number of parameters, making DenseNet both powerful and efficient.

InceptionV3

InceptionV3, a continuation of the Inception family originally introduced as GoogLeNet, is known for its sophisticated architecture that reduces computational costs while maintaining high accuracy. It incorporates techniques such as factorized convolutions and aggressive dimensionality reduction, making it an ideal choice for large-scale image classification tasks.

Custom Model

Custom models are often developed to address specific challenges that standard architectures may not fully capture. For our project, we designed a custom convolutional neural network tailored to the classification of tourist attractions. The model incorporates data augmentation techniques and a tailored architecture to ensure high performance and adaptability to our unique dataset.

Existing Applications

These applications inspired new ideas and helped identify existing gaps that informed the addition of new features.

Paris Travel Guide Offline

Paris Travel Guide Offline[1] offers a comprehensive offline travel guide available through a premium subscription. It provides detailed information on various aspects of traveling in Paris, including tours, hotels, and local attractions (see Figure 1 for details). The application features a language tool to assist users in overcoming language barriers, making it easier to communicate with locals.

Paris Guide Tickets & Hotels

Paris Guide Tickets & Hotels[2] provides a variety of travel features designed to present the visitor with an unforgettable experience in Paris (see Figure 2 for details). One of its key features is the "What to See" recommendation section, which offers suggestions for attractions and landmarks. Users can easily book tickets by getting redirected to official websites, ensuring secure and reliable transactions.

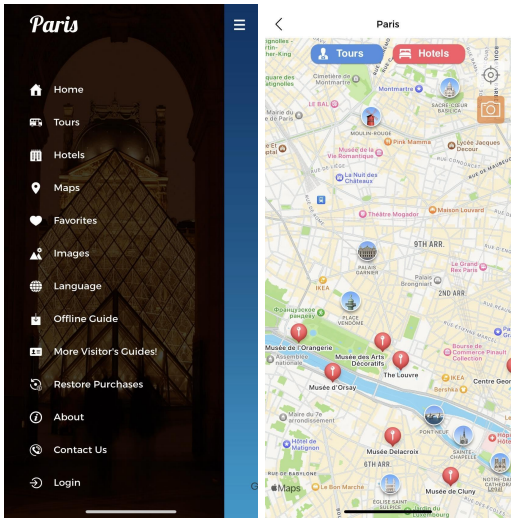

Figure 1. Paris Travel Guide Offline

Additionally, the application offers personalized city walks, generating customized itineraries based on user preferences and interests. This allows travelers to explore the city at their convenience while ensuring they don't miss out on key sights. The map feature offers detailed navigation to help users find their way around the city effortlessly.

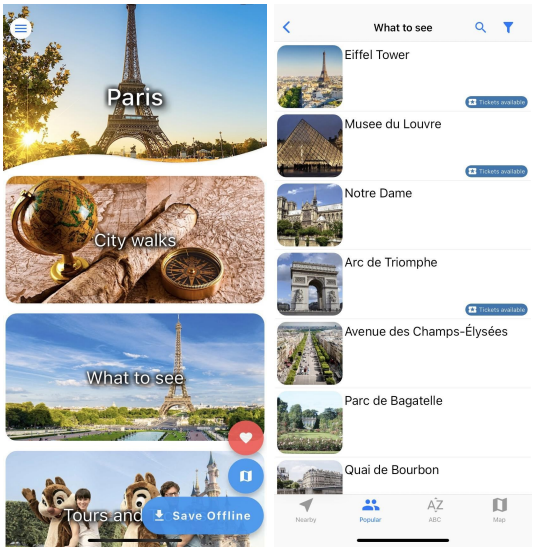

Figure 2. Paris Guide Tickets & Hotels

Paris Travel Guide

Paris Travel Guide[3] offers a multitude of functionalities, one considerable feature being *People's Choice*, which provides recommendations for points of interest such as top museums and popular *fun time* spots in Paris based on user preferences and reviews. Similar to other travel apps, it allows users to book tickets by redirecting them to official websites, ensuring secure navigation and transactions. Besides, the guide offers extensive information and historical facts about various attractions, which

[1] https://apps.apple.com/us/app/paris-travel-guide-offline/id396870085

[2] https://apps.apple.com/us/app/paris-guide-tickets-hotels/id1460779937

[3] https://apps.apple.com/us/app/paris-travel-guide/id330954821

helps the visitor understand and appreciate each place even more. With these features, the Paris Travel Guide serves as a comprehensive resource for exploring the city (see Figure 3).

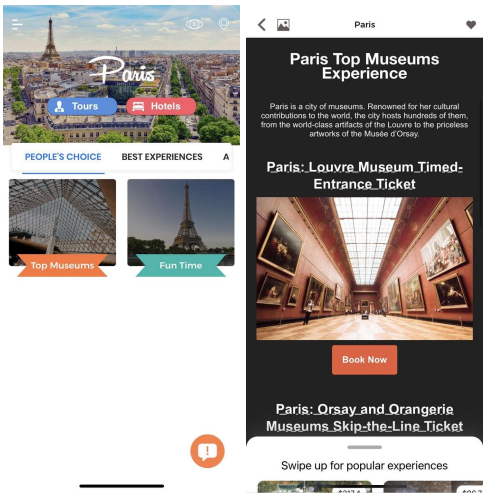

Figure 3. Paris Travel Guide

DATASET USED

As a source, we used the internet, especially Instagram, Google Images, Google Maps, Google Street View, and the TripAdvisor application. The rest of the photos were taken with the help of the phone in a short period, in February of 2024. The images capture the tourist sites from different angles, being taken both close to the points of interest and from a distance to have a wide range of ways in which they can be immortalized. For convenience, we also made videos from which we later extracted some frames. Some example images from the dataset for part of the points of interest are illustrated below in Figure 4.

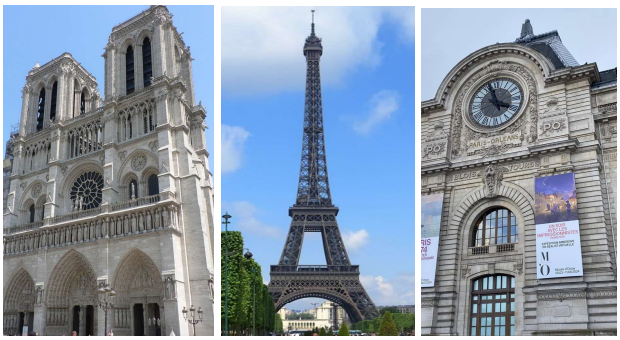

Figure 4. Notre-Dame Cathedral, Eiffel Tower, and Orsay Museum.

TRAINED MODELS FOR THE CLASSIFICATION OF TOURIST ATTRACTIONS

To obtain a model that classifies points of interest with high accuracy, we trained various neural network (NN) models which are presented below. The figures of the preprocessed images represent the transformed versions of the original images that have been modified to make them suitable for input into a machine-learning model. Preprocessing can involve several steps, such as resizing and normalization (scaling pixel values to a certain range, commonly between 0 and 1 or -1 and 1, to ensure consistency and help the model converge faster during training).

The **Residual Network** is a type of deep neural network architecture introduced by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun in their 2015 paper [8]. The key innovation of ResNet is the use of residual connections, which help mitigate the vanishing gradient problem, allowing for the training of much deeper networks. ResNet50 is a specific version of ResNet with 50 layers. This architecture represents a significant advancement in deep learning, particularly having the ability to train very deep networks efficiently. This CNN model creates a balance between depth and performance, making it an optimal choice for a wide range of computer vision applications. To prepare the dataset properly we mostly used the same technique, we split it into 3 parts: 80% for training images, and 10% for validation and testing (examples from the dataset are in Figure 5). We used most of the time the same parameters, such as a learning rate of 0. 0001, Adam's algorithm for optimization, cross-entropy loss function, and a batch size of 16. It managed to reach an accuracy of 99.7% on the test dataset for two main landmarks: Notre-Dame Cathedral and Eiffel Tower.

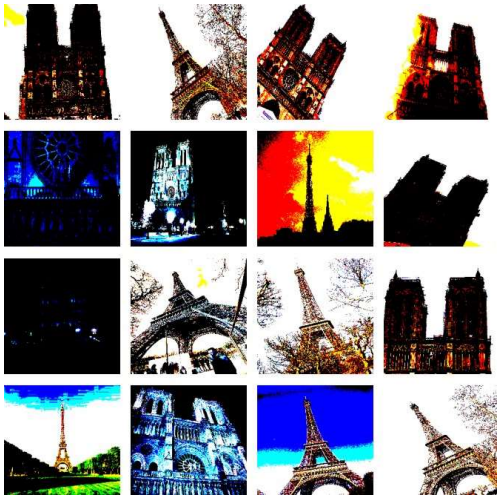

Figure 5. Preprocessed Images ResNet50 Model

The VGG19 model [9] is a convolutional neural network that won the ILSVRC18 competition in 2014 being able to classify images labeled in multiple categories with an accuracy of 92%. Visual Geometry Group was developed by researchers from Oxford University. The model gets as input an array of RGB format. The kernel matrices found in the convolutional layers have a size of 3×3 with a stride of one. At the same time, the max pooling layer used a filter

matrix of 2×2 size with a stride of two. Hidden layers use an activation function called ReLU, and the output layer uses an activation function called Softmax. To classify the two categories, we decided to use as parameters the following: cross-entropy loss function, a batch size of 64, 15 epochs, an Adam algorithm for optimization, and a learning rate with a value as small as possible. After training the model, an accuracy of 99.9% was reached after predicting the images from the part of the dataset that is meant to check the testing input images (examples from the dataset are in Figure 6).

Figure 6. Preprocessed Images VGG19 Model

The confusion matrix of the VGG19 model shows how many pictures were classified correctly (Figure 7). Based on the colors shown on the color bar, acting as a legend, the mapping between the colors and the numeric were shown, for example, the pink color means that the images classified wrongly was nearly 0, while yellow and orange means that most of the input images were classified correctly.

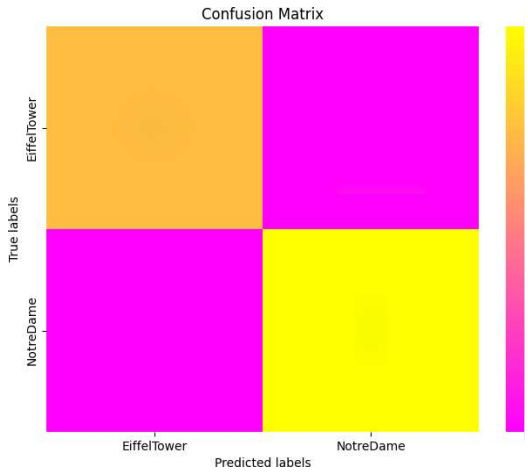

Figure 7. Confusion Matrix VGG19 Model

DenseNet [10] is a type of deep convolutional neural network that has been recognized for its strong performance in image classification tasks. DenseNet's architecture is characterized by dense connections between layers, where each layer receives input from all preceding layers. This design helps to mitigate the gradient problem and reduces the number of parameters, making the network more efficient and easier to train. DenseNet architecture is pre-trained on ImageNet and then fine-tuned for specific classification tasks. The model's input is set to (224, 224, 3), matching the standard input dimensions for many image classification models (see Figure 8). We've added a global average pooling layer to reduce the number of parameters and followed it with a fully connected layer with 512 neurons and ReLU activation. The final output consists of 2 neurons with softmax activation, corresponding to the 2 classes in the dataset. The batch size for training is set to
16. Adam optimizer is using a learning rate of 0. 0001 and a categorical cross-entropy loss function

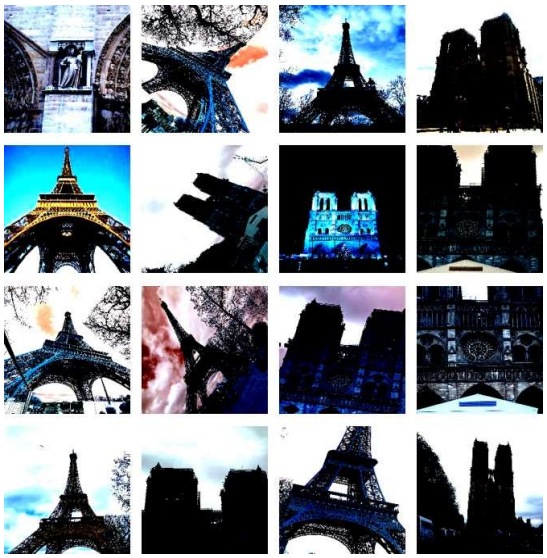

Figure 8. Preprocessed Images DenseNet Model

Inception V3 Model is a deep learning model that earned 1st Runner Up spot in the ILSVRC21 competition in 2015 and its architecture was first introduced by GoogLeNet named Inception V1. The Inception V3 model is an extension of the model created before named Inception V1. It incorporates multiple optimization techniques to better adapt to new training data. One major improvement is reducing the kernel sizes in the convolutional layers, which decreases the number of parameters and reduces the computational resources needed. We can see in Figure 9 examples with preprocessed images.

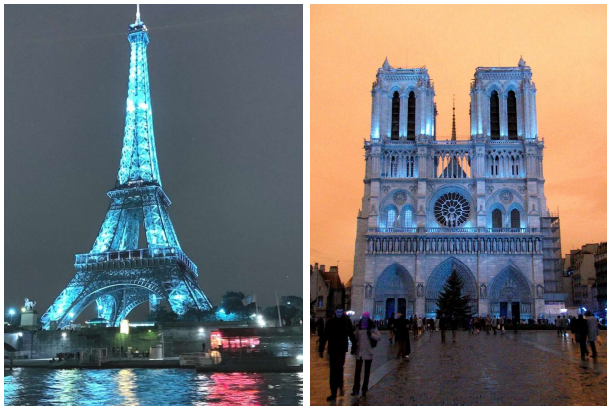

Figure 9. Preprocessed Images Inception V3 Model

The custom model has an architecture that employs convolutional layers for image classification, resizing images to 341×256 pixels and normalizing them (see Figure 10). It comprises five convolutional blocks with increasing filters (64 to 1024) and max-pooling layers to capture detailed features efficiently. A global average pooling layer condenses information, followed by dense layers with dropout for regularization. The final layer utilizes softmax activation to classify images into two categories. Compiled with the Adam optimizer and cross-entropy loss function, the model is trained for 15 epochs.

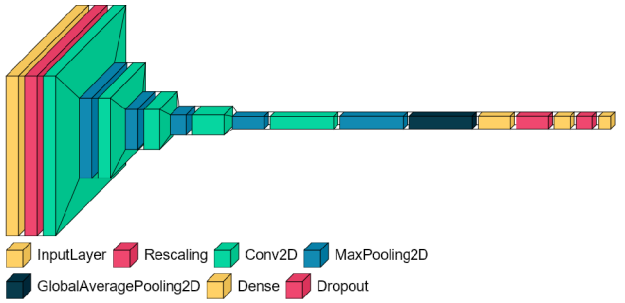

Figure 10. Custom Model's Architecture

In addition to the model architecture, data augmentation techniques were used to create a more complex dataset (see Figure 11). This process was meant to create new pictures by incorporating transformations like random shifts, brightness, contrast adjustments, RGB shifts, and horizontal flipping. The dataset was split into training, validation, and test sets using the *splitfolders* module. The custom model reached an accuracy of 100% during the training process with a batch size of 54. The high accuracy attained by the model is proof of a successful training process and the quality of the augmented dataset used.

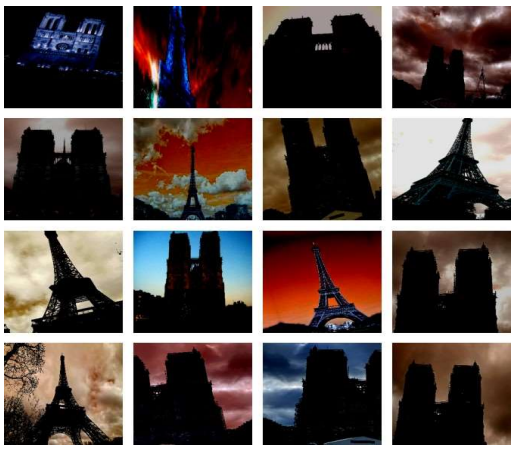

Figure 11. Augmented Images Custom Model

Figure 12 presents the confusion matrix for the custom model, revealing that all images were classified correctly.

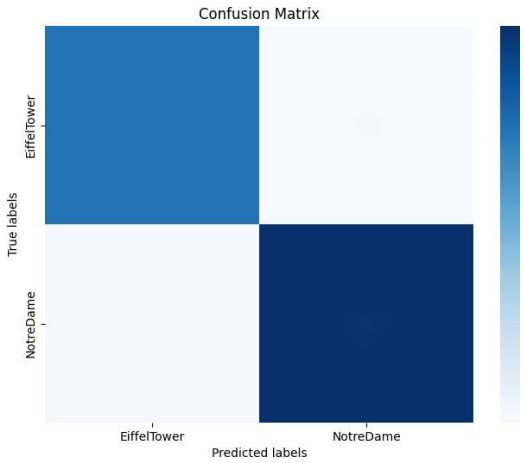

Figure 12. Confusion Matrix Custom Model

ARCHITECTURE AND IMPLEMENTATION

The system's architecture is designed with a strong emphasis on user experience and interaction, key components of HCI. This section presents the architecture and implementation of the whole system. It shows in detail concepts such as defining its functionality, performance, and user experience. The application includes features like a language preference option and a filtering service for landmarks, which are crucial for accessibility and personalization - key aspects of HCI. Figure 13 illustrates the services accessed by the application. Primarily, it is an Angular frontend application interacting with a .NET application and its services, as well as a Python FastAPI app for image classification. The image classification service also retrieves information from a database to display on the screen for the augmented reality feature.

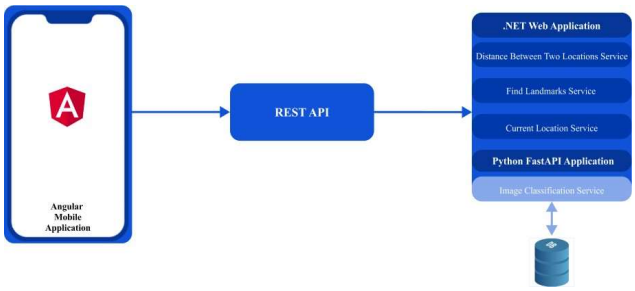

Figure 13. Diagram of services accessed by the application

Front End

For the front end of the application, we utilized Angular[5]. This framework allows for dynamic and responsive web development and we chose it for its performance and maintainability.

Homepage

The home page of the application contains a start button and a language preference feature located in the left corner. This language selection feature is implemented by creating a JSON file that holds the translations for different languages. Including language preferences is beneficial for user accessibility and experience because it allows users to interact with the application in their preferred language. This is one of the main features that makes the application more user-friendly.

Filtering Service for Landmarks

This section defines the main menu of the application, where landmarks are categorized into various filters such as top attractions, restaurants and cafes, hotels, and more. The filtering service allows users to easily navigate through different places, helping them find relevant information efficiently. Additionally, the main menu includes the AR experience button, which redirects users to the augmented reality feature.

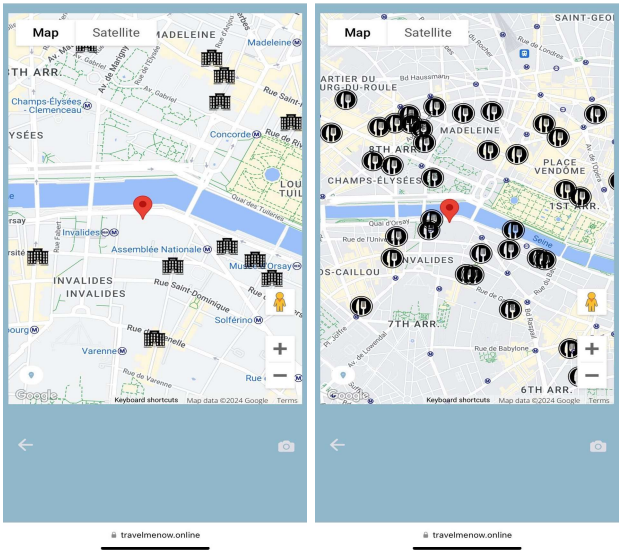

Figure 14. Landmarks Service

Landmarks View Service

The Landmarks View Service contains an integrated map using the Maps API in Javascript (see Figure 14). The client initiates a request by providing longitude coordinates from its location and a special word for the landmark.

Augmented Reality Experience Service

The AR Experience Service initiates by requesting camera permission from the user. Once permission is granted, it proceeds to detect landmarks from captured images. Upon detection, it overlays detailed information about the landmarks directly onto the image, including historical background, schedules, and a link to the official website for further exploration (see more details in Figure 15). This overlay of information onto the live camera feed is what constitutes the AR feature.

If the image does not correspond to any identifiable landmark, the service prints “Not identified” on the screen along with a link to the homepage of the application.

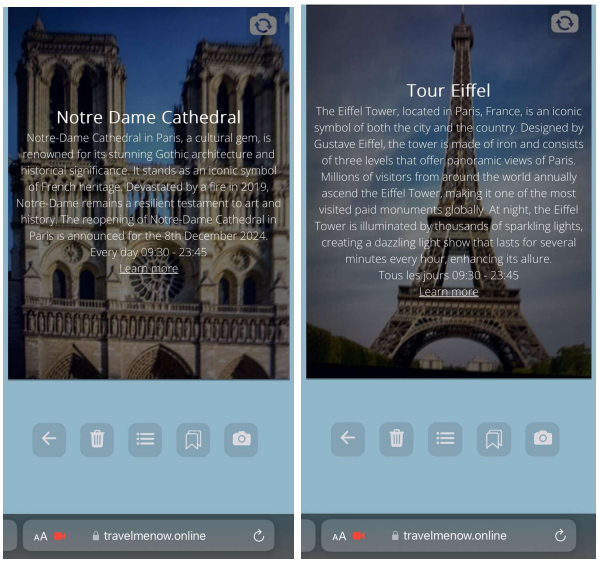

Figure 15. Augmented Reality Experience Overview

Bookmarks Feature

This feature enables users to save identified landmarks from the Augmented Reality Experience into local storage for future reference. When users encounter a landmark through the AR experience that they find interesting or want to revisit later, they have the option to bookmark it.

Back End

This section presents the backend architecture, which consists of two distinct servers designed to handle different aspects of the application’s functionality. The back end
includes two servers built using the .NET framework. These servers are responsible for providing services related to the integrated map functionality available in the front end. Additionally, there is a server built using Python and FastAPI. This server primarily manages the image classification services.

[5] https://angular.dev/

The key responsibilities of this server include utilizing CNN models to classify images and identify landmarks within them, and interacting with the database to retrieve necessary information.

Image Classification Service

Upon detection, the system utilizes a trained model to classify the image and determine whether it corresponds to a known landmark. The classification process is driven by a sophisticated neural network model that has been trained on a large dataset of tourist attraction images. Specifically, this model is loaded into the application using TensorFlow/ Keras and performs several steps. The image is received by the application in Base64 format. It is decoded and resized to a format suitable for the model, typically 224×224 pixels, as specified in the code. This resizing is crucial to ensure compatibility with the neural network's expected input dimensions. Before feeding the image into the model, it undergoes preprocessing to match the conditions under which the model was trained. This includes normalizing pixel values, which scales the pixel values to a range appropriate for the model. The preprocessed image is then passed through the trained model to generate predictions. The model outputs an array of probabilities, with each value corresponding to the likelihood that the image belongs to a particular class (e.g., *Eiffel Tower, Louvre Museum*, etc.). The system checks the highest probability value against a predefined threshold (in this case, 0.99). If the highest probability is below this threshold, the model determines that the image does not correspond to any known landmark and assigns it a label of "NotIdentified." If the threshold is exceeded, the model selects the class with the highest probability as the predicted landmark. Once a landmark is identified, the application retrieves detailed information about it from a database. This information includes historical background, opening hours, and a link to the official website. The data is structured into a JSON response that is sent back to the user. If the image does not match any known landmark (i.e., the classification result is "NotIdentified"), the system responds with a "Not identified" message and provides a link to the application's homepage for further exploration. This approach ensures that the system delivers accurate and meaningful results to the user, leveraging the powerful capabilities of deep learning models to enhance the experience of exploring tourist attractions.

Distance Between Two Locations Service

This service is meant to return the distance between two locations in kilometers and its ETA for walking. Its parameters are the longitudinal coordinates from the main destination. To compute the distance, we used Distance Matrix API.

Find Landmarks Service

To get the landmarks near the user's location, this service first has to specify the longitudinal coordinates from the user's location and also a keyword that was also used in the integrated map to filter all the places by category. Places API holds an important role because it was mainly used to create the HTTP call. The response consists of a list of found landmarks, with certain details such as name, reviews from users, schedule and address.

Current Location Service

In order to be able to find the exact address based on the longitudinal coordinates, this service makes a call to the Geocoding API, and returns the street name, city and county, corresponding to the coordinates received.

TESTING AND VALIDATION

To ensure the robustness and effectiveness of the proposed AR application, rigorous testing and validation were conducted. This section outlines the detailed performance metrics of the convolutional neural network (CNN) models employed, along with the evaluation of the application's response time and resource usage.

Performance Metrics

The performance of each CNN model was evaluated based on accuracy, prediction time, and the confusion matrix. These metrics provide a comprehensive understanding of how well each model performed in recognizing the selected tourist attractions.

Models Accuracy

The following table summarizes the accuracy achieved by each model:

Model	Accuracy (%)
ResNet50	99.7
VGG19	99.9
DenseNet	100
InceptionV3	100
Custom Model	100

Table 1. Models accurcy.

Prediction Time

The following table presents the average prediction time for the models that achieved 100% accuracy.

The Custom Model demonstrated the fastest prediction time at 0.2 seconds, making it the most suitable model for real-time applications where quick responses are essential. InceptionV3 and DenseNet also performed well but were slightly slower.

CONCLUSIONS

Travel Me Now is available in Romanian, English, Spanish and French and offers an interactive method of exploring tourist attractions using augmented reality. It provides the ability to detect points of interest and access interesting information about them in an immersive experience. By integrating CNNs for landmark recognition and focusing on user-centered design, this application significantly contributes to the field of HCI.

Additionally, it offers the functionality of visualizing attractions through an integrated map within the application. In the future, the image classifier could be expanded to predict a wider range of points of interest, requiring additions to the database containing information about them.

Model	Prediction Time (seconds)
Custom Model	0.2
InceptionV3	0.7
DenseNet	0.6

Table 2. Average prediction time.

Furthermore, the option of creating a user account could be introduced, allowing users to save various information and implementing a recommendation system for tourist attractions based on their previous experiences [11].

REFERENCES

1. C. Dimitriu, L. Rumeghea, B. Buzila, B. Florea, D. Silion, A. Iftene. (2023). "Spot the Story. Blending Augmented Reality Storytelling and Social Awareness", In Workshop on Intelligent Information Systems (WIIS2023), October 19-21, 2023, Chisinau, Republic of Moldova, 115-127.
2. S.S. Oprița, A. Iftene. (2022). "Meowgical AR - A Game based on Augmented Reality", In Proceedings of 19th International Conference on Human-Computer Interaction (RoCHI 2022), October, 6-7, Craiova, Romania, Matrix Rom, 21-24.
3. C. Mărtin, A. Gheorghiu. (2020). "Augmented Reality in Medicine: Revolutionizing Surgery and Medical Training", Journal of Medical Technology.
4. E.E. Opait, D. Silion, A. Iftene, C. Luca, C. Corciova. (2024). "Mixed Realities Tools Used in Biomedical Education and Training", In Proceedings of the 18th International Conference on INnovations in Intelligent SysTems and Applications (INISTA 2024), 4-6 September 2024, Craiova, Romania.
5. M. Billinghurst, A. Duenser. (2012). "Augmented Reality in the Classroom", Computer, 45(7), 56-63, doi: 10.1109/MC.2012.111.
6. A. Simion, A. Iftene, D. Gîfu. (2021). "An Augmented Reality Piano Learning Tool", In Proceedings of the 18th International Conference on Human-Computer Interaction RoCHI 2021, 16-17 September, Bucharest, Romania, 134-141.
7. D.R. Iacob, A. Iftene. (2023). "Inside Iasi - Augmented Reality Application for Discovering the City of Iasi, Romania", In International Journal of User-System Interaction (IJUSI), 15(4), 83-100.
8. K. He, X. Zhang, S. Ren, J. Sun. (2015) "Deep Residual Learning for Image Recognition", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, 770-778, doi: 10.1109/CVPR.2016.90.
9. K. Simonyan, A. Zisserman (2015) "Very deep convolutional networks for large-scale image recognition", In the 3rd International Conference on Learning Representations (ICLR 2015), 1-14.
10. B. Wang, W. Pei, B. Xue, M. Zhang (2022). "Explaining Deep Convolutional Neural Networks for Image Classification by Evolving Local Interpretable Model-agnostic Explanations", arXiv 2211.15143, 1-12, https://arxiv.org/abs/2211.15143.
11. C. Șerban, L. Alboaie, A. Iftene. (2016). "Image and user profile-based recommendation system", In Workshop on Social Media and the Web of Linked Data (RUMOUR 2015) at EUROLAN 2015 Summer School on Linguistic Linked Open Data. 18 July 2015, Sibiu, Romania. Springer International Publishing Switzerland. D. Trandabăț and D. Gîfu (Eds.): EUROLAN 2015, CCIS 588, 1-16. DOI: 10.1007/ 978-3-319-32942-0_5.