DMPedia Lecture Notes in Computer Science & Engineering
Volume 1, Issue 2, pp.230-240, 2026
ICICCS-2026

Deep Learning Framework for Indian Heritage Site Classification and Virtual Tour Generation using Transfer Learning

Deepti Singh[1], Raghavender K.V[2], O.Obulesu[3], Amarnath Manchuri[4]

[1] Department of Computer Science and Engineering, G. Narayanamma Institute of Technology & Science, Hyderabad, India

[2] Associate Professor,[3] Associate Professor (Data Science), Department of Computer Science and Engineering, G. Narayanamma Institute of Technology & Science, Hyderabad, India

[4] Director (R&D), Virat Integral Research and Development Foundation, Hyderabad, India

deeptisingh0819@gmail.com[1], drkvraghavender@gnits.ac.in [2], obulesh194@gnits.ac.in [3], drmanchuri@gmail.com[4]

*Corresponding author: Deepti Singh (deeptisingh0819@gmail.com)

ABSTRACT

The Indian heritage sites represent several centuries of history, architecture, and diversity of various cultures. Creative approaches are required to secure accessibility and awareness, and to maintain and promote these places. Traditional methods of heritage interpretation often fail to deliver a detailed, scalable, and automated experience because they rely on human guides, fixed displays, or limited mobile applications. This paper proposes a deep learning-based system that integrates a virtual tour system and heritage site classification. We predict the 55 Indian monuments using the datasets (Kaggle + GitHub) with transfer learning using EfficientNetB0 and ResNet50. To improve generalisation, a unified dataset of 1,091 images was processed and augmented. The trained models achieved good classification performance and high validation accuracy (EfficientNetB0: 99.39%; ResNet50: 99.08%). To enhance cultural interaction, we combine the classifier with a virtual tour module that captures the inside and outside views of the architectural site, along with textual descriptions in JSON files, with each landmark having its own. Users can upload an architectural image, label the monument, and automatically navigate its interior via a structured virtual tour using a web-based GUI (React frontend + FastAPI backend). The proposed system addresses gaps in existing systems by being automated, scalable, and accessible. Future developments for this system include multilingual support, AR/VR support, and implementation as an offline mobile/web application to increase digital heritage preservation.

Keywords: Classification, Transfer Learning, EfficientNetB0, ResNet50, Indian Heritage, Virtual Tour.

1. Introduction

India is home to numerous world-renowned historical sites, including the Taj Mahal, Charminar, Qutub Minar, and the Ajanta Caves. Besides being the best example of the country's immense cultural diversity, these monuments also greatly influence foreign tourism. Nevertheless, most tourists have limited knowledge of the historical and architectural significance of these places due to the absence of guided tours, language barriers, and insufficient contextual information. Cultural heritage digitisation, together with artificial intelligence (AI), has become a paradigm shift in enhancing visitor experiences. Deep-learning-based image classification has demonstrated impressive effectiveness in identifying natural scenes, works of art, and architecture. Nonetheless, there are unique challenges in applying such methods to Indian monuments, including visual similarity among architectural styles, varying illumination conditions, and the lack of labelled datasets. The need for AI-driven cultural tourism is the driving force behind this research. We intend to provide a more enriching learning experience by automatically classifying monuments based on visual evidence and opening organised virtual tours. The suggested system aims to promote the identification of the highest possible recognition accuracy and user interaction, which is achieved by combining principles of transfer-learning-based classification with the use of a JSON-structured virtual tour guide presented through a web-based platform.

Proceedings of ICICCS-2026 (Poster Papers), DMPedia Lecture Notes in Computer Science & Engineering.
ISBN: 978-81-993813-1-5

2. Related Work

Deep learning has revolutionised computer vision, especially after the game-changing breakthrough of AlexNet by Krizhevsky et al. [1], which demonstrated the effectiveness of deep convolutional neural networks for large-scale image classification. This milestone was followed by the introduction of ResNet, proposed by He et al. [2], which incorporated residual connections, thereby making it easier to train ultra-deep networks by addressing the problem of vanishing gradients. More recently, the EfficientNet architecture by Tan and Le [3] achieved state-of-the-art accuracy by scaling network depth, width, and resolution proportionally, using fewer parameters. These architectures provide the basis for state-of-the-art transfer-learning applications in domains that require both accuracy and efficiency.In the domain of Indian heritage, several investigations have attempted to identify monuments using convolutional neural networks. Shelke et al. [4] proposed a CNN-based classifier for Indian monuments but encountered challenges due to a small data set and class imbalance, leading to low accuracy. Jindam et al. [5] used VGG16 transfer learning for temple recognition. They achieved better results than traditional CNNs, but it was still not possible to apply this approach to a larger set of monuments. Sharma et al. [6] applied MobileNet for classification on mobile devices in a lightweight scenario, but accuracy deteriorated significantly when tested on a more diverse set of monument categories. Beyond the context of India, cultural heritage scholars have used deep learning techniques for the classification and restoration of artwork. For example, Bianco et al. [7] used deep convolutional neural networks to classify painting styles and attribute artists, achieving high accuracy and demonstrating the adaptability of CNNs beyond the domain of natural images. On the same note, Carneiro et al. [8] also examined visual recognition techniques for heritage preservation projects in Europe. They therefore suggested applying artificial intelligence to preserve cultural resources.

Simultaneously, research on augmented and virtual reality (AR/VR) technology to improve cultural heritage experiences has been conducted. Billinghurst and Dunser [9] investigated the impact of applying AR systems in the museum setting. They demonstrated that introducing contextual information superimposed on the scene can improve visitor engagement. Similarly, Zhang et al. [10] emphasised the importance of VR in recreating immersive digital replicas of monuments, thereby making heritage accessible through remote visitation. Nevertheless, these systems were generally based on manually curated data rather than automated classification mechanisms, limiting their scalability and adaptability. In general, previous efforts have either focused very narrowly on the recognition of a limited number of monuments [4], [5] or on immersive interaction without automated recognition capabilities [9], [10]. The present study distinguishes itself by combining deep learning-based recognition with automated virtual tour generation, encompassing fifty-five monuments in full. By augmenting data to improve robustness and implementing a scalable, JSON-based virtual tour framework, this approach bridges the gap between recognition accuracy and user-centred exploration.

3. Research Methodology

The methodology of the proposed framework aims to support end-to-end classification of cultural heritage monuments and automated generation of virtual tours.

3.1 Data Collection: The dataset employed in this study was curated by combining two publicly available repositories of data. The primary source of the data was the Kaggle Indian Heritage Dataset, comprising 4,895 labelled images across 24 monument categories. The next secondary source was a GitHub repository containing 961 images of monuments across 48 categories. A total of 1,091 images of 55 Indian historical monuments were combined after manual filtering of the dataset, which contained duplicates, low-resolution images, and irrelevant samples. To gain clear insights into the model, each monument was mapped to a unique class identifier.The dataset posed three intrinsic problems: (i) skew in class representation, with some monuments containing as few as 7 images and others containing more than 100 images; (ii) dissimilar data acquisition conditions, with varying lighting, camera angles and resolutions; and (iii) high-intra-class visual similarity, as many of the Mughal architectural styles are similar in their structural appearances. Image preprocessing and augmentation techniques were used to mitigate these problems.

3.2 Preprocessing and Augmentation: Deep learning models like EfficientNetB0 and ResNet50 are generally trained using the ImageNet dataset, where all the images follow the same and uniform structure of 224 by 224 by 3 in RGB colour space. To make the monument dataset compatible with these pretrained models, raw images undergo preprocessing.

3.2.1 Resizing: Original monument images were in varying resolutions (e.g., 640×480, 1920×1080). Using bilinear interpolation, each image was resized to a fixed resolution of 224×224 pixels. The "×3" indicates three channels corresponding to the RGB colour space. This ensures that the convolutional filters of pretrained CNNs can be directly applied without structural mismatch. If the original image has dimensions (h, w, c), resizing applies a scaling transformation:

$$I^1 = \text{Resize}(I, 224{,}224).$$

Where I is the original image tensor, and I^1 is the resized tensor of shape (224,224,3)

3.2.2 Normalisation: Raw pixel intensity values are typically in the range [0,255]. Normalisation rescales each pixel to a floating-point value between 0 and 1:

$$I_{Norm} = I^1 / 255.$$

This avoids large gradient values during training, improves numerical stability, and accelerates convergence of optimisers such as Adam.

3.3 Data Augmentation: Since the dataset was small (1,091 images) and imbalanced (some monuments had fewer than 10 samples), Keras ImageDataGenerator was used to increase the dataset size and variability artificially.

3.3.1 Geometric Augmentations:

1. Random Rotations (±30°): Each image is rotated by a random angle θ within [-30°, +30°]. Interpolation fills missing pixel values. This simulates different viewpoints from which tourists may capture images.
$$I_{Rot} = R(\theta).\ I_{Norm}$$
where R(θ) is a 2D rotation matrix.
2. Flips (Horizontal/Vertical): Horizontal flips simulate mirrored views (e.g., monuments captured from left or right). Vertical flips, while less common in real-world captures, introduce symmetry-based variability.
3. Translations (±10%): The image is shifted randomly in the x- or y-direction by up to 10% of its width/height. Padding ensures no information loss.
4. Random Cropping / Zooming: Zoom range (0.8–1.2) simulates cameras capturing close-up architectural carvings or wide-angle monument views. Cropping focuses on subregions, forcing the CNN to learn partial structural cues.

3.3.2 Photometric Augmentations:

1. Brightness Scaling (0.7–1.3): Pixel values are multiplied by a scalar factor in [0.7, 1.3]. These mimics are photos taken under poor lighting (dim) or overexposed conditions.

$$I_{Bright}=\alpha \cdot I_{Norm},\ \alpha \in [0.7,1.3]$$

2. Contrast Enhancement: Adjusts pixel intensity relative to the mean intensity, making architectural edges more pronounced.
3. Colour Jitter: Random shifts in hue and saturation to mimic variations in camera sensors or environmental lighting.

3.3.3 Noise Injection:

1. Gaussian Noise: Random Gaussian-distributed values N (0, σ^2) are added to pixel values. This simulates sensor noise and prepares the model for noisy real-world inputs.
$$I_{Noisy}=I_{Norm}+\epsilon,\ \epsilon \sim N(0, \sigma^2)$$
2. Blurring: To simulate motion blur or camera rotation, a Gaussian kernel is combined with the image. As a result, even with low clarity, CNN is forced to extract substantial features.

The dataset was enlarged from 1,091 to 3,273 photos when augmentations were applied. To provide a consistent class distribution, each class was artificially balanced to 20 samples as per figure 1.

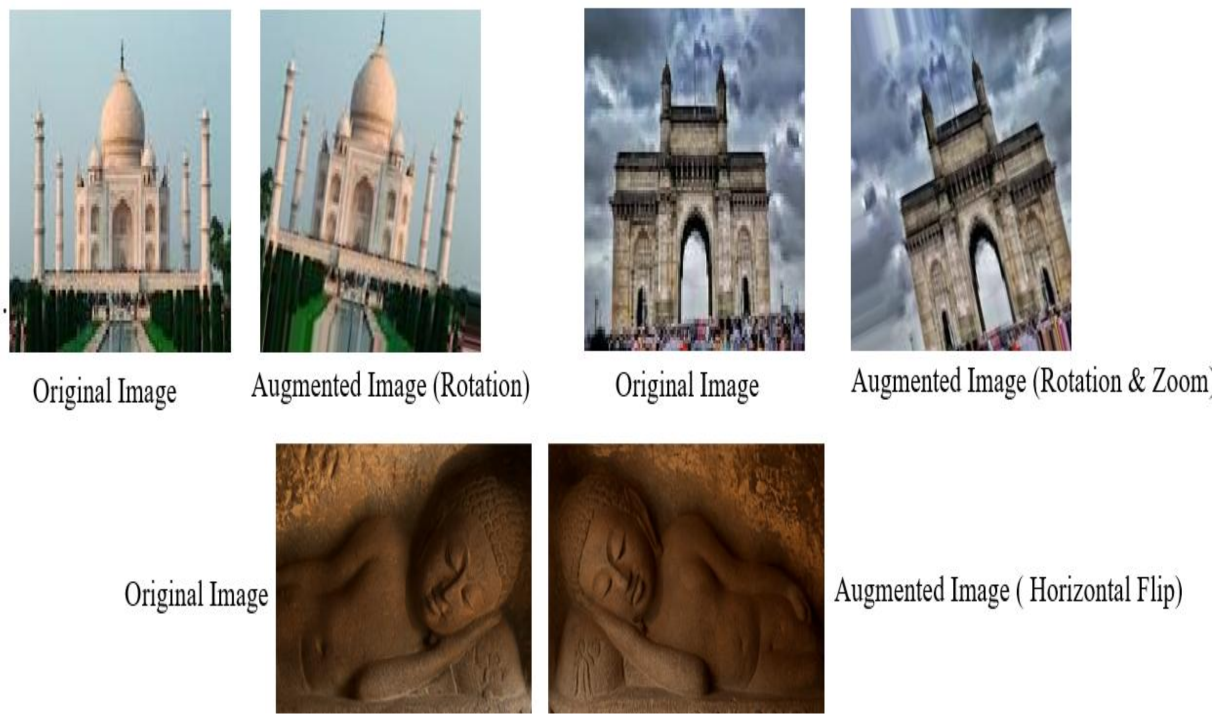

Figure 1: Dataset images and augmented samples

3.4 System Architecture: The proposed architecture is made up of a deep-learning classifier, EfficientNetB0, which is coupled with a virtual tour engine. A user uploads an image of a monument via the React-based frontend, which is processed and classified on the FastAPI backend using the trained EfficientNetB0 model. The predicted label then connects to a structured database (heritage tours. JSON) in the form of a JSON file that holds descriptive and visual assets of the monument. The client application now displays classification results and an orchestrated virtual tour, enabling users to transition from automatic recognition to culturally immersive exploration easily as per Figure 2.

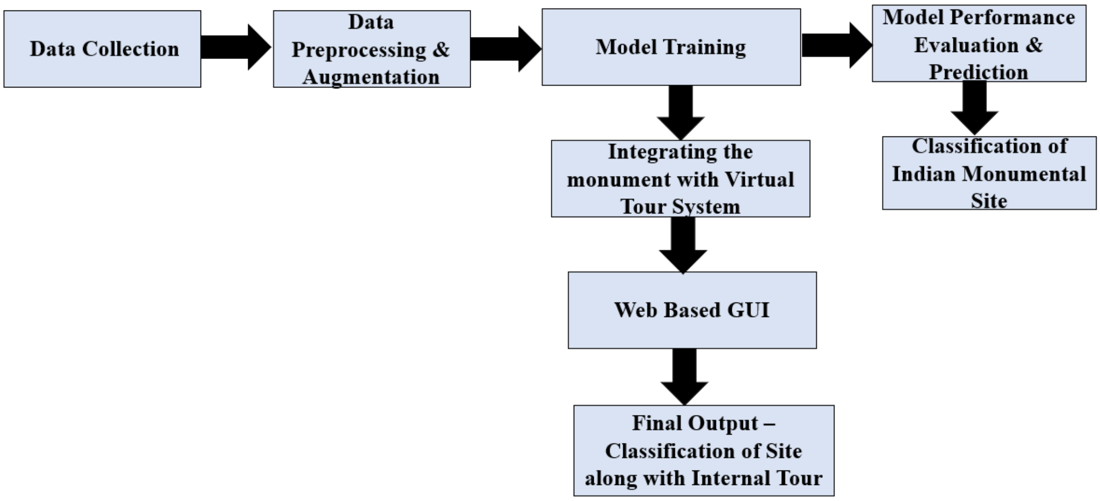

Figure 2: System Architecture

3.5 Deep Learning Framework: The proposed system classification backbone was developed based on transfer learning on two state-of-the-art architectures, EfficientNetB0 and ResNet50, both programmed to use ImageNet pre-trained weights. These networks were not used as general-purpose models. Still, they were adapted to the unique challenges of Indian heritage monument classification, such as high visual similarity among classes, fine-grained architectural patterns, and limited data.

3.5.1 Feature Extraction: EfficientNetB0 and ResNet50 convolutional layers were taken as the frozen base networks at the beginning of the training process. These layers extract low- and mid-level features, such as contours, textures, arches, and patterns, which are common characteristics of buildings. Since CNN filters are hierarchical, the model can gradually learn simple geometric shapes in the initial layers and more intricate patterns specific to monuments in subsequent layers.

3.5.2 Fine-Tuning Strategy: The final convolutional blocks were re-retrieved and used to fine-tune on the heritage dataset such that the networks were allowed to adapt to the peculiarities of the monuments. This retraining was selective, which allowed the network to modify weights at higher levels, enabling it to better distinguish between similar-looking monuments, such as Mughal and Indo-Islamic domes, and between inscriptions and carvings.

3.5.3 Custom Classification Head: On top of the feature extraction backbone, a lightweight task-specific classifier was appended. This head comprised:

1. A Global Average Pooling (GAP) layer, which incorporated spatial feature maps into a small vector, which greatly lowered the possibility of overfitting in contrast to flattening a fully connected layer.

2. A 512 neuron Dense layer with ReLU activation, which allows you to modify the extracted features in a non-linear manner and allows the model to be more powerful.

3. To regularise the network and prevent the neurons of the network from adapting too closely to each other, a Dropout layer with a probability of p=0.5 was introduced.

4. A SoftMax output layer that contains 55 neurons, each of which represents one of the various types of monuments to be classified.

The stratified nature of this structure makes it more flexible to the architectural imprints of monuments of Indian heritage, as well as to general-purpose visual features acquired from large data sets. Although the compound scaling method in EfficientNetB0 (balancing depth, width, and resolution) was more computationally efficient, the residual connections in ResNet50 enabled it to learn much deeper features. When used together, these models provided complementary strengths, as evaluated in a subsequent study.

3.6 Model Training and Optimisation: The models were trained using the following hyperparameters:

1. Optimiser: Adam with adaptive learning rate scheduling (initial 1e-4, reduced by a factor of 0.1 on plateau).
2. Loss: Categorical Cross-Entropy was used as a loss function
3. Batch size: 16.
4. Epochs: 25, with early stopping applied (patience = 5 epochs without improvement).
5. Regularisation techniques: Dropout (0.5) and L2 weight decay (1e-4).

To control for class imbalance, performance measures, such as accuracy, precision, recall, and F1-score, were calculated using macro-average metrics.

With a validation accuracy of 99.39, EfficientNetB0 was found to be better than ResNet50 (99.08) in an empirical analysis. EfficientNetB0 was selected as the deployment model due to its superior accuracy-to-parameter ratio (compared to 25.6M parameters in ResNet50) and lower inference latency, making it suitable for real-time systems.

3.7 Virtual Tour Integration: A correct classification and automated conversion from recognition to immersive examination of the monument through a virtual tour creation module is what renders the proposed framework unique. This has been made possible by mapping all 55 monument classes to a structured knowledge base defined in heritage_tours. json. This JSON database serves as a hierarchical data model, relating carefully selected textual and audiovisual data to classification outcomes.

All class labels of the monument’s classes act as individual keys in the key-value schema, which forms the JSON format, and its value is a nested object, which has:

1. The monument's exterior and interior elements are organised into manageable tour spots that provide a contextual impression (e.g., main gateway, prayer hall, domes, courtyards, inscriptions).
2. Descriptive narratives, which offer comprehensive descriptions of architecture, culture, and history. These chronologies were designed to artificially recreate the formation of a human tour guide and still maintain sectional coherence.
3. Linked multimedia objects, which contain well-selected visuals and images present in the assets folder of the project as local objects. The images were automatically pre-downloaded using automated scripts and associated with all elements of the JSON to ensure offline capabilities.

After classification, the predicted class label is fed into a lookup function that performs a precise key-based query in the JSON file. The JSON parser reads the associated object that contains the picture file paths and the description. This modular design enables scalability by eliminating the need for manual branching logic or hard-coded mappings. With the addition of new monuments, new JSON elements can be added to the dataset without altering the system's functionality. They can be defined as follows:

$$T(x)=Q(f(x))$$

In which Q ([.0]) refers to the query function on the JSON database and f(x) refers to the output of the classifier on an input image X. The output T(x) is a structured tour with {S, D, M} in which D is a set of descriptive narrative, M is the set of related multimedia objects and S is the sequence of sections.

The frontend of the system (React.js) dynamically creates the tour at runtime by displaying section S, followed sequentially by its corresponding description and photo as T(x) is retrieved. This is done through a content streaming system that ensures people move throughout the monument without difficulty, just as in a real guided tour.

By effectively bridging the gap between AI-based recognition and cultural interaction, this integration will ensure that every classification becomes a structured, educational, and interactive heritage discovery.

3.8 GUI Implementation: A modular architecture of the system is achieved with the help of a Fast API backend and a React.js frontend. The interface allows users to post images of heritage monuments, which are validated and sent to the backend. Once the data is received, the image is pre-processed, scaled, normalised, and formatted to meet the input requirements of the pre-trained EfficientNetB0 convolutional neural network at the backend.

After performing inference, the model produces a predicted monument label and an equivalent confidence score. To dynamically fetch pertinent virtual tour data, the prediction is used in the backend to query a well-organised JSON repository containing hand-selected written accounts and multimedia resources related to the monument. To provide a sensible and informative user experience, the frontend then displays classification results through a multi-step guided virtual tour with pictures of monuments and textual descriptions. This system is cross-platform, flexible, modular, and experiences low computational overhead

4. Results and Discussion

To ensure the model's performance was reliably tested on unseen samples, the experimental evaluation used an 80:20 train-validation split on the curated dataset. Training was performed over 25 epochs using the Adam optimiser, categorical cross-entropy loss, and an adaptive learning rate schedule that reduced the step size when validation accuracy plateaued. Regularisation was achieved using dropout and L2 weight decay, and a batch size of 16 was selected to balance stability during convergence and memory efficiency.

EfficientNetB0 achieved an accuracy of 99.39 per cent during both validation and training. The model obtained fine-grained features of the dataset's architectural patterns by leveraging its compound scaling method, which balances depth, width, and input resolution. The accuracy gain was greatly enhanced by the use of the preprocessing pipeline, which ensured compatibility with ImageNet-pretrained weights by resizing photos to 224x224 and normalising them to the range [0,1]. To achieve this performance, data augmentation was essential. Although photometric variations, such as brightness and contrast changes, rendered the model insensitive to different lighting scenarios, geometric transformations, such as random rotations and flips, exposed the network to multiple perspectives of the monuments. These augmentations can be attributed to the model's almost perfect validation accuracy, which generalised better to validation images, as they effectively broadened the training distribution. Table 1 summarises EfficientNetB0's overall performance characteristics, including precision, recall, and F1 Scores, supporting its deployment acceptability.

At 99.08%, ResNet50's validation accuracy was marginally lower. The greater number of trainable parameters (~25.6M vs ~5.3M in EfficientNetB0) increased the risk of overfitting on the very small dataset of 3,273 augmented images, even if residual connections allowed deep feature learning and reduced vanishing gradients. Because the final convolutional blocks were fine-tuned, the network was able to learn monument-specific features like domes, inscriptions, and arches that were not caught by the pretrained ImageNet filters, maintaining good classification capabilities. The efficiency benefit of compound scaling in low-data settings is highlighted by the slight performance improvement over EfficientNetB0, as reflected in Table 1, which demonstrates the superior trade-off between accuracy and efficiency attained by EfficientNetB0, further illustrating this comparison as per Table 1.

Table 1: Comparison of Results

Model	Precision	Recall	F1-Score	Validation Accuracy
EfficientNetB0	1.00	1.00	1.00	99.39%
ResNet50	0.99	0.99	0.99	99.08%

Both models' precision, recall, and F1 Scores were near 1.00 due to the balanced dataset produced by augmentation. Initially, some classes had fewer than ten instances; however, by balancing them to roughly twenty visuals per class, the classifier was prevented from being biased in favour of the majority classes. To prevent falsely inflated accuracy driven by skewed class distributions, this balance was paired with macro-averaged assessment measures to ensure performance increases were consistent across all monuments.

The system's end-to-end applicability was further confirmed by integrating EfficientNetB0 with the virtual tour engine. The guided tour began when classification was completed, and the projected class was mapped to the matching record in the structured JSON repository. In this case, classification robustness was essential, as even a single misclassification could redirect the user to an incorrect monument tour as per Figure 3.

Figure 3: GUI upload interface of the Indian Heritage Tour Guide System

The virtual tour component worked reliably in combination with the classifier, providing correct recognition followed by contextual exploration, thanks to the consistently high confidence evaluation during inference. Figures 3 and 4 show the GUI upload interface used for categorisation and the resulting virtual tour for the Taj Mahal, respectively, where the system smoothly moves from recognition to an interactive multi-step tour of the monument.

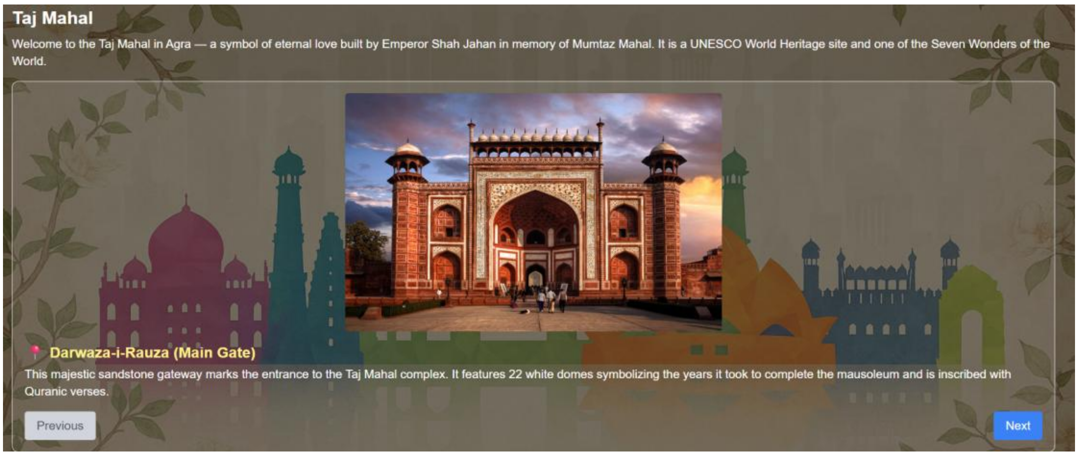

Figure 4: Virtual tour output for the Taj Mahal after classification

Overall, the findings indicate that the combination of transfer learning, targeted augmentation, and fine-tuning enabled the model to achieve high recognition performance. Along with high validation accuracy, low inference latency, and a minimal number of parameters, the EfficientNetB0 architecture ensured it was the best choice for deployment, which is why it was better suited to interactive, real-time tasks, such as the heritage virtual tour system. These findings are supported by Table 1 and proved by the figures. 3 and 4 show the extent to which the proposed paradigm can relate monument recognition to cultural involvement.
Based on the findings, EfficientNetB0 is slightly more efficient in terms of computation and validation accuracy than ResNet50. Given the limited sample size, the almost ideal accuracy suggests the possibility of overfitting. To resolve that, we propose collecting more differentiated data, e.g., realistic tourist photos under different lighting conditions. Nevertheless, several challenges remain. First, because cultural heritage datasets are skewed, it is hard to achieve uniform performance across all classes. Second, differences in lighting, viewpoint, and partial shadows can reduce the robustness of real-life tourist photographs. Third, the continuous curation and retraining of datasets would require adding additional monuments, which would become a scaling issue. Finally, the cost of real-time processing can also be a limitation when a mobile device is deployed that cannot access the internet.
Given these challenges, integrating virtual tours with classification is a new contribution. In practice, this method can be useful to cultural conservation groups, tourist departments, and learning institutions. The given system demonstrates how AI can bring history to life, making cultural property more accessible and engaging.

5. Conclusions

In this publication, a system of automated generation of virtual tours was integrated with a deep learning-based framework to classify Indian heritage monuments. The system employed transfer learning, using EfficientNetB0 and ResNet50 (pre-trained on ImageNet) to achieve impressive classification results with a validation accuracy of more than 99%. The models were capable of successfully generalising using data preprocessing, augmentation methods, and fine-tuning of higher-level convolutional layers, despite the difficulties posed by a fairly small and unbalanced dataset. EfficientNetB0 was more efficient in terms of parameter usage and inference speed than the other frameworks. Thus, it is suitable for use, especially in situations where lightweight deployment is needed, and ResNet50 confirmed the method's strength with its richer residual architecture.
The main innovative aspect of this framework is the seamless integration of classification and a virtual tour module, in which each monument class is linked to a structured knowledge base

in the form of a multimedia resource and a descriptive narrative in a JSON file. This design enables this system to transcend traditional picture recognition by providing the user with an interactive, educational exploration of monuments that closely resembles a cultural guided tour. The approach presents a combination of technological improvement and cultural enrichment by bridging the gap between real conservation of heritage and the study of artificial intelligence.

The system performed very well and suggested numerous areas that required further effort. The variety of real tourist images remains small compared to the dataset. Subsequent research will aim to expand the material through crowdsourced images, real-time deposits, and high-resolution scanning of inscriptions and carvings. This may enhance resilience in challenging situations, such as changing lighting, occlusions, or partial views. The framework can also be enhanced by using AR and VR technologies to allow users to navigate the categorised monument using real-time architectural features represented as an augmented overlay or an immersive 3D reconstruction. Such integration would greatly enhance user engagement and offer new teaching opportunities in both school and tourism settings. The proposed methodology provides a basis for AI-oriented cultural heritage conservation by showing that deep learning models can accurately classify cultural experiences, particularly when organised knowledge representation is used. The system can evolve into a full-fledged digital history reference, with additional extensions in dataset size, language support, AR/VR immersion, and offline access, making Indian cultural heritage much more accessible, engaging, and memorable for future generations to explore.

6. References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Lake Tahoe, NV, USA, 2012, pp. 1097–1105.

[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[3] M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Long Beach, CA, USA, 2019, pp. 6105–6114.

[4] P. Shelke, A. More, and R. Ingle, "Automated recognition of Indian heritage monuments using CNNs," *Int. J. Comput. Appl.*, vol. xx, no. xx, pp. xxx–xxx, 2023.

[5] R. Jindam and V. Reddy, "Heritage vs. non-heritage classification using deep learning," in *Proc. Int. Conf. Comput. Intell.*, City, Country, 2023, pp. xxx–xxx.

[6] S. Sharma and P. Patil, "Temple recognition on mobile devices using MobileNet," in *Proc. Int. Conf. Smart Technol.*, City, Country, 2021, pp. xxx–xxx.

[7] S. Bianco, M. Buzzelli, D. Mazzini, and R. Schettini, "Deep learning for artworks classification," *Pattern Recognit. Lett.*, vol. 113, pp. 83–90, 2018, doi: 10.1016/j.patrec.2018.07.026.

[8] G. Carneiro *et al*., "Computer vision for cultural heritage preservation," *J. Cult. Herit.*, vol. 45, no. xx, pp. 213–225, 2020.

[9] M. Billinghurst and A. Dünser, "Augmented reality in museums and heritage," *J. Cult. Herit.*, vol. 11, no. 4, pp. 456–463, 2010.

[10] J. Zhang, Y. Song, and C. Xiao, "Virtual reality for digital heritage: A review," *Comput. Graph*, vol. 98, pp. 142–153, 2021.