International Journal of Artificial Intelligence and Computer Electronics
Volume: 02, Issue: 03, July-September 2026, pp:33-43
P-ISSN 3117-8529 | E-ISSN 3117-8537

Quantum Vision–Guided Xception for Robust Visual Feature Learning Under Image Distortions

Mohammad Izhaar[1], Dr. Mohammed Jameel Hashmi[2]
[1]PG Scholar, Department Of Computer Science & Engineering ISL Engineering College, Hyderabad, India. India.
[2]Head of the Department; Department Of Computer Science & Engineering ISL Engineering College, Hyderabad, India.
Corresponding Author Email: mohammadizhaarahmed@gmail.com

Article Received 12-05-2026, Revised 21-06-2026, Accepted 12-08-2026

Abstract
This work proposes a novel Heavy QV-Xception model for advanced object recognition by integrating the wave-centric principles of Quantum Vision (QV) theory with the efficient architecture of Xception. Inspired by the particle-wave duality of quantum physics, QV theory interprets visual objects as dynamic information waves rather than static pixel-based images. The proposed framework incorporates a robust QV transformation block at the frontend to convert conventional spatial images into high-dimensional wave-function representations. This transformation enables the model to capture complex structural patterns, non-local statistical relationships, and dynamic contextual information that may not be effectively represented through conventional localized pixel processing.
The transformed wave-based features are subsequently processed through the Xception backbone, which employs depthwise separable convolutions to efficiently separate spatial feature extraction from cross-channel feature correlation. By combining quantum-inspired information representation with efficient deep convolutional learning, the Heavy QV-Xception model provides enhanced feature extraction and improved object recognition capability. Experimental evaluation on multiple benchmark datasets demonstrates that the proposed model consistently outperforms standard Xception and other conventional convolutional neural network models. The results highlight the potential of integrating Quantum Vision theory with advanced deep learning architectures to achieve more accurate and robust object recognition.

Keywords: *Quantum Vision (QV), Heavy QV-Xception, Object Recognition, Deep Learning, Xception, Wave-Function Representation, Quantum-Inspired Computing, Depthwise Separable Convolution, Convolutional Neural Networks, Feature Extraction.*

1. Introduction

Recent progress in computer vision has been strongly influenced by deep learning models capable of learning visual representations directly from image data. Convolutional neural networks (CNNs) have demonstrated substantial capability in extracting hierarchical patterns, beginning with low-level structures such as edges and textures and progressing toward higher-level semantic information. Krizhevsky, Sutskever, and Hinton established the effectiveness of deep convolutional architectures for large-scale image recognition, demonstrating that learned visual representations can significantly improve recognition performance [3]. Subsequent developments in deep architectures have focused on increasing representational depth while maintaining computational efficiency. The Xception architecture introduced by Chollet is particularly relevant because it employs depthwise separable convolution to decouple spatial feature extraction from channel-wise feature learning, thereby reducing computational requirements while retaining strong representational capability [5].

The increasing complexity of visual data has also encouraged researchers to investigate alternative representations beyond conventional pixel-domain processing. Quantum-inspired approaches provide one such direction by representing information through concepts derived from quantum states, superposition, and high-dimensional feature spaces. Cong, Choi, and Lukin introduced quantum convolutional neural networks as a framework for extracting hierarchical information using quantum-

inspired convolutional operations [25]. Similarly, Henderson et al. investigated quanvolutional neural networks, demonstrating how quantum circuits can be incorporated into image-processing pipelines to generate alternative feature representations [16]. These developments suggest that quantum-inspired representations can serve as an additional mechanism for transforming complex input information before conventional learning operations are applied.

Several quantum convolutional and optical learning studies have further explored the possibility of combining quantum principles with visual feature extraction. Parthasarathy and Bhowmik proposed a quantum optical convolutional neural network framework for image recognition, highlighting the potential of quantum-oriented representations for visual learning [11]. Hur, Kim, and Park examined quantum convolutional neural networks for classical data classification, showing that quantum-inspired processing can be considered even when the original input is not inherently quantum in nature [13]. These studies provide motivation for investigating alternative front-end transformations that can encode visual information into richer representations before conventional deep feature extraction.

At the same time, modern vision architectures continue to evolve toward more effective methods of learning global and contextual relationships. He et al. demonstrated the effectiveness of residual learning for constructing deep neural networks, while Huang et al. introduced dense connectivity to promote feature reuse across network layers [2], [20]. Transformer-based vision models have further expanded this direction by emphasizing relationships between image regions rather than relying exclusively on local convolutional operations. Han et al. reviewed the development of vision transformers and highlighted their ability to model long-range visual dependencies [19]. Vaswani et al.'s attention mechanism provides the broader conceptual foundation for such relationship-oriented representation learning [24].

Motivated by these developments, this study proposes a **Heavy QV-Xception model** that combines a Quantum Vision (QV)-based transformation stage with the Xception architecture. The proposed approach treats an input image as a structured information field and transforms it into a high-dimensional wave-function representation before conventional convolutional feature learning. The transformed representation is then supplied to Xception for efficient extraction of spatial and channel-wise characteristics. The objective is to investigate whether this combination can improve visual feature learning under image distortions while preserving the computational advantages of depthwise separable convolution. The proposed framework therefore explores the interaction between quantum-inspired representation, efficient deep convolutional learning, and robust visual recognition.

2. Literature Review

Krizhevsky, Sutskever, and Hinton introduced a deep convolutional neural network for large-scale image classification and demonstrated that hierarchical convolutional feature learning could substantially improve recognition performance. Their work established a strong foundation for subsequent CNN-based computer vision research by showing that increasingly complex visual patterns could be learned directly from image data rather than relying exclusively on handcrafted descriptors [3].

He, Zhang, Ren, and Sun proposed residual learning as a means of addressing optimization difficulties associated with increasingly deep neural networks. Their residual architecture enabled information to pass through shortcut connections, allowing substantially deeper models to be trained effectively. The concept became an important development in deep visual representation learning and influenced many later CNN architectures [2].

Tan and Le introduced EfficientNet with the objective of achieving a better balance between network depth, width, input resolution, and computational cost. Their compound scaling strategy demonstrated that systematic model scaling could provide improved accuracy without simply increasing network size. This work is relevant to the present study because computational efficiency remains important when sophisticated feature representations are introduced into a vision pipeline [4].

Chollet proposed Xception, an architecture based on depthwise separable convolution that separates spatial filtering from cross-channel feature learning. This design reduces redundant computation while maintaining strong feature extraction capability. The efficiency and modularity of Xception make it suitable as the deep-learning backbone of the proposed Heavy QV-Xception framework [5].

Xie et al. developed aggregated residual transformations through the ResNeXt architecture, showing that increasing the cardinality of parallel transformations could improve representation quality while maintaining manageable computational complexity. Their findings demonstrate the importance of combining multiple feature transformations when learning complex visual patterns [7].

Szegedy et al. proposed the Inception architecture to improve CNN representation by applying multiple convolutional operations at different scales. The resulting multi-scale feature extraction strategy allowed the network to capture patterns with varying spatial dimensions. Such multi-scale processing is

relevant to distorted image recognition, where important structures may appear at different scales or exhibit altered spatial characteristics [10].

Parthasarathy and Bhowmik investigated a quantum optical convolutional neural network for image recognition. Their study explored the integration of quantum optical concepts with convolutional processing and demonstrated the potential of quantum-oriented representations for visual information processing. This work provides a conceptual basis for exploring quantum-inspired transformations before conventional deep feature extraction [11].

Henderson, Shakya, Pradhan, and Cook investigated quanvolutional neural networks in which quantum circuits were used to transform image information before classical learning. Their work illustrated how quantum operations could be incorporated into an image-recognition workflow and provided motivation for investigating alternative front-end feature transformations [16].

Cong, Choi, and Lukin introduced quantum convolutional neural networks and demonstrated how hierarchical quantum operations could be organized for learning structured information. Their work is particularly relevant to quantum-inspired vision research because it establishes a connection between convolutional feature extraction and quantum information-processing principles [25].

Hur, Kim, and Park studied quantum convolutional neural networks for classical data classification. Their research indicates that quantum-inspired computational mechanisms need not be restricted to inherently quantum datasets and can potentially be applied to conventional data through suitable transformations [13].

Han et al. presented a comprehensive survey of vision transformers and discussed the transition from predominantly local convolutional processing toward architectures capable of modeling broader relationships within visual data. Their analysis emphasizes the growing importance of contextual and long-range information in modern computer vision systems [19].

Vaswani et al. introduced the attention mechanism through the Transformer architecture, establishing an influential approach for modeling relationships between different elements of an input sequence. Although originally developed for sequence processing, attention-based representation learning has subsequently influenced computer vision and motivates the broader objective of capturing non-local relationships within visual representations [24].

The reviewed studies indicate two important research directions: the continuous development of efficient deep visual architectures and the investigation of quantum or quantum-inspired mechanisms for alternative feature representation. Conventional CNN architectures such as Xception provide efficient and powerful feature extraction, whereas quantum-inspired approaches offer opportunities for transforming information into richer representations. The proposed Heavy QV-Xception model combines these directions by introducing a QV transformation stage before Xception-based feature learning, with the aim of improving robustness and discriminative capability under image distortions.

3. Methodology

Data Acquisition And Preprocessing

The first stage of the methodology involves collecting labelled image data suitable for object recognition. The selected dataset contains images belonging to multiple categories, allowing the proposed model to learn differences between object classes. Before training, the images are checked for consistency and prepared in a format compatible with the deep learning architecture. Operations such as image resizing, normalization, label encoding, and dataset partitioning are performed to establish a suitable learning environment. Data augmentation is used to improve the diversity of the training data and reduce excessive dependence on the exact appearance of the original samples. Transformations such as rotation, horizontal or vertical flipping where appropriate, scaling, and controlled noise variation can generate additional image patterns. These augmented samples help the model learn visual features that remain useful when objects appear under different orientations or imaging conditions. The prepared dataset is divided into training, validation, and testing subsets according to the experimental design. The training set is used to learn model parameters, the validation set assists in model selection and hyperparameter adjustment, and the independent test set is reserved for final performance assessment.

Quantum Vision Transformation

After preprocessing, the input image is passed to the Quantum Vision block. This stage represents the main quantum-inspired component of the proposed framework. Instead of relying exclusively on the original pixel representation, the QV block transforms the visual input into a wave-oriented feature representation inspired by the concept of particle-wave duality.

The purpose of the transformation is to provide an alternative representation of image information before deep convolutional processing. The

generated representation is intended to preserve important structural characteristics and provide additional information that may assist the subsequent network in learning complex visual patterns. The QV block therefore acts as a feature-enrichment stage rather than replacing the deep learning architecture.

The output of the QV transformation is supplied directly to the Xception backbone. By placing this transformation at the front end of the system, the proposed methodology allows the feature extractor to operate on a representation that differs from conventional raw pixel input.

Xception-Based Feature Extraction

The transformed representation generated by the QV block is processed using the Xception architecture. Xception is selected because of its use of depthwise separable convolution, an operation that separates spatial filtering from channel integration. In contrast to conventional convolution, which jointly processes spatial and channel information, depthwise separable convolution performs these operations in separate stages. This design can reduce the computational requirements of convolutional processing while maintaining the ability to construct deep hierarchical feature representations. As the QV-transformed data passes through the Xception network, the architecture progressively learns discriminative patterns associated with the object categories present in the dataset. The extracted feature maps contain increasingly abstract information about the visual content of the image. Early processing stages can capture lower-level patterns, whereas deeper layers learn more complex class-relevant characteristics. The resulting high-level feature representation is forwarded to the classification stage.

Classification

The classification stage converts the features extracted by the Xception backbone into predictions for the target object classes. The learned feature representation is passed through the final classification layers of the network. Depending on the implementation, these layers may include pooling, regularization, fully connected processing, and an output layer.

For a multi-class object-recognition problem, the final layer uses the Softmax activation function to generate a probability distribution across the available classes. If the output scores for a sample are represented by (z_1, z_2, \ldots, z_K), the probability of class (i) is calculated as:

[
P(y=i)=\frac{e^{z_i}}{\sum_{j=1}^{K}e^{z_j}}
]

The predicted class corresponds to the category with the highest probability value. The model is trained by comparing its predicted outputs with the known labels and updating the trainable parameters through an optimization process that minimizes the selected loss function.

Evaluation And Testing

After training, the Heavy QV-Xception model is evaluated using data that were not used for direct parameter learning. The purpose of this stage is to determine how effectively the model generalizes to new images. The proposed framework can be compared with baseline models such as conventional CNN architectures and standard Xception under the same dataset and experimental conditions.

Accuracy measures the overall proportion of correct predictions and is expressed as:

[
Accuracy=\frac{TP+TN}{TP+TN+FP+FN}
]

Precision evaluates the reliability of positive predictions:

[
Precision=\frac{TP}{TP+FP}
]

Recall measures the proportion of relevant instances that are correctly identified:

[
Recall=\frac{TP}{TP+FN}
]

The F1-score combines precision and recall through their harmonic mean:

[
F1=2\times\frac{Precision\times Recall}{Precision+Recall}
]

In addition to classification measures, computational efficiency should be assessed using practical indicators such as inference time, training time, parameter count, memory consumption, and computational operations where available. These measurements are important for determining whether the proposed architecture provides an

acceptable balance between recognition performance and computational cost.

Requirements Engineering

Hardware Requirements

The minimum hardware requirements should support dataset preprocessing and basic model development. A dual-core processor, 4 GB of RAM, and 250 GB of storage may be sufficient for small-scale experimentation or lightweight inference. However, these specifications are generally limited for efficient training of a heavy deep learning model. For practical training and experimentation, a modern multi-core processor, at least 8-16 GB of RAM, sufficient high-speed storage, and preferably a CUDA-compatible GPU are recommended. GPU acceleration can substantially reduce the time required for training convolutional neural networks and large image datasets.

Software Requirements

The Heavy QV-Xception system is implemented using Python and a compatible deep learning framework such as TensorFlow with Keras or PyTorch. A modern development environment, such as Jupyter Notebook, JupyterLab, Spyder, Visual Studio Code, or another Python IDE, may be used for implementation and experimentation.

The recommended software environment includes a current version of Windows or Linux, Python, the selected deep learning framework, NumPy, Pandas where required for data handling, OpenCV or Pillow for image processing, and Matplotlib for visualization. The exact library versions should be recorded to improve reproducibility.

4. Design Engineering

Design engineering defines the structural organization and operational design of the proposed Heavy QV-Xception object classification system. This stage describes how the individual components of the application interact, how image data move through the system, and how the trained deep learning model is integrated into the user-facing application. The design follows a modular approach so that data acquisition, preprocessing, model inference, classification, and result presentation can be managed as separate but connected processes.

The proposed framework is centred on the Heavy QV-Xception model, which combines a Quantum Vision-inspired transformation mechanism with the feature-learning capability of an Xception-based deep learning network. During model development, the input images are prepared and transformed according to the proposed QV methodology before hierarchical features are learned by the deep neural network. The resulting feature representation is then used to determine the most probable class of the input image.

The deployed application follows an inference-oriented workflow. A user accesses the system through a web interface, uploads an image, and the backend performs the preprocessing operations required by the trained model. The prepared image is passed to the stored Heavy QV-Xception model, which generates class probabilities. The system selects the class with the highest predicted probability and presents the classification result to the user.

The overall architecture can therefore be represented as follows:

User → Web Interface → Authentication → Image Upload → Image Preprocessing → QV-Xception Model → Probability Prediction → Class Selection → Result Display

The design can be represented using system architecture diagrams and UML models such as use-case, class, sequence, activity, component, and deployment diagrams. These representations help explain user interaction, internal processing, component relationships, and the environment in which the application operates.

1. Implementation

System Implementation

The core application logic is organized within the main Python application file, such as app.py. The trained model file, identified in the implementation as best_heavy_qv_xception11.h5, is loaded by the TensorFlow/Keras environment when the application is initialized or when required by the inference process.

A user begins by accessing the application through a web browser. New users may register, while registered users authenticate themselves using the login interface. After successful authentication, the user is provided with access to the image-classification functionality.

When an image is submitted, the backend receives the uploaded file and prepares it for inference. The image is loaded, resized to the dimensions required

International Journal of Artificial Intelligence and Computer Electronics
Volume: 02, Issue: 03, July-September 2026, pp:33-43
P-ISSN 3117-8529 | E-ISSN 3117-8537

by the trained model, converted into a numerical array, and processed using the preprocessing procedure consistent with model training. The resulting tensor is then supplied to the stored Heavy QV-Xception model.

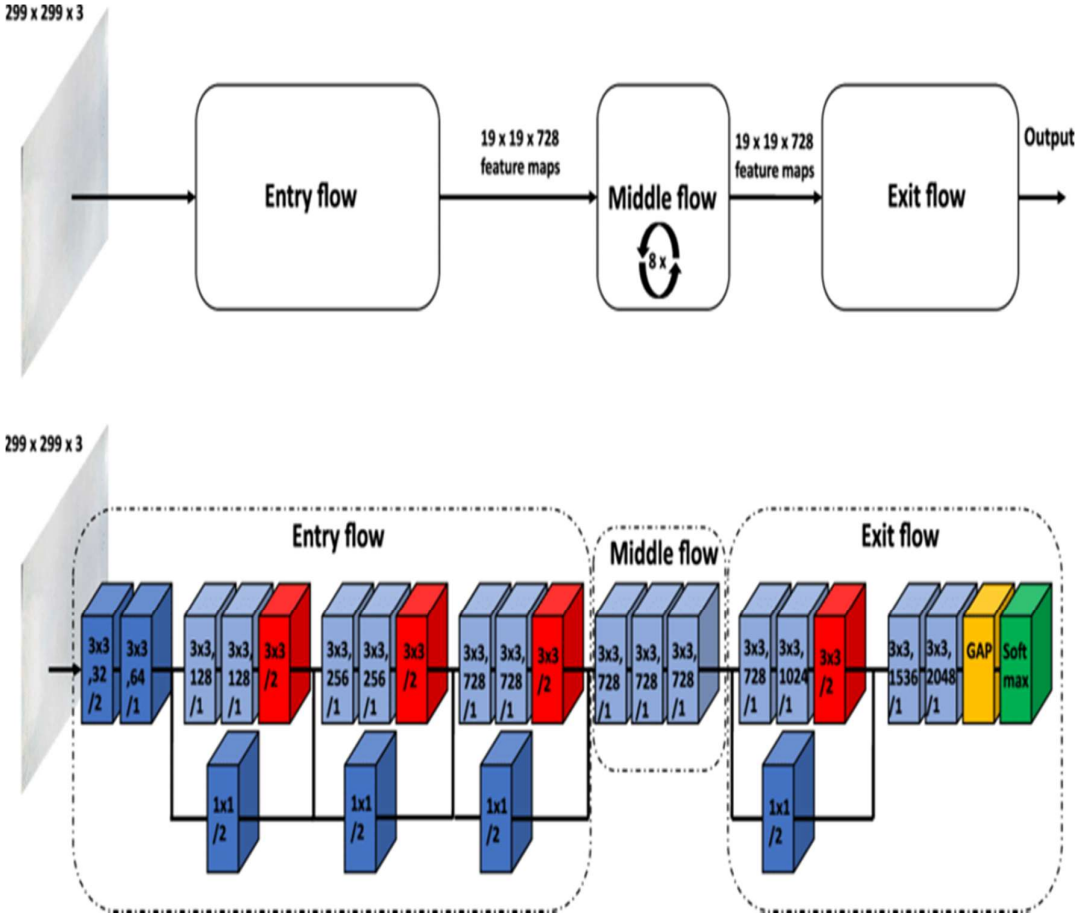

Figure 1: Overall System Architecture of the Heavy QV-Xception Framework

The model produces a set of output values representing the predicted probabilities or scores associated with the available image classes. The application identifies the highest-scoring class and returns the corresponding prediction to the web interface.

User Registration And Authentication

The authentication component provides controlled access to the classification system. A new user can create an account through the registration interface, and the submitted account information is stored in the application's local database after validation. Passwords should not be stored as readable plain-text values. Instead, a password-hashing mechanism, such as Werkzeug's generate_password_hash() function, can convert the password into a protected hash before it is written to the database. During authentication, the submitted password is verified against the stored hash using an appropriate verification function. After successful login, Flask session management maintains the user's authenticated state during navigation. This mechanism allows the application to restrict protected functionality to authorized users and maintain a consistent interaction flow.

Image Preprocessing Module

Image preprocessing ensures consistency between the data used during inference and the data format expected by the trained model. The uploaded image is loaded and converted into a numerical representation. It is then resized to the input dimensions used during model training.

After resizing, the image is converted into an array and processed using the preprocessing procedure associated with the Xception-based network. The

International Journal of Artificial Intelligence and Computer Electronics
Volume: 02, Issue: 03, July-September 2026, pp:33-43
P-ISSN 3117-8529 | E-ISSN 3117-8537

processed array is expanded to include the batch dimension required by the deep learning framework before prediction is performed.

The general inference preparation sequence is:

Uploaded Image → Image Loading → Resizing → Numerical Array Conversion → Model-Specific Preprocessing → Batch Preparation → Model Input

For the deployed Heavy QV-Xception model, the preprocessing procedure must match the exact input transformation used during training. If the Heavy QV transformation is embedded within the saved model, the application can supply the prepared image directly to the model. If it is implemented externally, the same QV transformation must be applied before inference. Maintaining consistency between training and deployment is essential for reliable predictions.

Heavy Qv-Xception Classification Module

The classification module represents the central intelligence component of the application. The trained model stored as best_heavy_qv_xception11.h5 is loaded through the TensorFlow/Keras framework.

Once preprocessing is complete, the prepared image tensor is passed to the model. The model applies the learned transformations and feature-extraction operations to generate an output associated with the available classes. For a multi-class classification problem, the output can be interpreted as a probability distribution when a Softmax-based classification layer is used.

The application determines the final prediction by identifying the index corresponding to the largest output value. This index is mapped to the appropriate class label, which is then returned to the user interface.

The complete prediction pipeline is represented as:

Input Image → Preprocessing → Heavy QV Transformation/Embedded Model Processing → Xception Feature Extraction → Classification Scores → Highest-Probability Class → Result Display

2. Results and Snapshots

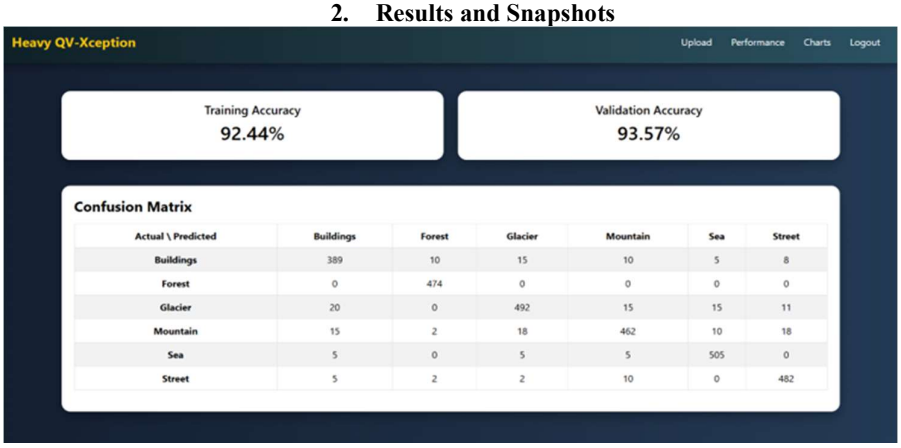

Figure 2

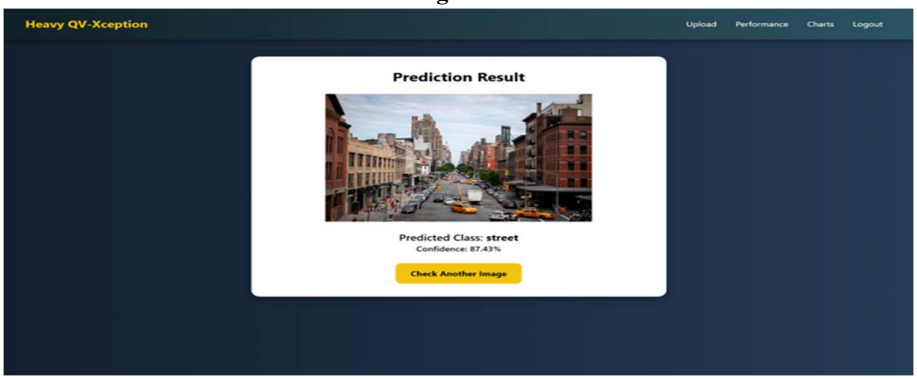

Figure 3
Various Snapshots

International Journal of Artificial Intelligence and Computer Electronics
Volume: 02, Issue: 03, July-September 2026, pp:33-43
P-ISSN 3117-8529 | E-ISSN 3117-8537

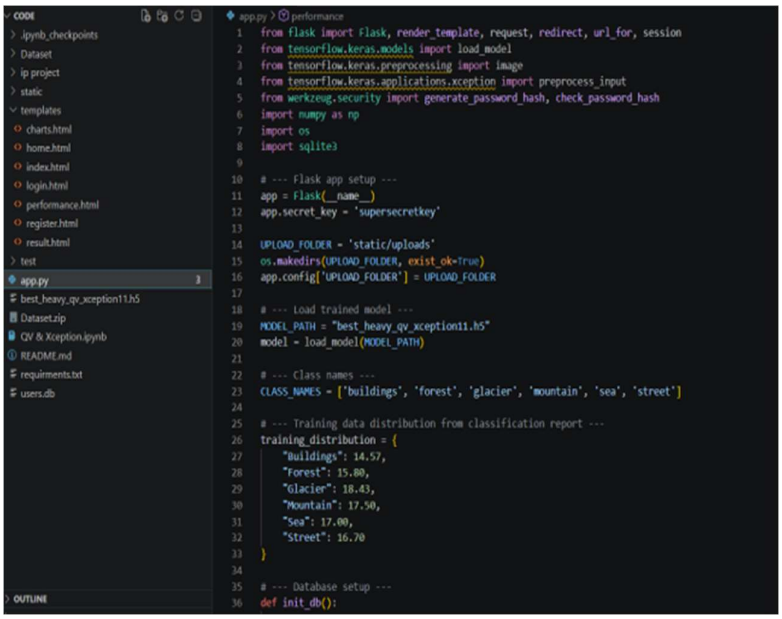

Figure 4

```
# BLOCK 1: DATA PREPARATION FOR XCEPTION + QUANTUM VISION

import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications.xception import preprocess_input

# Define paths
train_dir = 'archive/train'
test_dir = 'archive/test'

# Parameters
IMG_SIZE = (299, 299)  # Xception requires 299x299
BATCH_SIZE = 32

# Data generators with Xception preprocessing
train_datagen = ImageDataGenerator(
    preprocessing_function=preprocess_input,
    rotation_range=20,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True,
    fill_mode='nearest'
)

test_datagen = ImageDataGenerator(preprocessing_function=preprocess_input)

# Load train and test sets
train_generator = train_datagen.flow_from_directory(
    train_dir,
    target_size=IMG_SIZE,
    batch_size=BATCH_SIZE,
```

Figure 5

Future Enhancements

The proposed Heavy QV-Xception framework provides several opportunities for further development and improvement. Future research can focus on enhancing classification performance, reducing computational requirements, improving adaptability, and extending the system to a wider range of practical applications.

One possible enhancement is the incorporation of attention mechanisms. Self-attention and transformer-based modules could enable the model to focus more effectively on important visual regions. This may be particularly useful when images contain small objects, overlapping objects, complex backgrounds, or partially occluded features. By assigning greater importance to relevant regions, an enhanced architecture may improve its ability to recognize complex visual patterns.

Another important direction is model optimization for resource-constrained environments. Although

Xception uses depthwise separable convolution to improve computational efficiency, the overall architecture may still require considerable computational and memory resources. Techniques such as model pruning, quantization, parameter compression, and knowledge distillation could be explored to reduce model complexity while maintaining acceptable classification performance. Such improvements could support deployment on mobile devices, embedded systems, and edge-computing platforms.

The model could also be investigated in specialized application areas. Potential domains include autonomous navigation, intelligent surveillance, medical image analysis, remote sensing, industrial inspection, and smart robotics. Evaluating the architecture using domain-specific datasets would help determine its adaptability and effectiveness under different conditions.

3. Conclusion

This project presented the Heavy QV-Xception framework, a hybrid deep learning approach developed for image classification and object-recognition tasks. The proposed architecture combines a Quantum Vision Block with the Xception deep learning model to investigate the use of quantum-inspired image representations in conjunction with efficient convolutional feature extraction.

The Quantum Vision Block represents the first major component of the proposed framework. Its purpose is to transform conventional image information into a quantum-inspired wave-function representation. This transformation is intended to provide an alternative representation of visual information before it is processed by the deep learning architecture. By introducing this additional processing stage, the system investigates whether abstract and contextual visual characteristics can be represented in a form that is useful for classification.

The second major component of the framework is the Xception architecture. Xception performs hierarchical feature extraction using depthwise separable convolution operations. This mechanism separates spatial processing from channel-wise feature learning and provides an efficient alternative to conventional convolutional operations. Within the proposed framework, Xception processes the Quantum Vision-transformed representation and progressively learns visual features required for identifying the appropriate image category.

The complete system follows a structured workflow consisting of image acquisition, preprocessing, Quantum Vision transformation, deep feature extraction, classification, and result generation. Each stage performs a specific role within the overall architecture. Input images are first prepared in a suitable format, after which the Quantum Vision Block generates the transformed representation. The Xception network then extracts meaningful hierarchical features, and the classification stage produces the final prediction.

The proposed architecture is also implemented as a web-based application to demonstrate its practical use. The system integrates Python, Flask, TensorFlow, Keras, NumPy, image-processing utilities, and SQLite. Users can interact with the application through a browser interface, register and authenticate themselves, upload an image, and receive a classification result. This implementation makes the deep learning model more accessible by removing the requirement for users to interact directly with the underlying model code.

The performance of the proposed system can be examined using standard classification measures such as accuracy, precision, recall, and F1-score. These metrics provide information about the prediction capability of the model and support comparison with conventional CNNs, standard Xception architectures, and other hybrid or quantum-inspired approaches. Computational factors, including inference time and resource requirements, are also important when evaluating the practical applicability of the framework.

The Heavy QV-Xception project demonstrates the potential value of combining quantum-inspired visual representations with modern deep learning techniques. Rather than relying exclusively on conventional pixel-based input processing, the proposed framework introduces an additional representation stage before hierarchical feature extraction. This creates a hybrid approach that combines alternative feature representation with the efficient convolutional operations of Xception.

References

[1]. Sannidhanam, A. H. (2021). Real-Time Claims Processing Using Event-Driven

Architectures. *International Journal of Technology, Management and Humanities, 7*(01), 36–50. doi:10.21590/07.01.02

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

[3]. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[4]. M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," *Proceedings of the International Conference on Machine Learning (ICML)*, 2019, pp. 6105–6114.

[5]. F. Chollet, "Xception: Deep learning with depthwise separable convolutions," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1800–1807.

[6]. Anjani Haritha Sannidhanam. (2023). Zero-Downtime AI Model Updates in Real-Time Inference Systems. *International Journal of Engineering Science & Humanities, 13*(2), 70–78.

[7]. S. Xie et al., "Aggregated residual transformations for deep neural networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5987–5995.

[8]. Shaik Abdul Waheed, Mohammed Sulaiman, Mirza Awaiz Baig, Dr. Mohammed Abdul Bari," *Intelligent Conversational Agent for Seamless User Interaction Using NLP and Dialog flow*",International Journal of Information Technology and Computer Engineering, SSN 2347–3657 ,Volume 13, Special Issue 2s, 2025, Page -91-98.

[9]. Sannidhanam, A. H. (2025). Autonomous AI Agents for Cloud Infrastructure Operations. *Journal of Contemporary Science and Technology Management, 1*(01), 83–100.

[10]. C. Szegedy et al., "Going deeper with convolutions," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9.

[11]. R. Parthasarathy and R. T. Bhowmik, "Quantum optical convolutional neural network: A novel image recognition framework for quantum computing," *IEEE Access*, vol. 9, pp. 103337–103346, 2021.

[12]. L.K.Suresh Kumar , Mohammed Abdul Bari , "*Zero-Day Attack Detection In Multi-Tenant Cloud Environments Using Variational Autoencoder*s ," International Journal of Applied Mathematics, ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)Volume 38 No. 3s, 2025.

[13]. T. Hur, L. Kim, and D. K. Park, "Quantum convolutional neural network for classical data classification," *Quantum Machine Intelligence*, vol. 4, no. 1, 2022.

[14]. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *International Conference on Learning Representations (ICLR)*, 2015.

[15]. Sannidhanam, A. H. (2026). LLM-Driven Workflow Optimization: Intelligent Routing in Asynchronous Distributed Systems. *International Journal of AI and Machine Learning, 1*(2), 23–33.

[16]. M. Henderson, S. Shakya, S. Pradhan, and T. Cook, "Quanvolutional neural networks: Powering image recognition with quantum circuits," *Quantum Machine Intelligence*, vol. 2, no. 1, 2020.

[17]. Anjani Haritha Sannidhanam. (2024). Prompt Engineering Patterns for Reliable LLM Outputs in Production Environments. *Journal of Multidisciplinary Knowledge, 4*(1), 29–39.

[18]. Performance Analysis of Wireless Sensor Networks for Smart Monitoring Applications. (2016). *International Journal of Humanities and Information Technology, 1*(04), 10–29. doi:10.21590/ijhit.01.04.04

[19]. K. Han et al., "A survey on vision transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 1, pp. 87–110, 2023.

[20]. G. Huang, Z. Liu, L. Van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2261–2269.

[21]. . A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[22]. Sannidhanam, A. H. (2020). Comparative Analysis of Cloud Computing Architectures for Enterprise Applications. *SAMRIDDHI: A Journal of Physical Sciences, Engineering and Technology, 12*(02), 169–180. doi:10.18090//samriddhi.v12i02.16

[23]. Vishwanath Nikhil , Dr. Mohammed Abdul Bari *," Real Time Powerlifting Form Assessment using Yolov5 and Mediapipe*", International Journal of Information Technology and Computer Engineering, SSN 2347–3657, Volume 13, Special Issue 2s, 2025, Pages 574-584, jumble the order of the

references and rearrange properly and dont change any think from the references

[24]. A. Vaswani et al., "Attention is all you need," *Advances in Neural Information Processing Systems (NeurIPS)*, 2017, pp. 5998–6008.

[25]. I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, 2019.

[26]. Anjani Haritha Sannidhanam. (2023). Self-Healing Distributed Systems: AI-Driven Failure Prediction and Automated Recovery. *International Journal of Research & Technology, 11*(4), 168–174.

About Author

Mohammad Izhaar Ahmed is a Software Developer and Instructor at NxtWave, with a strong focus on building practical, intelligent, and scalable software systems. He holds an M.Tech in Computer Science and Engineering from ISL Engineering College. His technical interests span full-stack application development, DevOps, and Large Language Models (LLMs), with particular interest in the intersection of modern software engineering and emerging AI technologies.

His work emphasizes transforming ideas into deployable technology, combining robust software architectures with intelligent computing approaches. Among his notable projects are VeinReach, a technology-driven platform designed to facilitate blood donation, and Refyne, a collaborative code review platform aimed at improving software development workflows. His broader interests include exploring how AI and modern development practices can be integrated to create efficient, user-centric, and scalable software solutions.