Introduction to Heterogeneous Knowledge Transfer for Quantum Convolutional Neural Networks

Emily Jimin Roh, Kihun Lee, and Joongheon Kim
Department of Electrical and Computer Engineering, Korea University, Seoul, Republic of Korea
E-mails: {emilyjroh,heuk3a,joongheon}@korea.ac.kr

***Abstract*—The convolutional neural network (CNN) is widely utilized in computer vision due to its ability to effectively harness correlation information within data. However, when the data's dimensionality or the model's complexity becomes excessively large, CNN faces significant challenges in maintaining efficient learning. The quantum convolutional neural network (QCNN) offers an innovative solution by leveraging quantum computing environments, to address problems conventionally solved by CNN or enhance the performance of existing learning models. This paper proposes a model of the CNN structure for quantum computing and introduces a method for enhancing model performance by integrating heterogeneous knowledge transfer (HKT) for QCNN with the CNN model used in conventional computer vision tasks. Furthermore, this paper examines the feasibility of the QCNN-HKT model in comparison to CNN and QCNN by conducting training experiments on the KITTI dataset with quantum computing simulation libraries Torchquantum.**

I. Introduction

Quantum computers are rapidly emerging as an innovative solution to challenges that conventional computers have yet to overcome [1]. These devices introduce a unique computing paradigm, characterized by their use of superposition and entanglement—features not present in conventional computing environments—allowing for significant performance improvements through qubit parallelism [2]. Owing to these unique capabilities, make quantum computers increasingly recognized as a promising approach to solving complex algorithmic problems in real-world applications [3], [4]. As research advances in optimization techniques, such as gradient descent on quantum devices, the potential for more efficient quantum machine learning, particularly in fine-tuning hyperparameters, is becoming more apparent [5]. Among the various models, quantum convolutional neural networks (QCNNs) represent a specialized form of quantum neural networks (QNNs) that utilize QNNs as quantum convolutional filters. Due to their convolutional filter architecture, QCNNs have gained significant attention for their potential to reduce the number of qubits required [6]. Unlike the conventional QNN structure, which uses parameterized quantum circuits (PQC) as fully connected layers and thus requires increasing qubits based on input and output dimensions, QCNNs can function with fewer qubits by leveraging quantum convolution filters. This makes QCNNs particularly suitable for applications during the noisy intermediate-scale quantum (NISQ) era, such as autonomous driving. However, QCNNs are still in the developmental stages, unlike the well-established conventional CNNs, making their implementation more challenging [7].

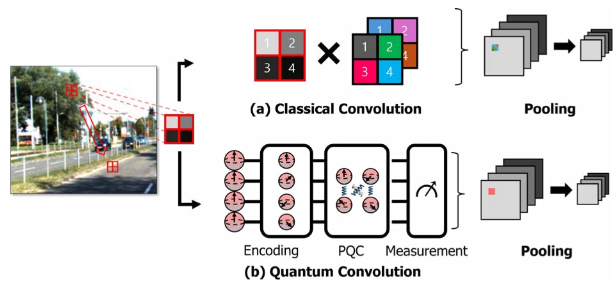

Fig. 1: Comparison between quantum convolution and conventional convolution.

Therefore, we introduce heterogeneous knowledge transfer (HKT) between conventional CNNs and QCNNs, QCNNs to enhance performance [8]. HKT is a training method that transfers knowledge from a teacher NN, typically a pre-trained conventional CNN, to a student NN, in this case, a target QCNN. Due to its simplicity and efficiency, HKT is widely used to compress models and improve performance. This paper presents a method for enhancing QCNN performance by integrating HKT with the CNN model used in conventional computer vision tasks. Additionally, the paper evaluates the feasibility of the QCNN-HKT model by comparing it to CNN and QCNN models through training experiments on the KITTI dataset using quantum computing simulation libraries such as Torchquantum.

II. Quantum Neural Network

QNNs are regarded as an advanced form of NNs that address the limitations inherent in conventional NNs [9]. A QNN is composed of three key processes: i) encoding, ii) PQC, and iii) decoding. This section provides an overview of these fundamental processes in QNNs, which are essential for constructing the proposed QCNN.

A. Basic Quantum Operations

In contrast to a conventional bit, which stores information as either $|0\rangle$ or $|1\rangle$, a qubit can exist in a superposition of states. The nature of a qubit can be mathematically represented as $|\Phi\rangle \triangleq \sum_{k=1}^{2^q} \alpha_k |k\rangle$, where q denotes the number of qubits, and α_k represents the corresponding amplitudes, satisfying $\forall q \in \mathbb{N}[1,\infty)$ and $\sum_{k=1}^{2^q} |\alpha_k|^2 = 1$. The initialized quantum state of q qubits is denoted as $|0\rangle^{\otimes q}$, where $\otimes$ indicates the tensor product. Quantum states are manipulated through quantum gates, which are operators analogous to conventional logic gates for bits. These quantum gates are unitary matrices. The set of quantum gates for encoding is denoted by U_E, while those within the PQC are represented by U_T.

B. Encoding

The encoding process is achieved through the application of the quantum gate set U_E. Utilizing this set of gates, the encoded quantum state is expressed as $|\psi_{\mathbf{x}}\rangle = U_E(\mathbf{x})|0\rangle^{\otimes q}$, where $\mathbf{x}$ represents the input conventional data.

C. Parameterized Quantum Circuit

Following the encoding of the quantum state $|\psi_{\mathbf{x}}\rangle = U_E(\mathbf{x})|0\rangle^{\otimes q}$, the PQC adjusts its parameters through the set of trainable gates [10]. The PQC is comprised of trainable rotation gates and controlled gates. The rotation gates modify the quantum state along the X, Y, and Z axes, while the controlled gates create entanglement among the target qubits. Consequently, the encoded quantum state $|\psi_{\mathbf{x}}\rangle$ is transformed into $|\psi_{\mathbf{x},\theta}\rangle = U_T(\theta)|\psi_{\mathbf{x}}\rangle$, where the PQC includes trainable parameters θ, distinguishing it from the encoding gates.

D. Decoding

Upon obtaining the transformed quantum state $|\psi_{x,\theta}\rangle$, it becomes essential to decode this quantum state into a conventional output. Given the inherent properties of qubits, quantum states cannot be directly utilized within conventional computing. Consequently, QNNs rely on the set of probability expectations. The expectation value $\langle O_{\mathbf{x},\theta}\rangle = \prod_{M\in\mathcal{M}} \langle\psi_{\mathbf{x},\theta}| M |\psi_{\mathbf{x},\theta}\rangle$ represents the expectation of the transformed quantum state $|\psi_{\mathbf{x},\theta}\rangle$ with respect to the Hermitian operator M. To decode the quantum information from the transformed quantum state, this paper employs Pauli-Z measurement, setting $\mathcal{M} = \{M_l\}_{l=1}^{q}$, where $M_l = I^{\otimes l-1} \otimes Z \otimes I^{\otimes L-l}$. Here, I and Z represent the identity matrix $\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$ and the Pauli-Z matrix $\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix}$, respectively. Through the decoding process, the QNN achieves values that satisfy $\langle O_{\mathbf{x},\theta}\rangle \in [-1,1]^{\otimes q}$.

III. Quantum Convolutional Neural Network

QNNs are recognized as next-generation NNs capable of addressing the limitations encountered by conventional NNs. However, the current availability of qubits still needs to be increased for many practical applications. Consequently, this paper focuses on the QCNN, which aims to overcome some of the constraints associated with QNNs. QCNN, inspired by the conventional CNN, substitutes the conventional CNN filters with those based on QNNs. By utilizing QCNN for input data analysis, it is possible to alleviate the limitations posed by the scarcity of qubits during the NISQ era [11]. Fig. 1 presents a comparison between conventional CNN and QCNN. In conventional CNN, a convolution operation is conducted, as shown in Fig. 1 (a). The convolution filter performs element-wise multiplication between the data within the sliding window of the input image and the trainable filter,

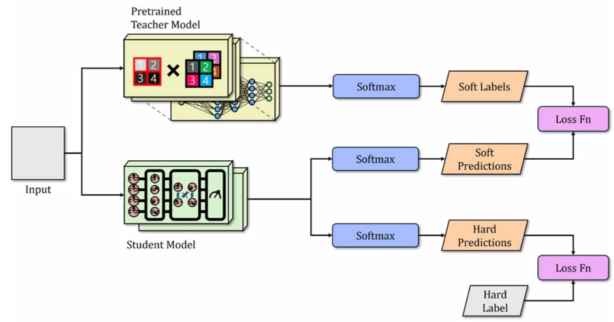

Fig. 2: Architecture of the heterogeneous knowledge transfer.

producing deterministic feature values [12]. During the training phase, the parameters of the trainable gates are adjusted to minimize the loss function. In contrast, QCNN employs a different convolution process. Fig. 1 (b) depicts the quantum convolution process within QCNN, where trainable quantum circuits, or QNNs, serve as the convolution filters [13]. Initially, conventional input data is encoded into the initialized quantum circuit. After encoding, the quantum states evolve through PQC, and the final quantum state is measured to be decoded into conventional values. The QCNN model integrates the convolution and pooling layers, which are essential components of CNN, into quantum systems. The convolution circuit discovers hidden states by applying multiple qubit gates between adjacent qubits. The pooling circuit then reduces the quantum system's size by either measuring a subset of the qubits or applying 2-qubit gates, such as CNOT gates. This sequence of convolution and pooling circuits is repeated. If the system size is sufficiently reduced, classification result is predicted by the fully connected circuit.

IV. Heterogeneous Knowledge Transfer

The QCNN is in the early stages of development, which presents several challenges in optimization techniques and training algorithms, leading to potential performance degradation. To mitigate this performance decline when applying QCNN to conventional CNN-based applications, this paper proposes the adoption of HKT as Fig. 2. KT is a training method widely employed in conventional NNs. KT is typically used when there are significant differences in performance characteristics between models. For instance, consider two conventional NNs: one with substantial width and depth and another with shallower dimensions. KT can be executed by designating the larger NN as the teacher and the smaller NN as the student, and by comparing the knowledge of these models. Additionally, if one NN is pre-trained while the other is not, KT can be applied by setting the pre-trained NN as the teacher and the other as the student. In this scenario, the target loss function is augmented with an additional loss regularizer, such as L2 loss, to facilitate KT by minimizing the difference between the logits of the models. This regularization enables robust student NN training. Drawing inspiration from this approach, this paper suggests implementing KT between

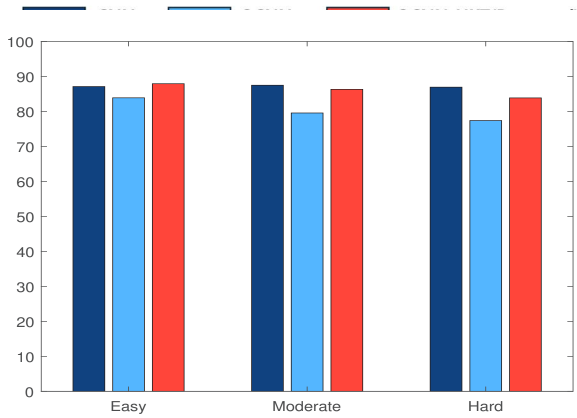

Fig. 3: Detection accuracy on KITTI for car category evaluated using the metric of the average precision AP_{50}.

heterogeneous models, where the teacher model is a pre-trained conventional CNN, and the student model is a QCNN. To achieve HKT between these heterogeneous models, the loss regularizer is defined as the difference between the logits of the pre-trained conventional CNN and the student QCNN. This regularizer is then incorporated into the loss function of the student QCNN, thereby allowing the student QCNN to be influenced by the logits of the teacher CNN. Thus, the loss function incorporating the HKT regularizer can be expressed as $\mathcal{L}_{total} = \mathcal{L}_{student}(y, \sigma(f(\langle O \rangle))) + \alpha D(\sigma(f(\langle O \rangle)), \sigma(\tilde{y}))$, where y and $\tilde{y}$ represent the actual labels and the predicted softened labels provided by the teacher model, respectively. $\mathcal{L}_{student}$ and $\mathcal{L}_{total}$ denote the original classification loss and the total loss with KT, respectively. $D(\cdot)$ represents the function calculating the difference between logits using softmax.

V. Performance Evaluation

For the simulation of the proposed QCNN-HKT, the software environment includes Python version 3.8.10, along with quantum computing simulation libraries torchquantum v0.1.5 and PyTorch version 1.8.2 LTS. To assess the performance and potential of QCNN-HKT, object detection is utilized as with the KITTI dataset. Fig. 3 illustrates the object detection results of conventional CNN, QCNN, and QCNN-HKT, each integrated with RPN modules for object detection. The performance of each model is evaluated based on the average precision (AP) across intersection over union (IoU) metrics for vehicles with scores on the (easy, moderate, hard) KITTI dataset. The conventional CNN achieves the highest AP across IoU metrics. However, the performance of QCNN is significantly improved when the HKT strategy is applied to the same metrics. Specifically, QCNN-HKT shows a performance increase in AP_{50} compared to QCNN alone, which is almost similar to conventional cnn. These results underscore the potential advantages of employing HKT training strategies for object detection during the NISQ era.

VI. Concluding Remarks

The QCNN integrates a CNN model with a quantum computing environment, allowing for a variety of innovative approaches. QCNN offers a potential solution for complex real-time classification tasks that are challenging to address with conventional methods and serves as a more effective and efficient learning model compared to conventional CNN models. Additionally, within the context of NISQ computers, QCNN is expected to deliver more efficient and advanced outcomes for complex and large-scale learning tasks. To further enhance performance, HKT is employed between different models. This paper corroborates the feasibility of QCNN with the HKT strategy.

Acknowledgements

This work was supported by Institute of This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2024-RS-2024-00436887) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation). J. Kim is the corresponding author of this paper (`joongheon@korea.ac.kr`).

References

[1] D. Ristè, M. P. Da Silva, C. A. Ryan, A. W. Cross, A. D. Córcoles, J. A. Smolin, J. M. Gambetta, J. M. Chow, and B. R. Johnson, "Demonstration of quantum advantage in machine learning," *NPJ Quantum Information*, vol. 3, no. 1, p. 16, April 2017.

[2] S. Bravyi, D. Gosset, and R. König, "Quantum advantage with shallow circuits," *Science*, vol. 362, no. 6412, pp. 308–311, October 2018.

[3] W. J. Yun, J. P. Kim, S. Jung, J.-H. Kim, and J. Kim, "Quantum multiagent actor-critic neural networks for internet-connected multirobot coordination in smart factory management," *IEEE Internet of Things Journal*, vol. 10, no. 11, pp. 9942–9952, June 2023.

[4] C. Park, W. J. Yun, J. P. Kim, T. K. Rodrigues, S. Park, S. Jung, and J. Kim, "Quantum multiagent actor–critic networks for cooperative mobile access in multi-UAV systems," *IEEE Internet of Things Journal*, vol. 10, no. 22, pp. 20 033–20 048, June 2023.

[5] H. Wang, Z. Li, J. Gu, Y. Ding, D. Z. Pan, and S. Han, "QOC: Quantum on-chip training with parameter shift and gradient pruning," in *Proc. IEEE/ACM Design Automation Conference*, San Francisco, CA, USA, June 2022, pp. 665–660.

[6] S. Y.-C. Chen, T.-C. Wei, C. Zhang, H. Yu, and S. Yoo, "Quantum convolutional neural networks for high energy physics data analysis," *Physical Review Research*, vol. 4, no. 1, p. 013231, March 2022.

[7] T. Hur, L. Kim, and D. K. Park, "Quantum convolutional neural network for classical data classification," *Quantum Machine Intelligence*, vol. 4, no. 1, p. 3, February 2022.

[8] J. Gou, B. Yu, S. J. Maybank, and D. Tao, "Knowledge distillation: A survey," *International Journal of Computer Vision*, vol. 129, no. 6, pp. 1789–1819, March 2021.

[9] M. Altaisky, "Quantum neural network," *arXiv preprint quant-ph/0107012*, July 2001.

[10] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," *Quantum Science and Technology*, vol. 4, no. 4, p. 043001, November 2019.

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

[12] R. Tombe and S. Viriri, "Effective processing of convolutional neural networks for computer vision: a tutorial and survey," *IETE Technical Review*, vol. 39, no. 1, pp. 49–62, September 2022.

[13] V. Rajesh, U. P. Naik *et al.*, "Quantum convolutional neural networks (QCNN) using deep learning for computer vision applications," in *Proc. Int. Conf. on Recent Trends on Electronics, Information, Communication & Technology (RTEICT)*, August 2021, pp. 728–734.