We thank the reviewers for their careful reading of the manuscript and their constructive suggestions.

Reviewer-1/3, Novelty of switching & comparing against Chimera: We compared Glyph against Chimera [14]. Chimera supports the switching between BFV and TFHE, while Glyph enables the switching between BGV and TFHE. **Chimera CANNOT support the switching between BGV and TFHE**. We selected BGV for two reasons. (1) Our baseline FHESGD [2] adopted BGV. (2) MultCPs and MultCCs of BGV are faster than those of BFV. We further demonstrated Glyph achieves faster privacy-preserving training speed than Chimera but obtains the same accuracy.

Reviewer-1/3, Novelty of transfer learning & comparing against EPIC: Glyph is the first work to use transfer learning to achieve fast non-interactive HE-based privacy-preserving CNN training. Based on transfer learning, EPIC replaces the last fully-connected layer of a neural network by a SVM and retrains the network by the same plaintext dataset. During an inference of EPIC, the first several layers are computed by the client, while the last layer of SVM is done by the server. EPIC depends on multi-party computation that exchanges huge amounts of data between the client and the server. Some users may not have such large network bandwidth. Moreover, EPIC CANNOT work with state-of-the-art CNNs. In contrast, Glyph first trains a CNN network model by a plaintext public dataset. And then, it homomorphically retrains the CNN model with a freshly initialized full-connected layer by an encrypted private dataset based on transfer learning. Except sending the encrypted input data, the training of Glyph does not involve the client.

Reviewer-1/2, Scalability, computing overhead and machines: We reported the training latency in Table 4. Compared to our baseline FHESGD [2], Glyph is more scalable, since it can support the training of deeper CNNs on larger datasets, e.g.., Skin-Cancer-MNIST. We will add the training latency on plaintext data in the next version of this manuscript. For data sizes (encrypted training data and key-switching keys), a BGV ciphertext with 60 slots is 256 KB. 60 Skin-Cancer-MNIST images cost $28 \times 28 \times 3 \times 8 \times 256KB$ = 4.6GB. The amortized size of each encrypted image is 76.6MB. One TFHE ciphertext with 1 slot is 2 KB. Each encrypted image occupies $28 \times 28 \times 3 \times 8 \times 2KB$ = 36.7 MB. Key-switching key samples have the size of 64 MB. Our baseline FHESGD uses a 2.30GHz Intel Xeon E5-2698v3 processor with two sockets and sixteen cores per socket. The machine has 250GB of main memory. And Glyph is tested on an Intel Xeon E7-8890v4 2.2GHz CPU with 256GB DRAM. The CPU also has two sockets, each of which owns 12 cores and supports 24 threads. Two machine configurations are similar. The peak main memory usage of Glyph is $\sim$150GB.

Reviewer-1, Performance improvement 69%$\sim$99%: Our Glyph-based CNN (BGV-TFHE) reduces the training latency by 69% over Chimera (BFV-TFHE) on the MNIST dataset. Compared to the FHESGD-based MLP, our Glyph-based MLP reduces the training latency by 97.4% on the MNIST dataset. We will update this number in our manuscript.

Reviewer-1, Bit-width of networks : We used 8-bit integers. We presented a 3-bit LUT in Section 3 as one example to explain the mechanism of TFHE-based activations.

Reviewer-2/3, Why not HEAAN?: Although HEAAN supports fixed-point numbers, we did NOT choose HEAAN for 3 reasons. (1) The training of state-of-the-art CNNs can be accurately done with only integers [R1]. (2) Our baseline FHESGD [2] uses BGV that supports only integers. (3) In order to support complex number, compared to BGV/BFV, HEAAN has only 50% batching slots, which degrades the speed of privacy-persevering training.

[R1] Wu, Shuang, et al. "Training and Inference with Integers in Deep Neural Networks." International Conference on Learning Representations. 2018.

Reviewer-2, Model poisoning and boarder impact: We used the same threat model as FHESGD [2]. In our threat model, Glyph aims to protect the privacy of clients, i.e., the input data and the output data are encrypted. We did NOT consider model poisoning in the threat model since this is a different security problem. We will consider this issue in our future work.

Reviewer-3, Why FHESGD is worse than Glyph, and the switching strategy: Both FHESGD and Glyph use BGV to compute linear layers. For activations, FHESGD uses BGV-based lookup tables, which is slow, as shown in Table 2. Glyph adopts TFHE to implement nonlinear activations, which is much faster, since TFHE can naturally support binary logic operations. Glyph uses BGV first, since the first layer is typically a linear layer. It switches to TFHE, since the following layer is an activation layer.