We thank all the reviewers for their valuable feedback and appreciating our contributions. Please find our response to each individual reviewer below.

**—— To Reviewer #1 ——**

**Empirically verify the equivalence between CNN and CNTK.** It would indeed be very interesting to empirically verify the equivalence. Note that the theoretical equivalence requires near-zero initialization, gradient flow (small learning rate), and a large number of channels. These require significant computation resource. We plan to conduct more thorough experiments along this line.

**Specific bound on $m$.** Our proof is based on the result in [Allen-zhu et al., 2018b], for which one needs a large polynomial like $m = \Omega(n^{24}L^{12})$. We believe such bound is not informative and can be significantly improved, so we did not specify the bound in the paper. Nevertheless we are happy to include it in the final version.

**LCN and CNN.** This is a good point. This equivalence still holds for CNTK-V but not for CNTK-GAP, because in CNTK-V we do not compute the cross-variance between patches at different positions but in CNTK-GAP we do (line 797). Thanks for the remark. We will add discussions in the final version.

**Notation in Lemma E.3.** Your explanation is correct. We will describe this more clearly. Thanks for pointing out!

**—— To Reviewer #5 ——**

**Performance on other datasets.** Thanks for the suggestion. We will conduct experiments on other datasets, including CIFAR100 and SVHN, in the final version of the paper, and open-source our implementation.

**Significance of depth in the performance of kernels.** We have not gone beyond 21 layers due to computation constraints, but the difference between the 21-layer kernel and the 11-layer kernel doesn't seem significant. It would be a very interesting future direction to investigate the effect of depth on the performance.

**Using NTK for neural architecture search.** One of the advantages of kernel methods is that they require **little computation** on a small dataset, which is a very appealing feature for architecture search. So we believe using the relevance between CNTKs and CNNs, we can develop more effective CNNs based on the performance of CNTKs on small datasets. We will stress this point more clearly in the final version.

**Actual computation time for kernels.** For 21-layer CNTK, we used 1000 GPU hours.

**Generalization to other losses.** Once we have the kernel matrix, we can also minimize the cross-entropy loss on the prediction space. The current result for the equivalence between NN and NTK cannot generalize to other losses though.

**Why is global average pooling so crucial?** This is a very interesting question worth further investigation. A possible explanation is that global average pooling may impose a certain data augmentation effect.

**—— To Reviewer #7 ——**

**Benefit of GPU implementation tricks & generalization to other activation.** The benefit is significant. We were able to compute the 21-layer CNTK on CIFAR-10 within 1000 GPU hours, while without these tricks this computation is simply infeasible (as mentioned in [Novak et al., 2019]). Note that another aspect that leads to the speed up is that we wrote native CUDA code. The efficient algorithm in Section I relies on homogeneity of the activation, so it can be generalized to Leaky ReLU. It is not clear how to deal with other activation functions.

**Experimental details.** We will describe the experimental details further in the full version. In particular: (1) # channels: The # channels is between 256 and 1024 in our experiments. All convolutional layers in a network have the same # channels. (2) Show a figure similar to Figure 6 in [Novak et al., 2019]? Thanks for the great suggestion. We will add this experiment in the final version of the paper.

**Strengthen Section 4.** The NTK formula for residual layers is not trivial to derive. It is possible to generalize to other loss functions (see response to Reviewer #5). We will open-source our implementation.

**—— To Reviewer #9 ——**

**Positivity of $\boldsymbol{H}^*$.** The positivity of $\boldsymbol{H}^*$ is proved in [Du et al., 2019] for two-layer ReLU NN and in [Du et al., 2018b] for multi-layer NN with smooth activation. Since our setting is very similar to these two papers' settings, we believe $H^*$ is positive definite in general. Furthermore, experimentally we find that $H^*$ is always positive definite. We will add this discuss to the final version.

**Proof of convergence to $f_{ntk}$.** Thanks for the suggestion, and we will add a proof to the final version. It can be derived by looking at the dynamics of $f(\boldsymbol{\theta}(t), \boldsymbol{x})$ for a given test input $\boldsymbol{x}$.