Abstract—Low latency, high throughput inference on Convolutional Neural Networks (CNNs) remains a challenge, especially for applications requiring large input or large kernel sizes. 4F optics provides a solution to accelerate CNNs by converting convolutions into Fourier-domain point-wise multiplications that are computationally ‘free’ in optical domain. However, existing 4F CNN systems suffer from the all-positive sensor readout issue which makes the implementation of a multi-channel, multi-layer CNN not scalable or even impractical. In this paper we propose a simple channel tiling scheme for 4F CNN systems that utilizes the high resolution of 4F system to perform channel summation inherently in optical domain before sensor detection, so the outputs of different channels can be correctly accumulated. Compared to state of the art, channel tiling gives similar accuracy, significantly better robustness to sensing quantization (33% improvement in required sensing precision) error and noise (10dB reduction in tolerable sensing noise), 0.5X total filters required, (33% improvement in required sensing precision) error and noise (10dB reduction in tolerable sensing noise), 0.5X total filters reduced, and an output sensor. Fourier transform, point-wise multiplication and inverse Fourier transform in 4F system is essentially constant time (the time of flight of light). Earliest optical correlators based on the 4F theory date back to the 1960s. Recently advancements in CNNs have renewed interest in 4F systems to speed-up the ‘expensive’ convolution process. Fast SLMs and fast cameras still remain a challenge despite substantial improvements for optical 4F computing.

In this paper, we focus on improving the accuracy and performance for high-speed, high-resolution optical 4F computing systems for neural network acceleration by proposing a simple tiling method to accumulate the convolution results of all channels of a filter inherently in optical domain, before the non-linearity applied by photodetectors/cameras. The main contributions of this work are summarized as follows.

- We propose a channel tiling approach which allows filters to have negative weights, an implicit non-linear activation by the camera/photodetector and optical domain accumulation of channels. We further combine channel tiling with input or kernel tiling to fully leverage available optical parallelism and achieves 10-50X utilization improvements compares with other approaches.
- Our results indicate that channel tiling dramatically improves inference accuracy (36% points on CIFAR10 dataset running VGG16 network) compared to alternative tiling approaches.
- We show that channel tiling can be done without any changes to optical hardware itself unlike optical alternatives to preserve sign and it can have (at least) 2X throughput advantage over the recently proposed pseudo-negative approach.
- We show that channel tiling can reduce the photodetector/camera resolution requirements by orders of magnitude alleviating the main bottleneck of 4F computing systems.
- We further show that channel tiling is much more robust to camera’s quantization error and photo-detection noise (1% vs. 15% accuracy drop compared to state of the art with 8-bit camera and 20dB SNR). This allows for channel tiling to support much faster and lower bit-depth cameras.

I. Introduction

Convolutional neural networks (CNNs) have proliferated in image classification and computer vision. Over the years, CNNs have become increasingly larger and deeper [1], making it harder to deploy such networks on traditional electronic machines due to convolution’s high computation complexity. Although many efforts have been made on both algorithms and hardware to speed up the convolution process, inference of large CNNs, especially those with high-resolution inputs, is still computationally prohibitive. There is renewed and growing interest in optical/photonics computation hardware for CNN inference acceleration, due to their low compute latency (essentially time of flight of light) and the potential to support large parallelism [2].

Current photonic CNN accelerators can be roughly classified into two main categories: (1) on-chip implementations using photonic devices including Mach-Zehnder Interferometers [3] and micro-ring resonators [4][5][6], and (2) free-space 4F system, using spatial light modulators (SLMs) and phase masks [7][8][9].

On-chip photonic implementations usually have a high clock speed (in GHz range), but the amount of parallelism is far less than the 4F system. The performance of on-chip photonic implementations is usually bounded by digital-to-analog converters’ relatively low operation frequency, and they might not be efficient when dealing with high-resolution inputs. Scaling is another issue of on-chip photonic implementations since the photonic hardware scale significantly slower than semiconductors.

In contrast, free-space 4F systems offer massive parallelism due to the high resolution of SLMs and phase masks, as well as efficient convolution computation using the well-known 4F theory, which states that convolution in space domain is equivalent to point-wise multiplication in Fourier domain. The 4F system can be implemented using two Fourier lenses, an input source, a device for multiplication and an output sensor. Fourier transform, point-wise multiplication and inverse Fourier transform in 4F system is essentially constant time (the time of flight of light). Earliest optical correlators based on the 4F theory date back to the 1960s. Recently advancements in CNNs have renewed interest in 4F systems to speed-up the ‘expensive’ convolution process. Fast SLMs and fast cameras still remain a challenge despite substantial improvements for optical 4F computing.

II. A Primer on 4F Optical Computing Systems

A. An Overview of Optical 4F Computing System

4F optical system usually consists of 4 parts: input source, two Fourier lenses, kernel light modulator and output sensor. Figure 1 shows the high-level system diagram of the 4F optical computing engine. The input source usually includes a laser emitter to generate coherent light and an SLM (Spatial Light Modulator) to encode the input by modifying the light intensity. Then the encoded light passes...
through the first Fourier lens to perform Fourier transform. The Fourier transformed light signal is projected onto the kernel light modulator (Fourier plane), where it actively or passively modulates the light to conduct point-wise multiplication in Fourier domain. Afterward, the light signal that contains multiplication results passes through the second Fourier lens for inverse Fourier transform. Then, it is captured by a high-speed camera and enters into electronic domain.

Most published works\cite{8,9,10} use phase masks as the kernel light modulator to passively modulate the light in Fourier plane. SLM and DMD (Digital Micromirror Device, a special type of SLM) can also be used as kernel light modulator to actively modulate the light, hence providing programmability \cite{11}. For passive approaches, multiplication can be implemented with zero latency and power, but with no flexibility since the weights are fixed. In contrast, active approaches have more flexibility (weights can be modified) and can hence easily scale to large multi-layer networks, albeit at the cost of latency and power overheads.

### B. Leveraging Massive Optical Parallelism Using Computation Tiling

![Fig. 1. Illustration of optical 4F system.](image1)

The input tiling can be implemented in a similar way, but in this case inputs are tiled and only a single kernel is used. The tiled input can be directly loaded onto input SLM, since the Fourier transform of input is carried out by the Fourier lens.

![Fig. 2. Illustration of kernel tiling method. (a): Padded input block and tiled kernel block. (b): Convolution visualization of kernel k1 with the input. (c): Effect of zero padding. The padding between kernels ensures that the input is not overlapped with two kernels at same time.](image2)

To demonstrate the 4F system’s advantage of ‘free Fourier transform and multiplication’, we compare a 4F system’s (4K SLMs and 4K camera) convolution performance against a GPU (Nvidia P100) implementation of Fourier transform based convolution using PyTorch’s FP16 FFT. Figure 3 shows the comparison with 4F systems for convolution time versus different input sizes for a 5x5 space domain kernel. GPU is fully batched to utilize their parallel processing power while for 4F system results with full tiling and no tiling are all included. The result suggests that the 4F systems can outperform GPU by several orders of magnitude if tiling is used (a full optical computing system architecture to harness this benefit is part of our ongoing work).

![Fig. 3. Single convolution time versus different input sizes for GPU FFT implementation and 4F systems operating at different frequencies](image3)

### C. The Positive-only Photodetection Challenge

For any optic/photonic CNN hardware, including the 4F system, the convolution/computation output needs to be converted to digital domain by photodetectors or cameras, which apply a square function on the results by measuring the intensity. Since the camera/photodetector readouts are all positive, the weights need to be all-positive to make the individual convolution result valid (can use a square root function to retrieve the original result). The positive-only weight restriction can drop inference accuracy to unacceptable levels or make networks fail to converge. Existing works deal with this limitation either optically \cite{12} or computationally \cite{8}.

Detecting the phase information using dedicated optical hardware can remove the positive weight restriction. \cite{13} uses balanced photodetectors with MZM (Mach-Zehnder modulator) to detect the intensity and phase for photonic neuromorphic networks, though this technique cannot use in conventional cameras and may require precise alignment. \cite{14} designs an optical ReLU module for the free-space 4F system, using SLMs and custom-built circuits. Polarization interferometry is used with reference beam to detect the sign of each pixel and then feed the result back to SLM to let corresponding light pass through or deflect away. This method may be used to purely...
detect the sign information of the outputs detected by camera for a conventional 4F system, at the cost of extra specialized hardware (at least double the number of photodetectors required and extra reference beam) and the complexity of integrating this method with conventional high-speed cameras (original design is based on SLMs).

[8] introduced a ‘Pseudo-negative’ method which can address the positive weight restriction without additional hardware, at the cost of doubling the number of filters and consequent computation overheads.

The main idea behind this method is using all positive weights for filter to ensure valid results, but labelling half of the filter as positive and the other half as negative. After the sensor readout the results of positive filters is subtracted by the results of negative filters to form the final convolution results. The advantage of this approach is the convolution results can be considered as identical to normal convolution due to the linearity of convolution (by using positive weights and taking square root after readout the sensor non-linearity can be removed). However, the number of filters required is doubled, which means double the weight memory and half the throughput to replicate the original network.

D. Performance Gap Between Camera and Light Modulators

For high speed 4F accelerators, usually the output sensing will be the performance bottleneck due to the performance gap between high speed SLM and high speed camera. In such systems the inputs are usually generated by high-resolution, high-speed SLM [15][13][12][16]. While the multiplication is carried out using either another SLM or phase mask. The resolution of commercial SLM can scale up to 4K with a nominal operating frequency of 20-30 KHz[12] with several research SLMs going to MHz regime [16]. The operating frequency of such SLM is way higher than the state-of-art commercial 4K high-speed camera which operates at roughly 1 KHz[17][18]. The main performance bottleneck of high-speed-cameras is the readout time, which scales with the camera’s resolution. The operating frequency of high-speed cameras can match or exceed SLM if the frequency of high-speed cameras can match or exceed SLM if the resolution goes down, as there are high speed framing cameras that can operate in MHz or even GHz range [19][20]. But it also means the SLM’s high resolution cannot be fully utilized since for input/kernel tiling the camera should have same resolution as the input SLM.

III. Channel Tiling for Optical Compute Parallelism

Existing tiling approaches in optical computation overlook the fact that nearly all neural network layers in modern CNNs have multiple input channels giving another axis along which to parallelize computation. To address the all-positive readout challenge, as well as eliminating the performance gap between SLM and high speed cameras, we propose channel tiling (and mixed tiling) that can sum all channels inherently in optical domain and with significantly lower output resolution requirement compared to other tiling methods. The channel tiling method tiles the channels of both input and kernel and produces a single output, with convolution results for all channels summed (essentially implementing the multi-channel convolution optically).

All tiling approaches are based on the assumption that the 4F system has the ability to multiply the inputs with complex-valued weights in Fourier domain, which can be achieved by using a phase mask, SLM or DMD pairs. [21] Thus, any real-valued filter can be transformed into Fourier domain and multiplied with inputs without loss. The 4F system can then essentially be thought of as a black box convolution engine that can take inputs and filters each of size up to its resolution. In the analysis we assume SLM with resolution $D \times D$ is used for implementing point-wise multiplication in Fourier domain, but the method generalizes to other devices/components as well.

Tiling is done in space domain and based on cross-correlation, which is more commonly used in the field of deep learning. To apply on a 4F system which performs convolution, the kernel needs to be flipped before doing Fourier Transform. There are two common convolution modes (‘same’ and ‘valid’) and they require slightly different tiling setup. For the ‘same’ mode where output has the same size as input, padding is essential during tiling in order to generate correct results. For ‘valid’ mode where the size of output is $(M - N + 1) \times (M - N + 1)$ (see table I), padding is not necessary since the correct result can be extracted by down-sampling the output. In this section, all analysis is based on ‘same’ mode which includes zero padding, the tiling approach holds for ‘valid’ mode except that padding is not required. For simplicity of analysis, actual SLM resolution is not taken into account for all tiling schemes except for mixed tiling. We validated all our analytical models of tiling (presented below) with computational experiments in a Python+Scipy+Numpy setup to confirm that tiling has no impact on the correctness of the convolution results.

A. Channel Tiling Operation

Consider the case where $N_t$ input channels with size $M \times M$ convolve with $N_t$ corresponding kernels with size $N \times N$. Like other schemes, the channels for both input and kernel need to be padded into blocks with size $(M + N - 1) \times (M + N - 1)$, to avoid overlap between single kernel with multiple input channels. The kernel needs to be padded to same size as well. Then both the input channel blocks and kernel blocks are tiled in same order to form two large blocks with size $\left[\sqrt{N_t} \right] (M + N - 1) \times \left[\sqrt{N_t} \right] (M + N - 1)$, as shown in fig [3](a). For simplicity, denote the size of tiled input and kernel blocks as $M_t \times M_t$, where $M_t = \left[\sqrt{N_t} \right] (M + N - 1)$. The formula for each output activation is

$$y_{i,j} = \sum_{a=0}^{M_t-1} \sum_{b=0}^{M_t-1} x(a+i,b+j) \times k(a,b)$$

where $x$ is a block with size $(2M_t-1) \times (2M_t-1)$, formed by circular padding of the tiled input block, which is inherently generated by Fourier transform. $k$ is the tiled kernel block. Both $y$ and $k$ have size $M_t \times M_t$.

Note that when $i, j$ are in range of $\frac{M_t-M}{2}$ to $\frac{M_t-M}{2}$, the convolution result $y(i,j)$ is equal to the summation of all convolution result of individual input channels with their corresponding kernels, as shown in fig [3](b). When $i, j$ is in this range, each individual input channel on the tiled input block aligns with its corresponding kernel on the tiled kernel block, hence the output activation is the summation of convolution results of all channels, which equals to the output activation of the channel-wise convolution. When $i, j$ are outside of this region, as the case in fig [3](c), the result is not valid, e.g., input channels are convolved with kernels for other channels. Those invalid results make the $(M_t-1)/2$ extra zero padding of tiled input and kernel due to inherent circular padding of Fourier transform unnecessary since circular padding only corrupts the results of invalid region. Therefore in this scheme only the center $M \times M$ region of the whole $M_t \times M_t$ result is equivalent to the multi-channel convolution result of all input channels and should be extracted as the final result.

The output of this tiling scheme $y$ is a single block with size $\left[\sqrt{N_t} \right] (M + N - 1) \times \left[\sqrt{N_t} \right] (M + N - 1)$ and only the center $M \times M$ region is valid. We can extract the valid region by directly selecting $y(i,j)$ in range of $\frac{M_t-M}{2}$ to $\frac{M_t-M}{2}$, the output is $M_t \times M$. Fig [3]e visualizes the output format. Since only the center $M \times M$ part is used as the output, the camera’s resolution requirement is massively
reduced to the size of a single input. This property can significantly reduce output bandwidth and improve camera readout time.

We propose this novel channel tiling scheme to carry out the channel summation inherently in optical domain, so that it won’t be affected by the camera’s non-linearity. By doing so the space domain kernel values no longer need to be positive only, the 4F system can be modeled by using absolute value or square function as activation function with unconstrained kernels during training, which is not possible in other schemes.

B. Mixed Tiling

Free-space 4F system can support high resolution (up to 4K), hence a large number of blocks can be tiled for inputs with small sizes. For some neural network structures, SLMs/phase masks cannot be fully tiled hence causing under-utilization. Taking Alexnet as an example, for systems that support resolution higher than 1K, the system is far from fully utilized for any layer except the first one if channels or kernels are tiled alone. To address this issue, we propose a mixed tiling scheme which combines channel tiling and kernel tiling to improve the SLM utilization, while still preserving the ability of channel tiling to carry out channel summation inherently in optic domain.

Mixed tiling scheme is essentially applying two tiling schemes sequentially. The first step is applying channel tiling to tile channels of inputs and kernels across a larger block. Then the tiled kernel blocks are further tiled across the kernel SLM using kernel tiling method, but this time zero-padding of individual tiled kernel blocks are not necessary. The output is the convolution result of a multi-channel input against multiple kernels with all channel results accumulated. By doing so the SLM utilization can be vastly improved compares with channel tiling scheme while channel summation capability is not affected.

To maximize tiling efficiency, in step one channels should be tiled horizontally into rows to better utilize the SLM resolution. The size of padded individual channel blocks are \((M + N - 1) \times (M + N - 1)\), the maximum number of such blocks that can be tiled on a single SLM is denoted as \(T\), which equals to \(\frac{\sqrt{N}}{D}\). The tiled block \(B\) has shape \(D, (M + N - 1) \times \left\lceil \frac{\sqrt{N}}{D} \right\rceil\). The condition for mixed tiling is \(N_c < \frac{T}{D^2}\), where \(N_c\) is total number of blocks needs to be tiled for the first tiling scheme.

If the above condition is true, \(B\) can be further tiled across all available SLM area from top to bottom. Zero padding inside \(B\) blocks is not necessary, the padding within each single \((M + N - 1) \times (M + N - 1)\) is enough to generate correct result, since only the invalid region (as discussed for channel tiling) will be corrupted by overlapping of multiple kernels.

The output format is a combination of the two applied tiling schemes. To extract the valid outputs, first split the raw output into results of individual kernels and then extract the valid region inside each kernel’s result. Compared with other tiling schemes, the number of pixels required for detection is still low due to the combined channel tiling. However, since for mixed tiling the valid outputs are located at different regions, modification of detection device is necessary to utilize the low output pixel count property in the mixed tiling case, for example a specialized controllable photodetector array. Without this, mixed tiling still delivers the required output bandwidth reduction from the camera but without improvement in resolution requirements.

C. Tiling Efficiency Analysis and Optimization

The tiling efficiency, or how much the (fixed size) SLM is utilized, is determined by several factors including number of inputs/channels/kernels, their sizes and whether padding is required or not.

For the case where only a single tiling scheme is used, the number of blocks that needs to be tiled \(N_{i,c,k}\) should be larger than \(T\). For best performance, \(N_{i,c,k}\) should be multiples of \(T\) so that for each SLM update all pixels are fully utilised. The utilization rate is

\[
U = \frac{M^2 \times N_{i,c,k}}{D^2 \times \left\lceil \frac{\sqrt{N}}{D} \right\rceil}
\]

The numerator in equation is the total number of pixels for all blocks that require tiling and the denominator is total number of pixels actually used, which is a multiple of \(D^2\). When \(N_{i,c,k}\) is small compares with \(T\), mixed tiling scheme should be adopted for optimal performance.

For mixed tiling scheme, two tiling schemes are combined to improve SLM utilization. Again we use \(B\) to denote the tiled blocks for the first scheme, \(N_1\) and \(N_2\) to denote number of blocks needs to be tiled for the first and second tiling scheme respectively (i.e. number of channels and kernels). Since we assume SLMs are square shaped, the number of individual blocks can be tiled in one row or column is \(\sqrt{T}\). Similarly, the utilization for mixed tiling is

\[
U = \frac{M^2 \times N_1 \times N_2}{D^2 \times \left\lceil \frac{\sqrt{T}}{N_1} \right\rceil}
\]

Where \(T_B\) is the total number of tiled block \(B\) that can be further tiled across a single SLM, calculated by

\[
T_B = \left\lceil \frac{\sqrt{T}}{N_1} \right\rceil
\]

D. Hardware Requirement Analysis

Table II shows the resolution requirement for different tiling schemes. Input and kernel tiling are analogous to broadcast where the un-tiled plane is broadcast to the tiled plane. For kernel tiling (used in along with pseudo-negative method), the input plane is not tiled hence the input plane resolution is same as a single input. For input tiling, kernel plane is not tiled, but it still require full resolution since kernel plane is in Fourier domain and must have the same resolution as input to perform point-wise multiplication. Channel-tiling and mixed tiling require full resolution of input and kernel plane, while having very low resolution requirement for output. Modern CNN models usually have input image resolution lower than 300×300 and is further downsamples in subsequent layers. Consider a 4K system setup with 300×300 input resolution, channel tiling will reduce the output resolution requirement by 18x compared with input/kernel tiling when fully tiled. Such a massive
reduction in output resolution will eliminate the performance gap between output sensing unit and input/kernel SLM. Even for mixed tiling the output resolution requirement is \(N_c\) times less than input/kernel tiling, where \(N_c\) is the number of input channels in a layer and is usually between 32 to 512 depends on exact network structure and layer.

| Tiling schemes    | Input resolution | Kernel resolution | Output resolution |
|-------------------|------------------|-------------------|-------------------|
| None              | \(M^2\)          | \(M^2\)           | \(M^2\)           |
| Input tiling      | \(D^2\)          | \(D^2\)           | \(D^2\)           |
| Kernel tiling     | \(M^2\)          | \(D^2\)           | \(D^2\)           |
| Channel tiling    | \(D^2\)          | \(D^2\)           | \(M^2\)           |
| Mixed tiling      | \(D^2\)          | \(D^2\)           | \(D^2\)           |

**TABLE II** COMPARISON OF THE RESOLUTION REQUIREMENT FOR DIFFERENT TILING SCHEMES, ASSUMING ALL CASES ARE FULLY TILED.

E. Impact of Camera Bit-Depth and Sensing Noise

For free-space 4F system, high-speed camera is usually used as the output detection device and it adds two kinds of errors to the system, namely the quantization error and random noise. Getting high bit-depth (i.e., precision), high resolution, high speed and low noise is tough challenge for cameras and photodetectors. Most high speed cameras with reasonable resolution are limited to 8 or 12 bit precision (e.g., see [18][23][17]).

Due to limited precision or bit-depth of cameras, all outputs need to be quantized to 8-bit (or 12-bit) fixed point. While CNNs usually do not require a high precision hence the effect of activation quantization on accuracy may be small, input and kernel tiling quantize each channel as channel summation happens electronically after optical sensing (i.e., the partial sums themselves are quantized and not just the final activation value). For channel tiling channel accumulation is carried out in optical domain at full precision and only the accumulated results are quantized to 8-bit.

Similarly, channel tiling is less susceptible to sensing noise in the camera. Sensing noise can be especially limiting for fast, high resolution cameras needed for optical computing. Random sensing noise increases error in every channel in input/kernel tiling unlike channel tiling. The error scales with the number of channels so it impacts more for large networks. Furthermore, camera SNR (Signal to Noise ratio) scales with the photon flux (or number of photons captured by a pixel)\(^{[24]}\)[25][26]. Intuitively, if the physical size of camera’s sensor is fixed, then the higher resolution it supports (more pixels), the less photons individual pixels will receive. As discussed, channel tiling (and mixed tiling) requires significantly less camera resolution compared to other tiling methods, which means it can have higher camera SNR than other methods.

Channel tiling inference accuracy is expected to suffer much less from camera quantization and noise. Results illustrating his benefit of channel tiling are discussed in Section IV-C.

IV. Experimental Results and Discussion

In this section, we compare the proposed channel/mixed tiling approaches with other state-of-the-art 4F optics approaches, both in terms of performance and inference accuracy. We use popular datasets FashionMNIST, CIFAR10, SVHN running appropriately scaled VGG16-style neural network for accuracy comparisons.

A. Comparing Performance of Tiling Approaches

To compare the tiling efficiency of different tiling methods, we calculate the average system utilization for kernel, channel and mixed tiling over two network setups, Alexnet with ImageNet and VGG-16\(^{[27]}\) with CIFAR-10. The system resolution is assumed to be 1080\(\times\)1080 and system frequency is assumed to be 50KHz. The average utilization is defined as \(\frac{N_{\text{pixelcount}}}{N_{\text{updates}}}\), where \(N_{\text{pixelcount}}\) is the total pixel count of all kernels and channels for the network, \(N_{\text{updates}}\) is the total kernel SLM updates for the network and \(R_{\text{SLM}}\) is the SLM resolution.

Table [III] shows the system utilization and throughput for the two cases, which shows kernel tiling and channel tiling suffer from severe under-utilization for both cases. The utilization for kernel/channel tiling on VGG-16/CIFAR10 is barely 1%, due to CIFAR10’s low input resolution. Mixed tiling achieves around 90% utilization for both cases, which is roughly 10X and 50X improvement over kernel/channel tiling methods in throughput. On a 4K system, this improvement would be even higher. Furthermore, the required output camera resolution (and hence bandwidth) for the mixed tiling approach is going to be 3X less than the SLM size (since minimum number of channels across the network is 3).

| AlexNet (ImageNet) | VGG-16 (CIFAR10) |
|--------------------|------------------|
| **Tiling method**  | **Utilization**   | **Throughput** |
| Kernel tiling      | 9.7%             | 36.5           |
| Channel tiling     | 8.0%             | 43.3           |
| Mixed tiling       | 92.1%            | 116.7          | **TABLE III** SYSTEM UTILIZATION AND THROUGHPUT COMPARISON FOR DIFFERENT TILING METHODS ON TWO NETWORK SETUPS, THROUGHPUT IS CALCULATED BY NUMBER OF INPUT FRAMES (INCLUDING ALL CONVOLUTION LAYERS IN THE NETWORK) CAN BE PROCESSED PER SECOND ON A 50KHz 1K RESOLUTION 4F SYSTEM.

B. Impact of Tiling Approaches on Inference Accuracy

As discussed previously, different tiling schemes impose different restrictions on the network: input/kernel tiling places a square function on each channel’s convolution result before summation while channel/mixed tiling restricts the activation function to be absolute value function (taking a square root after camera readout). The effect of these restrictions on network-level accuracy is reported in Table [IV] evaluating three datasets trained with VGG-16 like model. The camera is assumed to have unlimited precision (addressed in the next section). All-positive kernels are not used in input/kernel Tiling method as it effectively nullifies the purpose of activation and make deep CNNs like VGG-16 extremely hard to train properly since there are (almost always) multiple input channels. Optical compensation approaches (e.g., the sign detection proposed in [2]) and pseudo-negative [8], in absence of any optical non-idealities, should give same accuracy as standard convolution.

The results clearly show that input or kernel tiling are unacceptably inaccurate for anything but simplest of classification tasks. Our proposed channel and mixed tiling approaches lose <1% accuracy compared to unconstrained electronic implementations or pseudo-negative approach (which uses twice as many filters). This small gap in accuracy can be bridged in future by better optical non-linearities (an active area of research\(^{[28][29]}\), improved training methods (e.g., different regularization terms during training to incentivize positive kernel weights) or combining with the pseudo-negative approach\(^{[8]}\), all are part of our ongoing work but not explored further in this paper.

C. Impact of Camera Limitations on Inference Accuracy

The previous accuracy results do not consider hardware precision and noise. To simulate quantization error, square function is applied to the convolution results for simulating the intensity measurement and then the results are quantized to scaled 8-bit and 12-bit format (256/4096 uniform intervals). Square root is taken on the quantized...
results to get the absolute value. We simulate the sensing noise using white Gaussian noise with average SNR at 15dB, 20dB and 30dB [30]. Figure 5 shows the CIFAR-10 and Fashion MNIST classification accuracy using VGG-16 for pseudo-negative with kernel tiling and proposed channel/mixed tiling, with camera quantization error and different level of random noise. The results clearly show that channel/mixed tiling is far more robust to both camera precision and sensing noise due to its error-free channel summation. Pseudo-negative approach requires at least 12-bit camera precision as opposed to 8-bit for channel/mixed tiling to remain within 5% accuracy drop from full precision. Interestingly, once camera bit-depth is taken into account, channel tiling always has higher accuracy than pseudo-negative approach despite it being somewhat more restrictive.

For cases with random noise, the accuracy of proposed channel/mixed tiling method is higher than pseudo-negative method for almost all cases, 30dB or 20dB SNR is good enough for channel tiling accuracy to be within 5% of noiseless case while other tiling approaches need at least 30dB SNR. Moreover, the achievable SNR for channel/mixed tiling should be higher than kernel tiling since it requires lower resolution cameras.

Altogether, our results indicate that the proposed channel tiling approach can reduce required camera precision by 33% (8-bit vs 12-bit) and improve noise tolerance by 10dB. Though we do not explore it here, such relaxation can substantially improve speed, energy and cost of sensing in 4F computing systems.

V. Conclusions

In this work, we propose the channel tiling and mixed tiling method for optical 4F systems to boost the performance and accuracy, without extra hardware or computation. By utilizing the properties of convolution and 4F system, channel tiling and mixed tiling make 4F system able to accumulate all channel’s convolution result in optical domain and use camera’s non-linearity as activation function. Compared to recent pseudo-negative approach with kernel tiling [8], our method gives similar accuracy (<1% difference on three datasets), significantly better robustness to sensing quantization (33% improvement in required sensing precision) error and noise (10dB reduction in tolerable sensing noise), 0.5X total filters required, 10-50X+ throughput improvement and as much as 3X reduction in required output camera resolution/bandwidth. The proposed channel tiling methods provide a simple way to fully utilize the massive parallelism inherent in optical 4F computing system to accelerate CNNs. Our ongoing work addresses building a physical free-space 4F computing prototype and modeling optical non-idealities.

References

[1] W. Rawat and Z. Wang. “Deep convolutional neural networks for image classification: a comprehensive review”. In: Neurocomputing 291 (2017), pp. 2352-2349.

[2] L. De Marinis et al. “Photonic Neural Networks: A Survey”. In: IEEE Access 7 (2019), pp. 175827–175841.

[3] H. Bagherian et al. “On-chip optical convolutional neural networks”. In: arXiv preprint arXiv:1808.03030 (2018).

[4] A. Mehrabian et al. “PCNNA: a photonic convolutional neural network accelerator”. In: 2018 31st IEEE International System-on-Chip Conference (SOCC). IEEE, 2018, pp. 169–173.

[5] V. Bangati et al. “Digital electronics and analog photonics for convolutional neural networks (DEAP-CNNs)”. In: IEEE Journal of Selected Topics in Quantum Electronics 26.1 (2019), pp. 1–13.

[6] M. A. Nalcioglu et al. “Photonic multiply-accumulate operations for neural networks”. In: IEEE Journal of Selected Topics in Quantum Electronics 26.1 (2019), pp. 1–13.

[7] S. Colburn et al. “Optical front-end for a convolutional neural network”. In: Applied optics 58.12 (2019), pp. 3179–3186.

[8] J. Chang et al. “Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification”. In: Scientific reports 8.1 (2018), pp. 1–10.

[9] K. H. Wagner and S. McComb. “Optical rectifying linear units for back-propagation learning in a deep holographic convolutional neural network”. In: IEEE Journal of Selected Topics in Quantum Electronics 26.1 (2019), pp. 1–18.

[10] X. Lin et al. “All-optical machine learning using diffractive deep neural networks”. In: Science 361.6406 (2018), pp. 1004–1008.

[11] Y. Guo et al. “All-optical neural network with nonlinear activation functions”. In: Optica 6.9 (2019), pp. 1132–1137.

[12] Texas Instruments. DLP4701E datasheet. [online] https://www.ti.com/document-viewer/DLP4701EDatasheet.

[13] Texas Instruments. DLP9000X datasheet. [online] https://www.ti.com/document-viewer/DLP9000XDatasheet.

[14] A. N. Tait et al. “Neuromorphic photonic networks using silicon photonic weight banks”. In: Scientific reports 7.1 (2017), pp. 1–10.

[15] HOLOEYE. GAE2-4K. [online] http://holoh Ye: com/gae -4k. phase- only- spatial- light- modulator.

[16] J. L. Schmid et al. “High-speed one-dimensional spatial light modulator for Laser Direct Imaging and other patterning applications”. In: MOEMS and Miniaturized Systems XIII. Ed. by W. Piyawattanametha and Y.-H. Park. Vol. 8977. International Society for Optics and Photonics. SPIE, 2014, pp. 163-176.

[17] Phantom. Flex4k Camera. [online] https://www.phantomhighspeed.com/products/cameras/4kmegaflex4k.

[18] Phantom. S990 Camera. [online] https://www.phantomhighspeed.com/products/cameras/s990/machinevision/6909.

[19] Specialized Imaging. Krauma7M Camera. [online] https://www.specialised-imaging.com/products/video-camera/krauma7m.

[20] Specialized Imaging. S950 Camera. [online] https://www.specialised-imaging.com/products/frameing-cameras/s950.

[21] J. W. Goodman. Introduction to Fourier optics. Roberts and Company Publishers, 2005.

[22] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “Imagenet classification with deep convolutional neural networks”. In: Advances in neural information processing systems. 2012, pp. 1097–1105.
G. Healey and R. Kondepudy. “CCD camera calibration and noise estimation”. In: Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Los Alamitos, CA, USA: IEEE Computer Society, June 1992, pp. 90, 91, 92, 93, 94, 95.

G. Healey and R. V. Kondepudy. “Modeling and calibrating CCD cameras for illumination-insensitive machine vision”. In: Optics, Illumination, and Image Sensing for Machine Vision VI. Ed. by D. J. Svetkoff. Vol. 1614. International Society for Optics and Photonics. SPIE, 1992, pp. 121–132.

M. Konnik and J. Welsh. High-level numerical simulations of noise in CCD and CMOS photodetectors: review and tutorial. 2014.

K. Simonyan and A. Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: arXiv preprint arXiv:1409.1556 (2014).

I. A. D. Williamson et al. “Reprogrammable Electro-Optic Nonlinear Activation Functions for Optical Neural Networks”. In: IEEE Journal of Selected Topics in Quantum Electronics 26.1 (2020), pp. 1–12.

J. George et al. “Photonic Neural Network Nonlinear Activation Functions by Electro-Optic Absorption Modulators”. In: Frontiers in Optics / Laser Science. Optical Society of America, 2018, JW3A.123.

W. Huang and Z. Xie. “Characteristics and performance of image sensor communication”. In: IEEE Photonics Journal 9.2 (2017), pp. 1–19.