function takes up most of the GPU resources (e.g., shared memory, registers, etc.). In that case, a different CUDA function will only be scheduled after all the thread blocks in the previous CUDA function finished execution [3]. Therefore, on GPUs with few SMs, this is less likely to be an issue as the GPU's resources are already saturated due to the large computational demand of DNN inference.

Bias Extraction in INT8 Implementations We demonstrated the extraction of biases from convolutional layers for FP16 implementations but not INT8 implementations. These implementations only use INT8 for the weights but not for the bias. The bias in these implementations has FP32 data type, which is beyond our computing capabilities to extract.

6.4 Mitigation

Traditional ways to contain electromagnetic emanation, such as proper shielding or introducing noise to decrease the Signal-to-Noise ratio, could alleviate the problem [42]. Specifically against parameter extraction, one of the possible countermeasures, which is also mentioned in the CSI-NN paper [13], is shuffling [61] the order of multiplications in the layers, which can make it significantly harder for an adversary to recover the weights. Additionally, masking [19, 48] can also decouple the side-channel measurements and the processed data. However, this comes at the price of execution speed, which might not be desired in real-time systems. Specifically for convolution, the registers containing the results of the partial sums can be initialized with the bias of the kernel instead of initializing them with zeros. This would prompt an adversary to mount a CEMA attack where the correct $b + w_1$ pair has to be recovered first. The complexity of this attack would be 32 bits due to 16 bits of complexity for the weight and bias separately in the FP16 case. However, in the INT8 case, the bias is a single-precision float, so it cannot be used to initialize the accumulator registers.

Author	Platform	Clock (MHz)	Side channel	Parameter datatype
Batina, et al. [13]	MCU	20, 84	EM	FP32
Dubet, et al. [21]	FPGA	24	Power	Binary
Yoshida, et al. [64]	FPGA	25	Power	INT8
Regazzoni, et al. [53]	FPGA	N/A[8]	EM	Binary
Yli-Mäyry, et al. [63]	FPGA	N/A[8]	EM	Binary
Li, et al. [37]	FPGA	25	Power	INT8
Joud, et al. [31]	MCU	100	EM	FP32
Gongye et al. [25]	FPGA	320	EM	INT8
BarraCUDA	**GPU**	**625**, **920**	EM	INT8, FP16

Table 3: Comparison with related work.

6.5 Related Work

To the best of our knowledge, no previous work has been able to extract the parameters of neural networks on GPU using physical side-channel. Previous works have demonstrated parameter extraction on microcontrollers and FPGAs using power or EM side channel, as shown in Table 3. In addition, these attacks were performed on neural networks with binary parameters [21, 53, 63], 8-bit parameters [13, 25, 37, 64] or 32-bit parameters [13, 31]. Our work demonstrates parameter extraction of 8- and 16-bit parameters. Furthermore, our work presents a CEMA attack on weights where the number of cores and the clock frequency at these cores operate are significantly larger than in related works. The large number of cores, with almost 1GHz clock frequency, presents a challenge in both the measurement and attack stages. Given that GPUs are the backbone of AI, it is of utmost importance to assess the resilience of GPU accelerated workloads against weight extraction attacks, a task our research addresses.

7 Conclusions

In this work, we analyzed the GPUs of Nvidia Jetson Nano and Nvidia Jetson Orin Nano, commonly chosen platforms for real-world neural network implementations, for resilience against side-channel attacks that aim to extract the weights of the target NN. First, we find multiple vulnerable points where the GPUs leak information about the parameters of the target DNN. Subsequently, we demonstrate the extraction of weights and biases of convolutional and dense layers. Overall, the neural network implementations of Nvidia's TensorRT framework are vulnerable to parameter extraction using EM side-channel attack despite the networks running in a highly parallel and noisy environment. Protecting their implementations in security or privacy-sensitive applications remains an open problem.

Acknowledgments

This research was supported by: an ARC Discovery Project number DP210102670; the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972; AiSecTools (VJ02010010); PROACT project of Dutch Research Agenda (NWA.1215.18.014) and Netherlands Organisation for Scientific Research (NWO); TTW PREDATOR project 19782 (NWO).

References

[1] https://www.langer-emv.de/en/product/mfa-active-1mhz-up-to-6-ghz/32/mfa-r-0-2-75-near-field-micro-probe-1-mhz-up-to-1-ghz/854. Accessed: 2022-01-25.

[8] The clock frequency is not disclosed in these attacks, but it is at most 800MHz as both attack XILINX ZYNQ chip [12].