Tendências em Hardware e Estratégias de Otimização para Deep Learning na Borda: Uma Revisão para a Indústria 5.0

Heduardo Witkoski Barcelos da Rocha[1], **Fábio Luís Livi Ramos**[2]

[1]Universidade Federal do Pampa (UNIPAMPA) – Bagé/RS – Brazil

heduardorocha.aluno@unipampa.edu.br, fabioramos@unipampa.edu.br

Resumo. *A transição para a Indústria 5.0 exige que modelos de Deep Learning (DL) operem localmente em dispositivos de borda para garantir baixa latência e privacidade. Contudo, arquiteturas tradicionais enfrentam limitações severas devido ao custo energético do tráfego de dados. Este artigo fornece uma revisão de componentes de DL, otimizações de software e tendências de hardware. A contribuição central é uma análise de estratégias para mitigar o gargalo de von Neumann em ambientes com restrição de recursos.*

1. Introdução

O avanço das Redes Neurais Profundas (DNNs) revolucionou setores como saúde e veículos autônomos [Moreira et al. 2025, Zaman et al. 2022]. Com a projeção de que o mercado de IA embarcada atinja US$ 22,4 bilhões até 2030, a computação de borda (Edge Computing) tornou-se imperativa para reduzir a latência e assegurar a soberania dos dados. Todavia, modelos modernos possuem milhões de parâmetros, excedendo a capacidade de armazenamento e processamento do hardware portátil convencional [Boumendil et al. 2024, Samanta et al. 2024].

Um dos principais obstáculos técnicos reside no ”gargalo de von Neumann”, em que a separação física entre memória e unidade de processamento gera um custo energético proibitivo durante a transferência de dados. Enquanto uma operação aritmética de 32 bits consome cerca de 0,9 pJ, um acesso à memória DRAM externa pode demandar 640 pJ, um impacto energético 700 vezes superior ao processamento em si [Zaman et al. 2022]. Este artigo analisa as tendências tecnológicas para mitigar esse custo através da utilização de técnicas de software e hardware.

2. Visão Geral de Deep Learning (DL)

As DNNs estruturam-se predominantemente sobre camadas convolucionais (CONV), responsáveis pela extração de padrões espaciais, representando frequentemente mais de 90% das operações totais da rede. Em contrapartida, as camadas Totalmente Conectadas (FC) executam a classificação final, mas demandam um volume massivo de parâmetros e largura de banda de memória [Zaman et al. 2022].

A evolução para a borda forçou a criação de arquiteturas compactas. Modelos como SqueezeNet utilizam módulos Fire para reduzir parâmetros via convoluções 1x1, enquanto a família MobileNet introduz as Depthwise Separable Convolutions. Esta técnica decompõe uma convolução padrão em duas etapas, reduzindo o custo computacional em até 9 vezes com perda mínima de acurácia, tornando-as o padrão para visão computacional em dispositivos móveis [Moreira et al. 2025, Samanta et al. 2024, Howard et al. 2017, Iandola et al. 2016].

2.1. Treinamento e Inferência na Borda

A inferência ocorre quando um modelo pré-treinado processa novos dados para gerar predições em tempo real. Já o treinamento *on-device* permite que o dispositivo aprenda e se adapte a novos contextos localmente. O desafio reside na precisão numérica: enquanto a inferência tolera representações de baixa precisão (8 bits ou menos), o treinamento exige alta fidelidade para o cálculo de gradientes via *backpropagation*, elevando drasticamente a demanda por energia e memória SRAM *on-chip* [Boumendil et al. 2024, Samanta et al. 2024].

3. Otimizações de Software e Eficiência Algorítmica

A Tabela 1 sintetiza alguns dos principais métodos para reduzir a carga computacional das DNNs antes de sua implantação no silício.

Tabela 1. Tendências de Software para DL na Borda

Método	Descrição Técnica	Vantagem	Maturidade
Quantização	Reduz a precisão numérica (ex: FP32 para binário) para minimizar o uso de memória [Zaman et al. 2022].	Modelo 32x menor	Alta
Poda (Pruning)	Elimina conexões redundantes com pesos próximos a zero, gerando redes esparsas [Boumendil et al. 2024].	3,7x menos energia	Alta
AdderNets	Substitui multiplicações por somas baseadas na norma L_1, reduzindo o gasto operacional [Samanta et al. 2024].	44,6% economia no treino	Média
NAS	Automatiza a busca por arquiteturas integrando restrições de hardware [Zaman et al. 2022].	63% economia na inferência	Média

4. Arquiteturas de Hardware e Fluxo de Dados

A eficiência de um acelerador de IA depende de como ele gerencia o reúso de dados nos Elementos de Processamento (PEs). No fluxo *Weight-Stationary*, os pesos permanecem estáticos nos PEs para minimizar acessos à memória. Já o fluxo *Row-Stationary*, implementado no processador Eyeriss, maximiza o reúso local de pesos e ativações, sendo altamente eficiente para convoluções ao reduzir o tráfego de dados com a DRAM externa [Zaman et al. 2022, Samanta et al. 2024, Chen et al. 2017].

A Tabela 2 contém inovações de hardware projetadas para superar as limitações das arquiteturas genéricas (CPUs e GPUs).

5. Discussão: Sinergia e Trade-offs no Co-design

A análise dos dados apresentados revela que a eficiência em Deep Learning na borda não é fruto de uma única técnica, mas da convergência entre otimização algorítmica e arquitetura de hardware. Atualmente, o ecossistema é liderado por plataformas versáteis como o Raspberry Pi e a linha NVIDIA Jetson [Moreira et al. 2025], mas a Indústria 5.0 pode se beneficiar da operação abaixo de 1W.

Tabela 2. Tendências de Hardware para DL na Borda

Tecnologia	Descrição Técnica	Desempenho	Maturidade
FPGA (SECDA)	Hardware reconfigurável que adapta a infraestrutura à rede [Samanta et al. 2024, Haris et al. 2023].	3,4x aceleração	Produção
SparkNet	Arquitetura leve com camadas paralelas e uso de RAM on-chip [Xia et al. 2021].	44,48 GOP/s/W	Produção
CIM (ISAAC)	Realiza cálculos dentro de arrays de memória (ReRAM) [Zaman et al. 2022].	14,8x throughput	Protótipo
Neuromórfico	Chips baseados em eventos (spikes) que emulam o cérebro [Zaman et al. 2022].	1278 GOPS/W	Pesquisa
MCU (AIfES)	Framework para treino e inferência on-device em microcontroladores [Samanta et al. 2024].	54% economia de RAM	Produção

Um ponto crítico de discussão é o trade-off entre precisão e eficiência energética. Enquanto a quantização para modelos binários (XNOR-Net) reduz drasticamente a área de silício e o consumo de energia, ela introduz uma degradação de acurácia que pode ser inaceitável em contextos de missão crítica, como o diagnóstico médico em tempo real ou o controle de robótica colaborativa (cobots). A Destilação de Conhecimento surge aqui como um paliativo promissor, permitindo que redes compactas(alunos) herdem a robustez de modelos massivos (professores).

Além disso, a transição para a Computação Neuromórfica prioriza o processamento baseado em eventos (spiking), no qual as SNNs emulam a eficiência do cérebro humano. Contudo, desafios de fabricação em tecnologias de Computação em Memória (CIM) e a imaturidade de algoritmos de treino favorecem os FPGAs como uma das soluções industriais mais viáveis no curto prazo devido à sua flexibilidade pós-fabricação.

6. Conclusão

Este artigo analisou tendências e estratégias fundamentais para viabilizar o Deep Learning em ambientes de borda, sublinhando que o gargalo de memória é um dos principais entraves para a autonomia tecnológica na Indústria 5.0. Conclui-se que a simples redução de operações aritméticas não é suficiente; é imperativo otimizar o fluxo de dados (*data-flow*) para minimizar o acesso à DRAM externa, que consome até 700 vezes mais energia que o processamento local.

As contribuições desta revisão demonstram que:

- O co-design hardware-software apresenta-se como um dos possíveis caminhos para atingir latências de milissegundos com orçamentos energéticos restritos.
- Arquiteturas como MobileNet e SqueezeNet mostraram que a eficiência estrutural pode ser tão importante quanto a força bruta computacional;
- Paradigmas emergentes como a Computação Neuromórfica e CIM prometem alinhar a IA aos princípios da Indústria 5.0, focando em sustentabilidade e sistemas autônomos de baixo consumo (milliwatts).

Como trabalhos futuros, sugere-se a investigação de protocolos de Aprendizado Federado aplicados a aceleradores neuromórficos, permitindo que dispositivos de borda

aprendam de forma colaborativa sem comprometer a privacidade dos dados. O horizonte aponta para sistemas que se moldam dinamicamente às restrições físicas do silício e às demandas mutáveis do chão de fábrica moderno.

Referências

Boumendil, A., Bechkit, W., and Benatchba, K. (2024). On-device deep learning: Survey on techniques improving energy efficiency of dnns. *IEEE Transactions on Neural Networks and Learning Systems*. Early Access.

Chen, Y.-H., Krishna, T., Emer, J. S., and Sze, V. (2017). Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE Journal of Solid-State Circuits*, 52(1):127–138.

Haris, J., Gibson, P., Cano, J., Agostini, N. B., and Kaeli, D. (2023). SECDA: Efficient hardware/software co-design of FPGA-based DNN accelerators for edge inference. *Journal of Parallel and Distributed Computing*, 173:140–151.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., and Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.

Iandola, F. N., Han, S., Moskewicz, M. W., Ashraf, K., Dally, W. J., and Keutzer, K. (2016). Squeezenet: Alexnet-level accuracy with 50x fewer parameters and ¡0.5 mb model size. *arXiv preprint arXiv:1602.07360*.

Moreira, L. F. R., Moreira, R., Travençolo, B. A. N., and Backes, A. R. (2025). Deep learning based image classification for embedded devices: A systematic review. *Neurocomputing*, 623:129402.

Samanta, A., Hatai, I., and Mal, A. K. (2024). A survey on hardware accelerator design of deep learning for edge devices. *Wireless Personal Communications*, 137:1715–1760.

Xia, M., Huang, Z., Tian, L., Wang, H., Chang, V., Zhu, Y., and Feng, S. (2021). Sparknoc: an energy-efficiency fpga-based accelerator using optimized lightweight cnn for edge computing. *Journal of Systems Architecture*, 115:101991.

Zaman, K. S., Reaz, M. B. I., Md Ali, S. H., Bakar, A. A. A., and Chowdhury, M. E. H. (2022). Custom hardware architectures for deep learning on portable devices: A review. *IEEE Transactions on Neural Networks and Learning Systems*, 33(11):6068–6088.