Enabling Embedded Vision Neural Network DSPs

Pulin Desai, Director, Product Marketing, Tensilica Imaging and Vision Group, Cadence

Neural networks are now being developed in a variety of technology segments in the embedded market, from mobile to surveillance to the automotive segment. The computational and power requirements to process this data is increasing, with new methods to approach deep learning challenges emerging every day. The Cadence® Tensilica® Vision C5 DSP, the industry's first DSP dedicated to neural network processing and architected specifically for multi-processors, offers fast and vast computational capacity, future-proofing at its best, and can be used in all applications that require neural network processing.

Contents

Introduction

Vision processing systems must be designed holistically, for all platforms, with hardware and software developed in tandem. To develop this technology, designers must use tools and IP that enable:

- Efficient algorithms, to speed training and minimize computation
- Hardware platforms that meet the target cost and power usage for each application

By designing with the entire system in mind, designers can create transformative vision-enabled products as quickly and efficiently as possible. Taking a systems-level perspective pays off with in faster and better design, shorter verification cycles, software that works with the hardware, and new product leadership.

Background

The expression "deep learning" in the context of neural networks was introduced in 2000 (Aizenberg, Aizenberg and Vandewalle 2000). By then, convolutional neural networks (CNNs) already processed an estimated 10% to 20% of all the checks written in the US (Lecun 24 March 2016). By 2008, deep learning and machine learning were well under way (Figure 1).

As neural networks evolve, the need for embedding the processors in devices—rather than using CPUs and GPUs—has grown. The processing power and speed required for these networks have not kept up with the requirements of neural network applications, particularly in the field of vision applications.

Until now, the requirements of these networks can only be done using the resources of a datacenter. The sheer volume of computational requirements can achieve multiple TMACs per second (TMAC/s), which currently doesn't exist on the small scale of embedded devices.

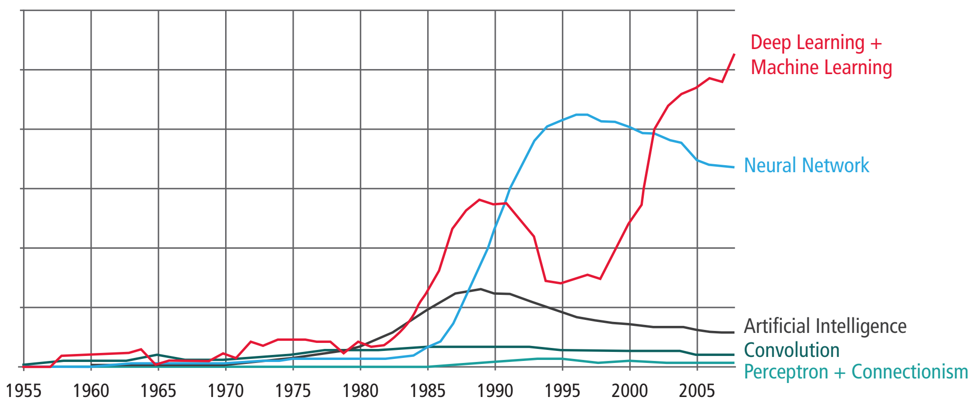

Figure 1: Google's Ngram Viewer (Jean-Baptiste Michel 2010) shows a graph of how frequently an inputted term appears in its corpus of books. While not a precise method to determine specifics, this tool shows the arc of general interest in a topic*

Current Neural Network Trends

The current neural network trends have introduced three significant challenges:

- Increased computational needs
- Fast-evolving networks
- Increased use cases for neural networks

Increased Computational Needs

The deeper the network, the shorter the training and inference, providing a higher-accuracy output (Figure 2), but at the cost of using a great deal of computational resources and power.

We can perform image recognition tasks and training in the cloud. Computational value is theoretically infinite. But that value is greatly diminished if you must go back to the cloud for each recognition task. Using a deterministic model, after training is complete and downloaded, image processing should be able to function independently, and new data to enhance the training dataset should be uploadable, as well. Using a stochastic model, the training stage and the inference stage can be done in parallel.

The real question becomes: How do we embed the task of recognition into the application?

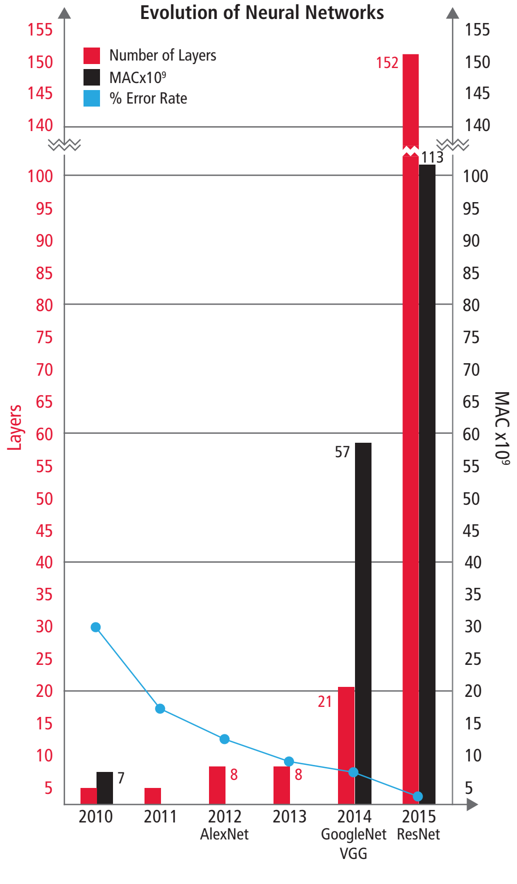

Figure 2: Layers, computation, and error rate (He 19 June 2016); the number of layers required to increase accuracy is more than inversely proportional

Fast-Evolving Networks

Considering the speed at which neural networks are changing and developing (Figure 2, showing the evolution of MAC requirements for the last five years), how do we even pick a platform today for a product that may ship in two years from now? Five years from now? With the development of new neural networks with ever-changing architectures, there is no guarantee that what works now will work in a system in the future.

As Steve Roddy from Cadence has said of the automotive market: "Manufacturers' biggest fear is that by the time their product comes to market, the platform they picked will be as useless as a hand crank starter."

Use Cases

When neural networks were first introduced, they were first used as a subject of mere interest, but were limited by the computational requirements to make them practical (Miikkulainen, et al. 2017). They focused on simple pattern recognition, and were used in a limited scale, including:

- Written character detection (for example, handwriting detection to deposit checks at the bank)
- Image compression
- Some industrial applications (fault detection, mining, etc.)

Only after large amounts of computing power have become readily available and machine learning systems have scaled up to handle the computer requirements necessary, have neural networking systems become applicable to all industries. These applications not only recognize patterns, but can also make actionable decisions based on the training they have received. These networks can be applied to:

- Aerospace and military applications
- The arts, including visual, music, other performance art
- Augmented/virtual reality and gaming
- Automotive applications, including ADAS, infotainment, GPS, autonomous driving
- Cryptography
- Tracking economic trends and making advanced market predictions
- Infrastructure management, including the power grid, water resources, waste management, roadways, emergency services
- Language applications, including real-time translation, lip reading, voice-to-text translation, other human/computer interactions
- Medical applications, including diagnostic tools, genetic predictions, drug development, and tracking treatment outcomes
- Mobile applications
- Process engineering
- Robotics
- Scientific research
- Security, surveillance, and policing, including camera monitoring, facial recognition, and fraud detection
- Telecommunication systems and infrastructure
- Transportation and fleet management
- Wearable and implanted devices

All of these applications must not only exist in the cloud; they must also be embedded into the end application. And all are power hungry.

The Question

In his *Notes from the Neural Edge*, Bernard Murphy asks, "What is the best platform for neural nets as measured by performance and power efficiency? It's generally agreed that CPUs aren't in the running, GPUs and FPGAs do better but are not as effective as DSPs designed for vision applications, [and] DSPs tuned to vision and neural net applications do better still. And as always, engines custom-designed for [neural network] applications outperform everything else … designed around minimizing power per MAC, minimizing data movement and optimizing MACs per second." (Murphy 2017)

Current Solutions to Implement Neural Networks

Currently, there are two main choices for implementing neural networks: CPUs/GPUs or using hardware accelerators with a CPU/GPU or an imaging DSP. Each of these options solves some of the challenges facing designers, but none meet all the requirements to meet the trends listed above.

CPUs and GPUs

Using software on CPUs and GPUs are possibilities. They are easy to use, and there are many tried-and- true tools to choose from with this option. They are always re-programmable, so they can always be changed to fit changing environments. They can perform all the computations required.

The downside is that because they are designed as general-purpose processors, they are not power efficient. GPUs are slightly more efficient than CPUs, but they are still too power hungry to be practical. Simply stated: you can't carry a datacenter in the trunk of your car, much less in your mobile phone.

Hardware Accelerators

This option offloads the convolutional layers to hardware, separate from the processor or DSP. While this may speed up the processing time of the entire system to up to 1TMAC/frame, some major drawbacks must be considered before employing this option.

Development Time

Taking this option requires that it be worked into the development schedule and be completed by tapeout. Also, employing this option requires that the software be partitioned between the programmable core (the CPU, GPU, or DSP) and the accelerator. The time required to develop this solution may not be feasible.

Data Transfer Overhead

Accelerators are designed to offload (and therefore accelerate) only convolution layers. All the other layers—pooling layers, fully connected layers, vectorized layers, output layers—are not touched by the accelerator. The non-convolutional layers must be run on an imaging DSP, control CPU, or GPU—with all their inherent problems listed above.

Transferring data between the hardware accelerator and the other processors (see Figure 3) simply takes time—slowing down performance— and power. In addition, while running the neural network, the hardware accelerator may prevent imaging DSPs from running on other applications. This limits its versatility.

Hardware accelerator solutions may be able to provide the necessary speed, but at the cost of power usage and increased ramp-up time.

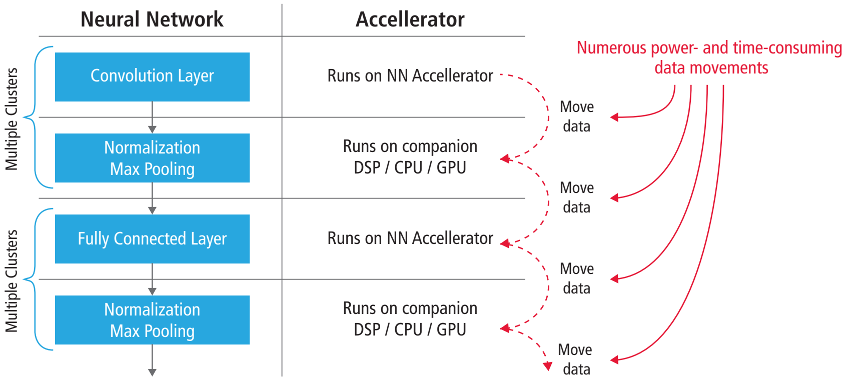

Figure 3: The problem with accelerators

“Future Proofing”

The hardware accelerator option doesn’t allow for any scalability or adjustment after the development is complete. The hardware is fixed at tapeout; it cannot be changed to address any future neural network developments. “Future proofing” is impossible using this option.

Until now, we have not been able to solve the problems of implementing neural networks in embedded systems using the architectures we have today.

The Cadence Solution: The Tensilica Vision C5 DSP

Optimized for vision, radar/lidar, and fused-sensor applications, the Cadence Tensilica Vision C5 DSP is the industry’s first DSP dedicated to neural network processing and architected from the ground up specifically for multi-processors. Achieving unprecedented speeds and low power usage, the Vision C5 DSP set meets all the requirements of advanced neural network technology.

Built on almost twenty years of Tensilica Xtensa® multi-processor experience, this solution features a shared memory architecture and allows for interrupts, queues for synchronization, and synchronous multi-processor debugging. The Vision C5 DSP accelerates all layers, not just convolutional functions, allowing the DSP free to run other applications.

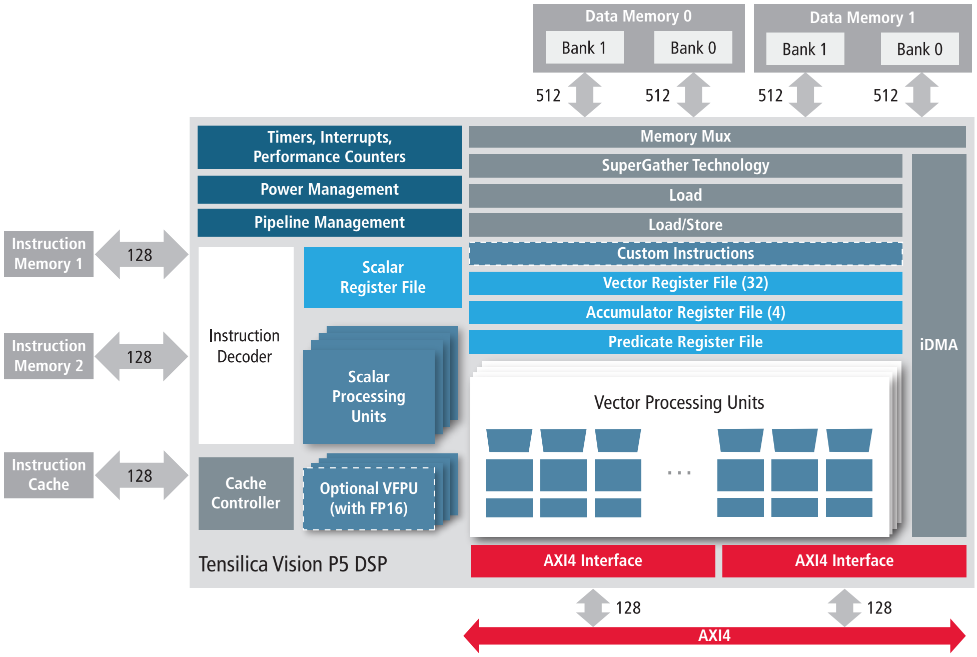

Figure 4: Vision C5 DSP architecture

Speed and Computational Capacity

The Vision C5 DSP directly addresses the issue of increased computational needs, offering 1TMAC/s/core, and is architected for multi-core design allowing a multi-TMAC solution to be possible. The Vision C5 DSP is immediately useful for applications that run constantly, such as in automotive applications.

This solution has the highest computational capacity in the industry, featuring:

- 1024MAC (8-bit)
- 128-way SIMD VLIW processor
- 1024-bit memory interface with dual load and store
- Integrated DMA

Scalability: Future Proofing at its Best

Eliminating the need for expensive data transfers to and from a hardware accelerator, the Vision C5 DSP consolidates the processing of multiple layers simultaneously, resulting in fast speeds and low power.

This is a flexible and future-proof solution, supporting:

- Variable kernel sizes, depths, and input dimensions
- Different compression/decompression techniques
- On-the-fly decompression
- Flexible instruction set for quantization
- Normalization and pooling layers

Ease of Development

This solution fits in with other software tools offered by the Cadence Tensilica suite, so developers don't need to learn a new platform, thus allowing for aggressive time to market. The solution fits within various neural network frameworks, combining the users' code and existing tools and libraries, streamlining the development process.

Use Cases

Even though the word "vision" is in its name, the Vision C5 DSP is designed for any kind of neural network processing. This DSP is architected for all multi-processor clusters. Whether your neural network is for the mobile industry, surveillance, automotive, or anywhere in between, this solution is flexible enough for all applications, from the minute scale to the grand.

Conclusion

To summarize, a successful neural network DSP solution must be:

- **Embedded:** True for all markets, from mobile to automotive, a vast amount of data must be processed on the fly. While the training of a neural network may take place mostly offline, the applications that use them must be embedded within their own system, whether in the mobile, surveillance, or automotive markets.
- **Able to process a staggering amount of data:** No matter the application, the amount of data to be processed must happen as instantaneously as a car accident. Current solutions can, at best, reach up to 1TMAC/frame, but this is not enough. The Vision C5 DSP is architected for multi-core design, allowing a multi-TMAC solution, with the ability to run multiple neural networks constantly.
- **Use power efficiently:** Just as we don't carry datacenters around with us in our car or on our device, we also can't carry around power sources with us wherever we go. The Vision C5 DSP is optimized for neural networks, without wasting time and power.
- **Future proofed:** As the development of neural network processing grows, the products using neural networks in development now may need reprogramming by the time they are shipped. The platform must be able to grow with the industry implementing them. Simply put, the platform must be future-proofed.

The Cadence Tensilica Vision C5 DSP meets all of these requirements. It is the industry's first standalone, fully dedicated neural network DSP, architected for multi-processor clusters. Faster and using lower power than previous solutions, the Vision C5 DSP can do anything that a hardwired accelerator can do, only with less power and more flexibility, with support for new layers as the design changes.

Ultimately by designing with the entire system in mind, you can create transformative vision-enabled products as quickly and efficiently as possible. The concept of System Design Enablement guides everything we do at Cadence, and taking a systems-level perspective pays off for designers in elegant design, shorter verification cycles, software that works with the hardware, and new product leadership.

Further Information

To learn more about the Tensilica Vision C5 DSP, visit https://ip.cadence.com/vision.

References

1. Aizenberg, Igor, Naum N. Aizenberg, and Joos P.L. Vandewalle. 2000. *Multi-Valued and Universal Binary Neurons: Theory, Learning and Applications*. 10.1007/978-1-4757-3115-6. doi:10.1007/978-1- 4757-3115-6.
2. Canziani, A., Paszke, A., & Culurciello, E. 2016. "An Analysis of Deep Neural Network Models for Practical Applications, Abs/1605.07678." *Computing Research Repository (CORR)*.
3. He, Kaiming. 19 June 2016. "Deep Residual Networks: Deep Learning Gets Way Deeper." *International Conference on Machine Learning*. New York: ICML.
4. Jean-Baptiste Michel*, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K. Gray, William Brockman, The Google Books Team, Joseph P. Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, Steven Pinker, Martin A. Nowak, and Erez Lieberman Ai. 2010. "Quantitative Analysis of Culture Using Millions of Digitized Books." *Science*. https://books.google.com/ngrams.
5. Krizhevsky, A, I Sutskever, and G. Hinton. 2012. "ImageNet Classification with Deep Convolutional Networks." *Advances in Neural Information Processing Systems* (NIPS).
6. Lecun, Yann. 24 March 2016. "Deep Learning and the Future of AI." *CERN Colloquium*. Meyrin, Switzerland: CERN.
7. Miikkulainen, Risto, Jason Liang, Elliot Meyerson, Aditya Rawal, Dan Fink, Olivier Francon, Bala Raju, et al. 2017. "Evolving Deep Neural Networks." *eprint arXiv:1703.00548* (Cornell University Library). https://arxiv.org/abs/1703.00548v2.
8. Murphy, Bernard. 2017. *Notes from the Neural Edge*. February 09. https://www.semiwiki.com/forum/content/6589-notes-neural-edge.html.

cādence®

Cadence Design Systems enables global electronic design innovation and plays an essential role in the creation of today's electronics. Customers use Cadence software, hardware, IP, and expertise to design and verify today's mobile, cloud and connectivity applications. www.cadence.com