CASE STUDY

Internet of Things
Computer Vision

Bringing AI Inference to the Edge

Computer vision applications generate too much data to send to the cloud for machine learning inference. That's why iAbra created a framework for machine learning that enables embedded intelligence on the camera

At a Glance:

- To avoid overloading networks with computer vision data, iAbra are bringing machine learning to the edge.
- I-Abra's solution mimics the human brain using FPGAs, and fits in devices that have small space, power, and weight profiles.
- Rather than mapping problems to generic neural networks, which are too large to fit on FPGAs, they build an optimized neural network for each problem.

Most neural networks today run in a data center, but it's not always easy to get data to the cloud. Computer vision applications, for example, are extremely data intensive and would overwhelm even a 5G network. iAbra's tools create neural networks that are small enough to run on a field programmable gate array (FPGA), so that inference can be carried out at the edge on a low power, small and light device.

Challenge

- Enable machine learning inference of high definition (HD) video streams, and other machine learning applications that require huge amounts of data.
- Overcome network limitations by enabling inference to take place on embedded devices, rather than in the cloud.
- Create an embedded solution that can withstand the harsh environment of the edge, and work within strict power, space and weight constraints.

Solution

- iAbra's tools create neural networks that run on Intel® Arria® 10 FPGAs, providing the performance required in a low power, space and weight device.
- The tools help democratize the creation of neural networks by using what-you-see-is-what-you-get (WYSIWYG) tools to create the neural network configuration.
- The Intel® Xeon® Gold 6148 processor is used by iAbra Neural PathWorks to create the neural networks.
- The Intel Atom® x7-E3950 processor manages the workflow on the embedded platform.

Results

- The iAbra Neural Synapse framework can be used for machine learning inference at the edge in smart cities, healthcare and other data-rich applications.

Bringing Machine Learning to the Edge

Making the best use of resources across a city is a huge challenge. That's why there's been growing interest in recent years in the idea of the smart city. Using sensors and video cameras, the smart city is able to collect data about what's going on in urban areas. Artificial intelligence (AI) could detect, for example, how many

people are standing in a bus queue, which modes of transport people are using, and where there is flooding or road damage. In exceptional circumstances, AI could alert emergency services to investigate if vehicles are driving against the traffic flow, or people have suddenly started running. By collecting reliable, real-time data from the streets, it becomes easier to manage resources to improve quality of life, productivity and emergency response.

There is a challenge, though. To be effective, computer vision applications require a huge amount of data. A single HD stream generates in the region of 800 to 900 megabits per second. 5G will bring higher broadband and the ability to increase the density of devices within a geographic area, but computer vision solutions still risk overwhelming the network. Operators will want to protect quality of experience for consumer applications, and may choose to charge smart city applications heavily for the use of their finite bandwidth. Even if bandwidth is available, it often requires an impractically huge amount of energy to transmit HD streaming video over radio frequencies, given that the cameras are working within a limited power budget. It's not just about video: high resolution sensor data can be used to detect pollution or gather granular data on how the community is using resources.

A single HD stream generates in the region of 800 to 900 megabits per second.

The answer, then, is to put the machine learning inference at the edge of the network so that only essential information is sent to the cloud. That processing requires a new kind of AI device, better able to withstand the harsh environment and resource constraints of the edge than a data center server would be. Power at the edge may be battery-based, with solar and wind recharging the battery - such sustainable implementations are sometimes referred to as the "green edge". Space may be restricted, and low weight may be a requirement where compute devices need to be mounted inside cameras.

For these reasons, smart cameras are not yet widely deployed: putting a server at the edge of the network that can withstand a wide range of heat tolerances is expensive, especially considering the cost of cabling.

Solution Details

iAbra has found the answer by mimicking the human brain using FPGAs, which are compute devices where the circuitry can be reconfigured to carry out the algorithm in hardware.

A CPU has a fixed instruction set, and creating an application requires working within that instruction set, repeating instructions as necessary to get your desired output. "Using an FPGA, we create a custom instruction set for each task, with every clock cycle becoming a complete application cycle for the work," said Compton. "The application becomes the instruction set, which is only possible on FPGAs. Because you can create a custom instruction set based on your application, you can greatly optimize the resources you use, and the number of clock cycles required to execute your application. Other neural network processors use high speed memory, which requires high bandwidth and is power intensive. We are able to do everything on the FPGA itself, with no back and forth to memory. That eliminates the power required to drive memory and reduces the number of wasted cycles waiting for memory to respond to requests."

"Our solution lays out the chip, so it vaguely mimics the human brain."

—Greg Compton
CTO, iAbra

The result is that it's possible to process a neural network at the edge of the communications network using a device with a small space, power and weight profile. Neural networks are in themselves not new: they're widely used in AI, but typically within the data center where there are significant compute resources available.

iAbra's innovation has been to flip conventional thinking on how neural networks are created. Typically, problems are mapped to generic neural networks, such as ResNet, which is used for image recognition. Such networks are too big to fit into an FPGA. iAbra instead builds a new neural network for each problem, that is uniquely tailored and is highly optimized for the FPGA architecture where it will run.

The iAbra solution comprises two parts:

- **iAbra Neural PathWorks:** This is a tool for creating neural networks for embedded AI inference on the FPGA, including the intellectual property (IP, or configuration) required for the FPGA. Neural PathWorks runs on a server based on the Intel Xeon Gold 6148 processor with an attached Intel Arria 10 FPGA to ensure the neural network runs consistently in deployment. The tool enables users to provide an annotated data set, or to add annotations to a data set, and will then automatically search the design space for the optimum neural network. This software enables data citizens to create FPGA neural networks, by abstracting away the complexity of programming for FPGAs, and the complexity of building and hand-tuning traditional neural networks.
- **iAbra Neural Synapse:** This is the framework that neural network designs from PathWorks run on. It includes software for the Intel Atom x7-E3950 processor on the embedded platform and a framework to host the IP on the FPGA. The framework running on the Intel Atom processor hosts communications, receives data and sends it to any intermediate processing steps, as well as moving it in and out of the FPGA. The workflow engine in Neural PathWorks enables users to design their own process flows.

iAbra also provides a hardware development kit to help with prototyping. It is a single board computer that includes an Intel Arria 10 board attached to an Intel Atom processor over PCIe. It enables a keyboard, mouse and Ethernet network cable to be plugged in for easier testing than in the smaller production kit.

"We believe the Intel Arria 10 FPGA is the most efficient part for this application today, based on our assessment of the performance per watt," said Compton. "The embedded platform also incorporates the latest generation Intel Atom

processor, which provides a number of additional instructions for matrix processing over the previous generation. That makes it easier to do vector processing tasks. When we need to process the output from the neural network, we can do it faster with instructions that are better attuned to the application."

He adds: "A lot of our customers are not from the embedded world. By using Intel Atom processors, we enable them to work within the tried and tested Intel® architecture stack they know."

"We chose the Intel Xeon Gold 6148 processor for the network creation step as much for economics as performance," said Compton. "We optimize for matrix operations per dollar, and there's a balancing act with the FPGA. There's no point having a CPU that is faster than the amount of inference the FPGA can do, and vice versa. It is important to find the right combination."

The solution has been developed using OpenCL, a programming framework that makes FPGA programming more accessible by using a language similar to C, and enabling code portability across different types of processing devices. iAbra has used Intel® Quartus® Prime Software and Intel® C++ Compiler to develop its software, and has incorporated Intel® Math Kernel Library (Intel® MKL) which provides optimized code for mathematical operations across a range of processing platforms.

"With Intel MKL, Intel provides highly optimized shortcuts to a lot of low-level optimizations that really help our programmer productivity," said Compton. "OpenCL is an intermediate language that enables us to go from the high level WYSIWYG world to the low-level transistor bitmap world of FPGAs. We need shortcuts like these to reduce the problem domains, otherwise developing software like ours would be too big a problem for any one organization to tackle."

Technical Components of Solution

- **Intel® Xeon® Gold 6148 processor.** This processor, from the previous generation Intel® Xeon® Scalable processor family, has 20 cores supporting 40 threads and a maximum turbo frequency of 3.70GHz. It includes Intel® Advanced Vector Extensions 512 (Intel® AVX-512) which provides vectors for parallel processing that are twice the width of the previous generation processor's.
- **Intel® Arria® 10 FPGA.** These FPGAs are used in a wide range of applications including communications, data center, military, broadcast, and automotive. Integrated in iAbra's solution stack, Intel Arria 10 FPGAs enable neural networks to be processed in an embedded form factor.
- **Intel Atom® x7-E3950 processor.** Based on 14nm semiconductor technology, this processor has four cores and supports four threads, with a base frequency of 1.60GHz and a burst frequency of 2.00GHz. In iAbra's solution, the Intel Atom processor is used to manage the workflow on the embedded platform.
- **Intel® Math Kernel Library (Intel® MKL).** Intel MKL accelerates math processing routines by providing a library of highly optimized functions that take advantage of microprocessor features in Intel® architecture.

Intel Works Closely with iAbra

iAbra has a close relationship with Intel, whose engineers have helped with optimizing performance on the FPGA.

"It's been a huge help to have Intel's support as we refine our solution, and develop our code using Intel's frameworks and libraries. We've worked closely with the Intel engineers, including helping them to improve the OpenCL compiler by providing feedback as one of its advanced users."

> "The Intel FPGA Partner Program is a great way to understand the technology and the rest of the community."

iAbra participates in the Intel FPGA Partner Program and Intel® AI Builders Program. "The Intel FPGA Partner Program is a great way to understand the technology and the rest of the community," said Compton. "It's a good way to meet board vendors and build business relationships. We've been using a lot of in-depth technical information available there too. Now we are looking at moving our designs to use the new Intel® HyperflexTM architecture, we need to understand low level details on how that works. The Intel AI Builders Program helps us to present our solution to end customers."

"The Intel® AI DevCloud enables us to get cloud access to the very latest hardware, which may be difficult to get hold of, such as some highly specialized Intel® Stratix® 10 FPGA boards," said Compton.

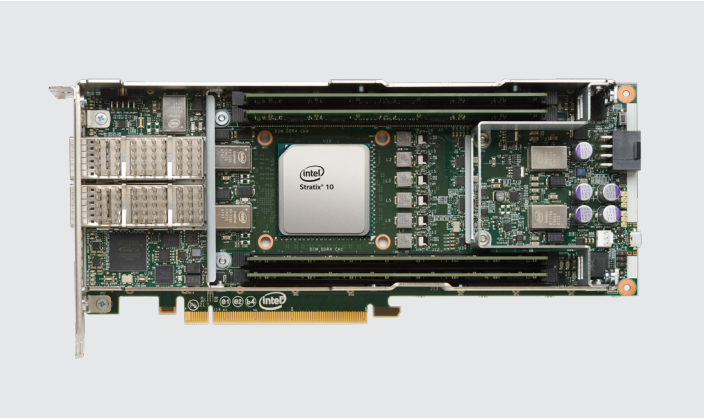

Figure 1: Intel® Stratix® 10 FPGA Programmable Acceleration Card

"It gives us a place where Intel customers can come and see our framework in a controlled environment, enabling them to try before they buy. It helped us with our outreach for a smart cities project recently."

Conclusion

iAbra has developed a portfolio of products that help to democratize the creation of neural networks on FPGAs, and solve the problems of bringing AI inference to the edge. As well as being suitable for smart cities, the platform can be used in other applications that require machine learning inference at the edge of the network.

> "The best thing Intel has done is given us a voice in the marketplace, as a small organization, it's been a massive help for us to have access to Intel's sales and partner network, as well as their technologies that enable us to bring machine learning to embedded devices."

One example is in public health where AI can be incorporated in microscopes for sample scanning. This not only helps to increase accuracy by eliminating the problem of human operators losing concentration, but also makes it possible to process high volumes of samples without a communications network, in locations such as field hospitals if there is a disease outbreak.

Another example is in industry, where iAbra helped a company to extract QR codes from shipping packets. Using AI to extract the QR codes meant that packets could be scanned at oblique angles, so fewer cameras could be installed.

Spotlight on iAbra

iAbra was founded in 2010 to create actionable information from the growing volumes of sensor data in the world. Use cases include smart cities, automotive, defense, manufacturing, healthcare and agriculture. iAbra's tools ensure that efficient neural networks are created and optimized for embedded FPGA silicon, delivering low power AI inference for a variety of applications.

Learn More

You may also find the following resources useful:

- Intel® Xeon® Gold 6148 processor
- Intel® Arria® 10 FPGA
- Intel Atom® x7-E3950 processor
- Intel® Math Kernel Library (Intel® MKL)
- Intel® Quartus® Prime Software
- Intel® C++ Compiler
- Intel® AI Builders
- Intel® FPGA Partner Program

0420/CB/CAT/PDF

342867-001EN