Linley Fall Processor Conference 2021

Premier Sponsors

Platinum Sponsor

Gold Sponsors

Silver Sponsors

Industry Sponsors

Media Sponsor

A Flexible Yet Powerful Approach to Processing Evolving Edge AI Workloads

Linley Fall Processor Conference, October 20, 2021

Cheng C. Wang, Co-Founder and Senior VP

flex-logix.com

Flex Your Computing

Company Overview

Co-Founders: **Serial Business Builder + Proven Technologist** Who Deliver

Geoff Tate, CEO

- Experienced executive taking company public
- Rambus: 4 people to IPO to $2B

Cheng Wang, Co-founder

- Industry expert with track record in tech innovation
- Winner: ISSCC Outstanding Paper Award, the premier chip design award. (Recent winners include IBM, Toshiba, Nvidia and Sandisk)

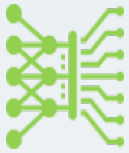
Flex Logix:

- Founded in 2014
- Profitable eFPGA Business in 2020
- Backed by top technology and innovation investors

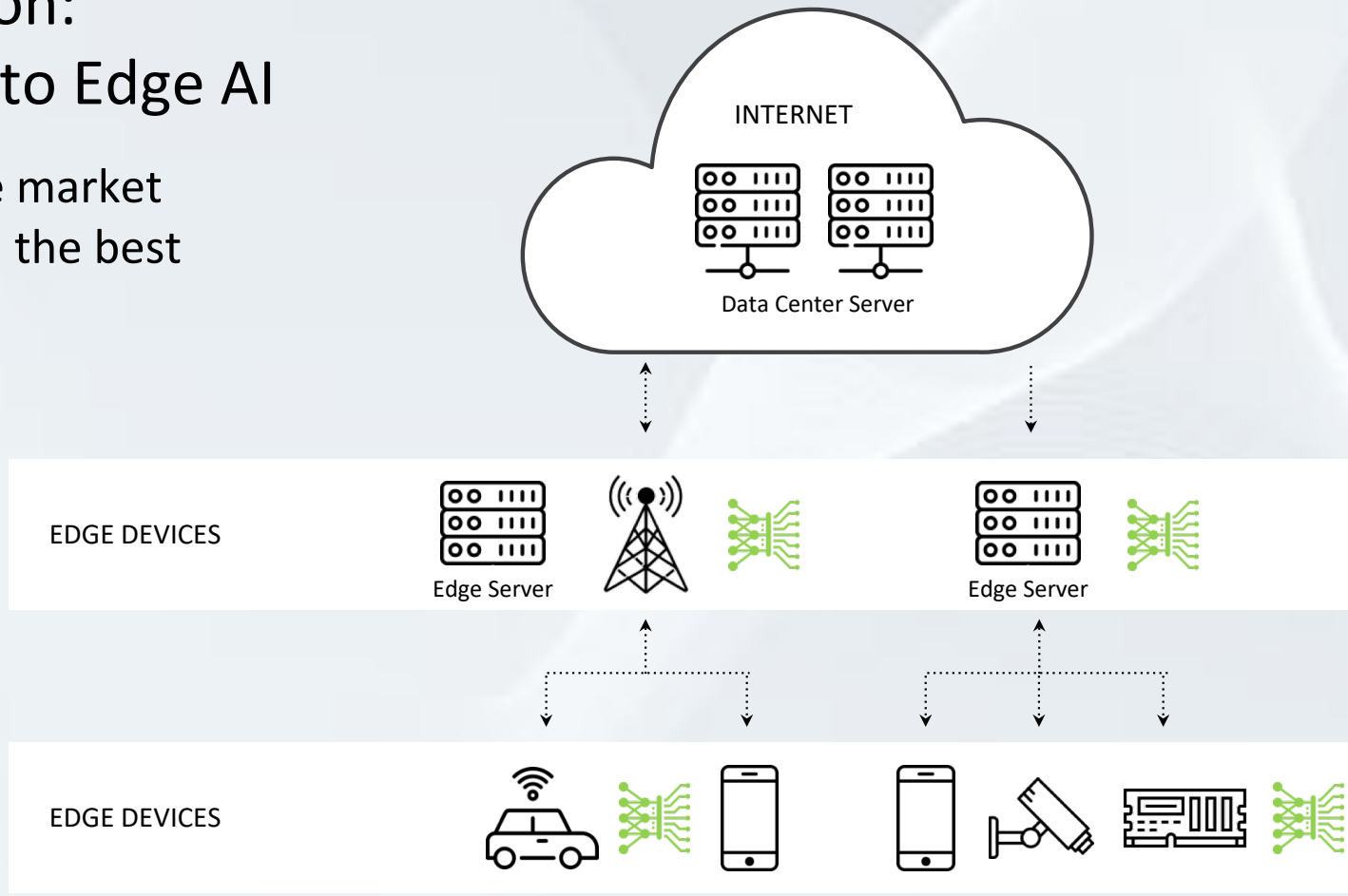
- 600 years combined experience in software, systems and semiconductors
- 25 issued US patents; 1 issued European and 1 issued Chinese patent; dozens more in application in USA and major countries

Company Evolution: Embedded FPGA to Edge AI

We identified what the market needed and developed the best solution

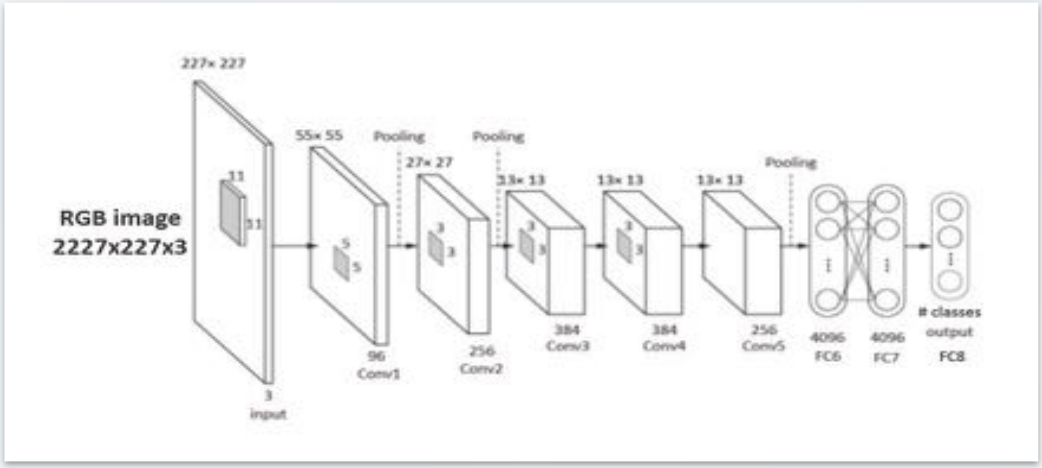


= Deep Learning

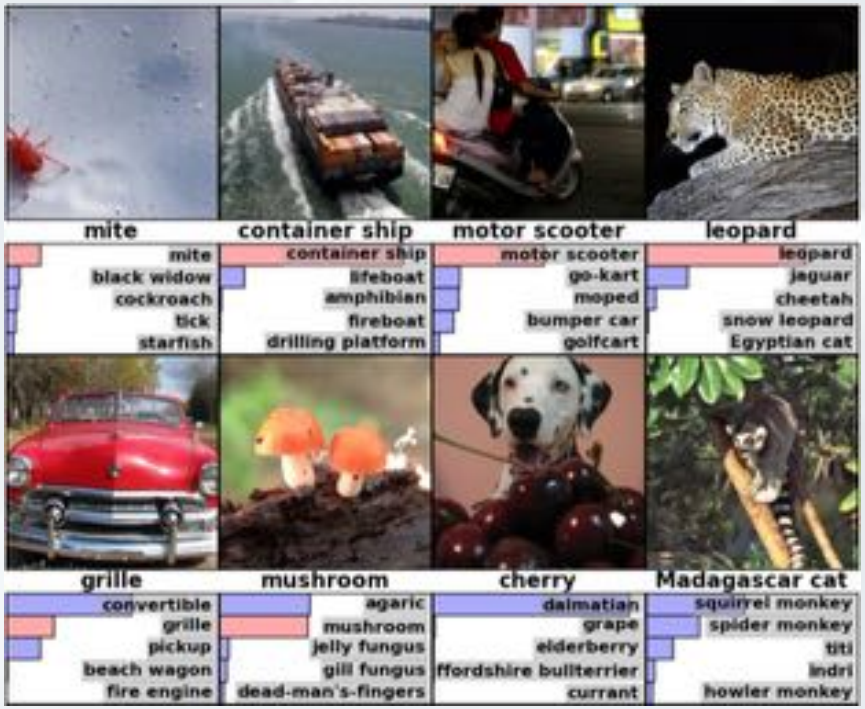


AlexNet 2012

ImageNet Competition Winner

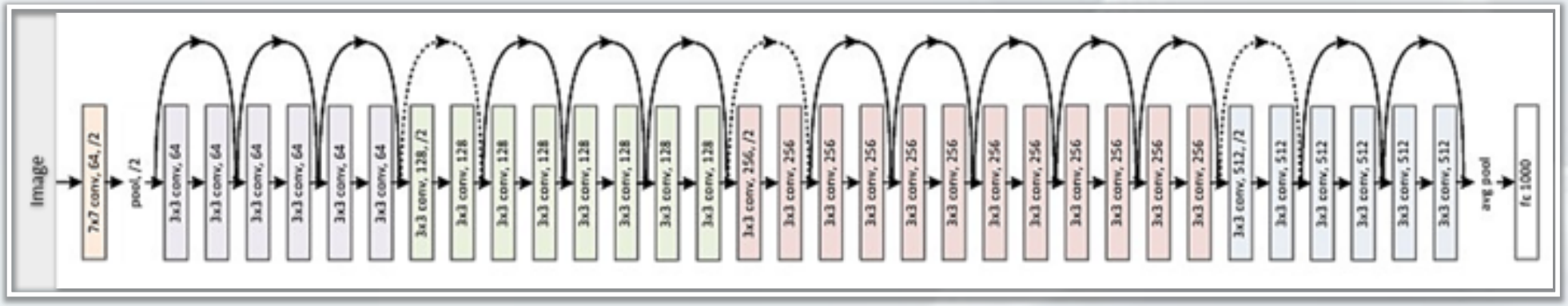


- Operator Types: 11x11, 5x5, 3x3n, MaxPool 3x3s2, FC
- Total Layers: 8
- Output is classification to 1000 classes
- Operations per Inference: 724 Million



ResNet 2015

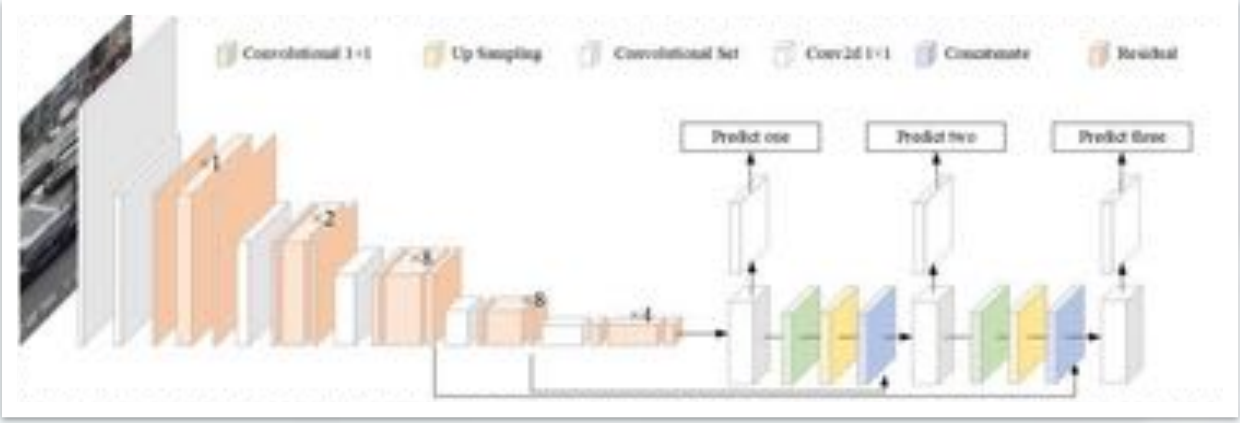
Solves Vanishing Gradient Problem by using skip (residual) connections

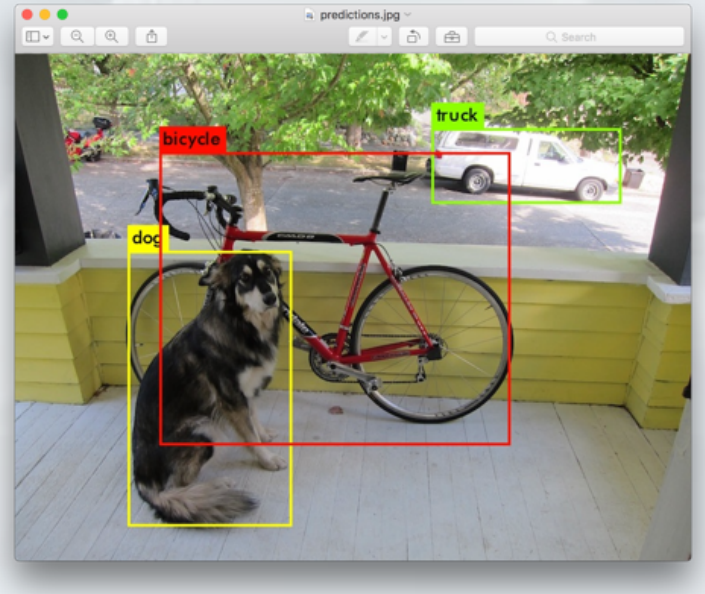


- Operator Types: 7x7s2, 3x3, 1x1, Max Pooling and Average Pooling 3x3s2, Fully Connected
- Total Layers: 18,34, 50, 102, 158 (varies by accuracy/computation tradeoff)
- Output is classification to 1000 classes
- Operations per Inference: 1.8B to 11.3B (Depending on network depth)

Yolov3 2018

Integrated Detector and Backbone

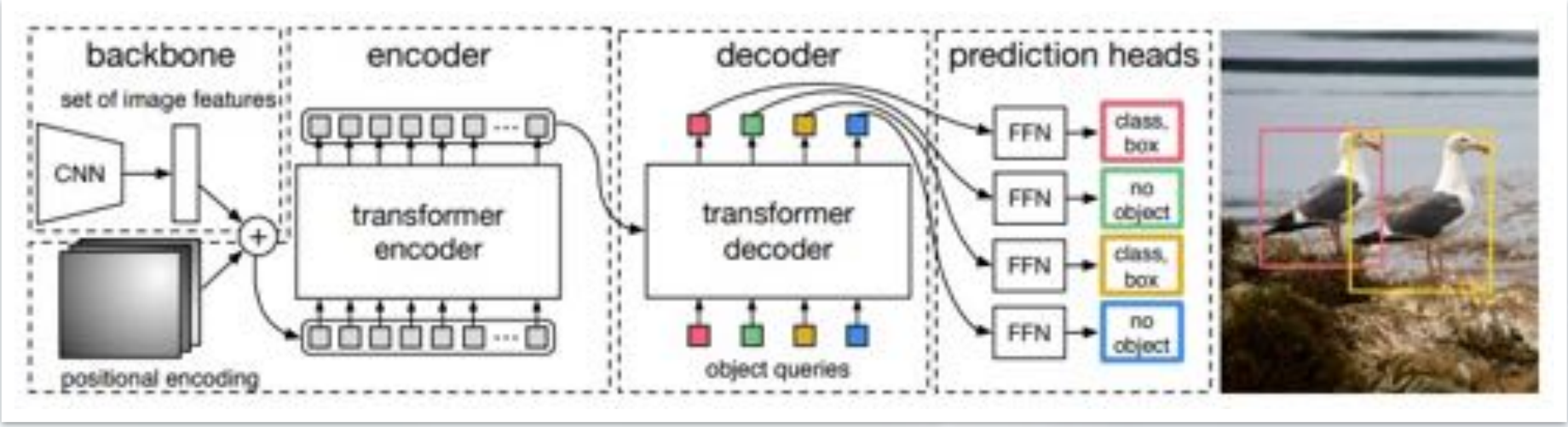




- Operator Types: 3x3, 1x1, FC
- Total Layers: 75
- Output is object detection and location for trained categories
- Operations per Inference: 178B at 608x608

DETR 2020

Combines CNN Backbone with Transformer-based Detector



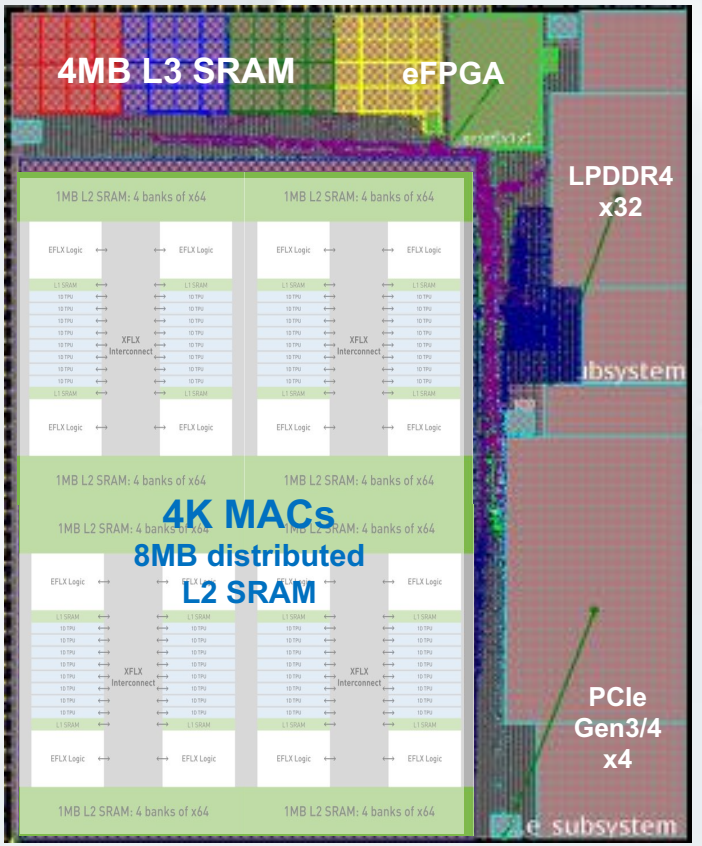
- Uses ResNet or similar CNN backbone for feature extraction
- Followed by Transformer-based Detector
- Output is object detection and location for trained categories
- Does not add large # of OP/inference (15B in transformer vs 178B in YOLOv3 backbone)
  BUT the computation is very different from CNNs

Flex Logix X1 Introduction

Dynamic TPU Array

- ASIC performance but dynamic to new models
- Accelerator/Co-processor for host processor
- Low power/High performance
- Designed for edge (B=1) applications

GPIO

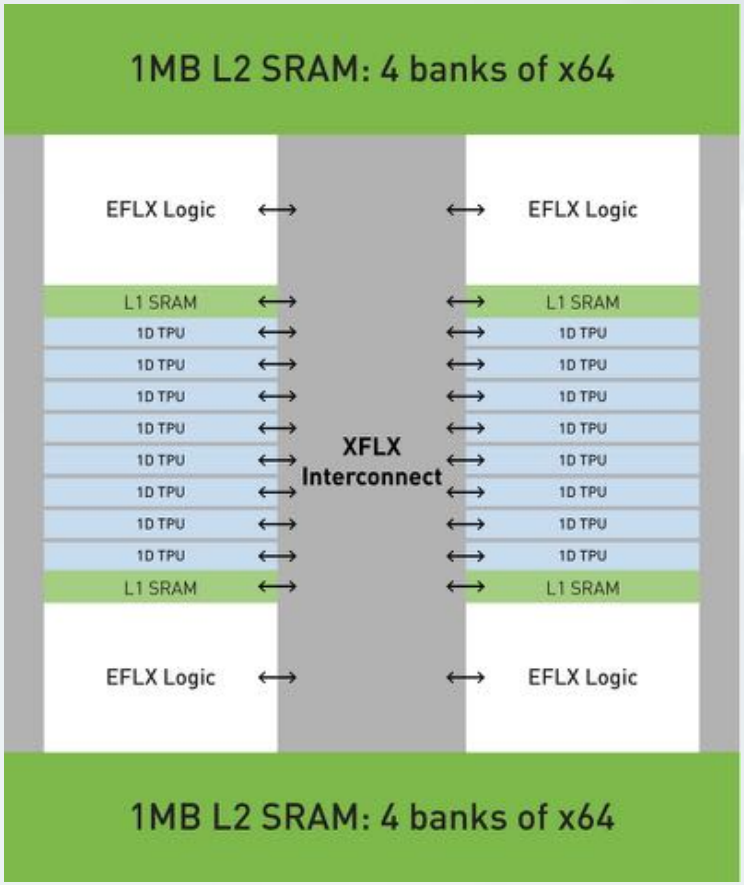


Dynamic TPU Memory Utilization

L2 SRAM
holds activations

Each **1D TPU** contains
16KB **L0 SRAM**
for active weights

L1 SRAM
holds weights
for the next layer

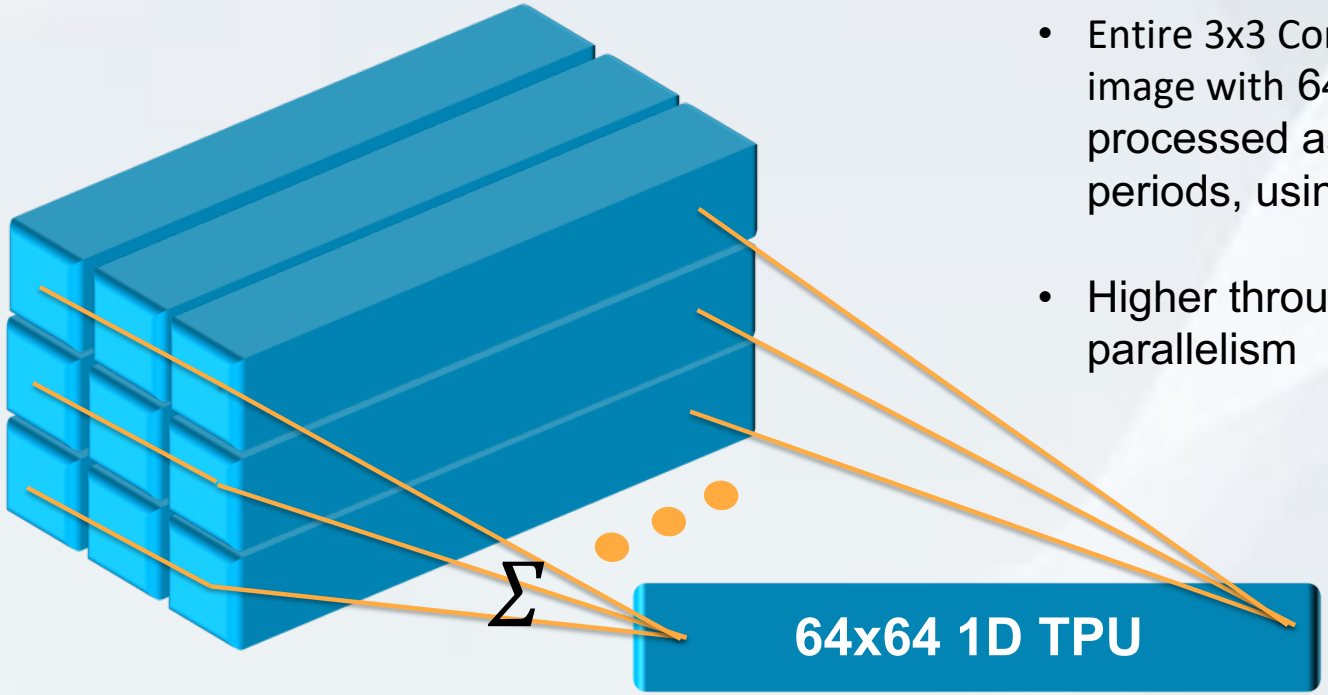


- Each tile offers 1024 MAC operations per cycle
- Each clock 64B of activations loaded from L2 SRAM and 64B of results transferred to L2 SRAM
- Weights are held in L0 SRAM in the TPU, with next layer weights pre-loaded in L1 SRAM

Dynamic TPU Array Approach

64 Int8 MACs per TPU, 4 Tiles in X1

Dynamic TPU Array Approach



- Total of 9x64 = 576 MAC operations per clock
- Entire 3x3 Convolutional layer for 608x608x64 image with 64 output filters could be processed as 608x608 windows of 64 cycle periods, using 9 1D-TPUs
- Higher throughput achievable through high parallelism

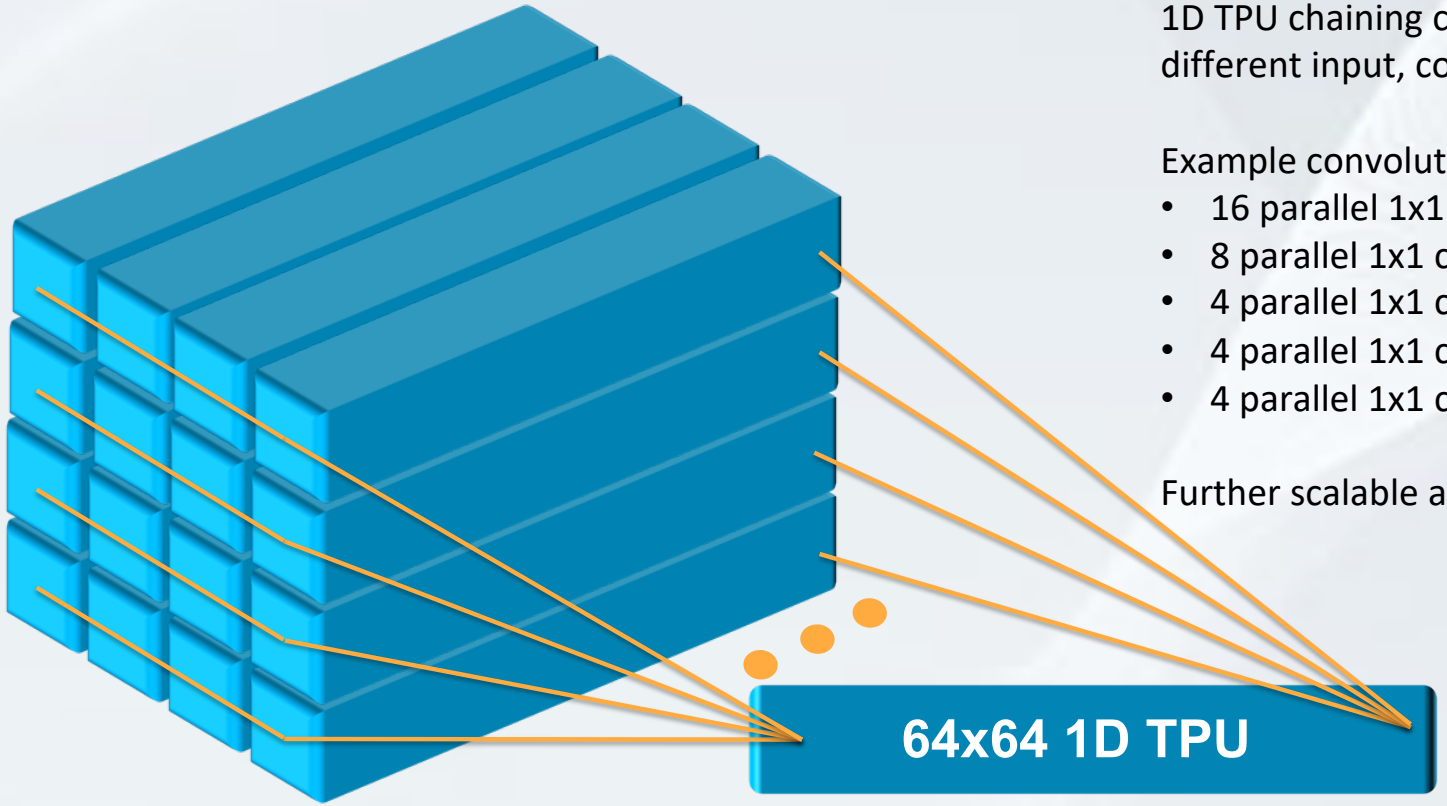
Reconfiguration done through Softlogic in microseconds

Dynamic TPU Array Approach

64 Int8 MACs per TPU

16 TPUs per tile, 4 tiles in X1

TPU's configured for 1x1 operator



1D TPU chaining can be reconfigured to support different input, compute, and output dimensions

Example convolutions with 16 1D-TPUs in 1 tile:

- 16 parallel 1x1 convolution of 64x64
- 8 parallel 1x1 convolution of 128x128
- 4 parallel 1x1 convolution of 256x256
- 4 parallel 1x1 convolution of 512x128 or 128x512
- 4 parallel 1x1 convolution of 1024x64 or 64x1024

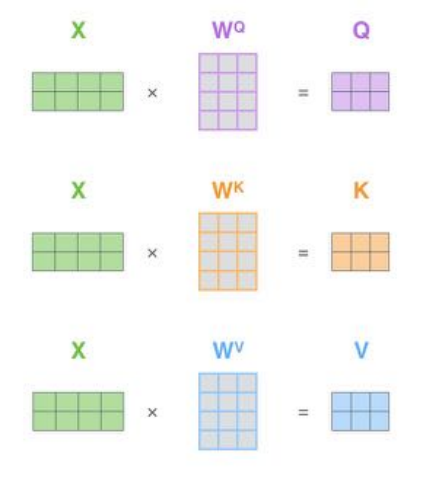
Further scalable across 4 tiles in the X1 chip

Dynamically Reconfigured to Support different operators

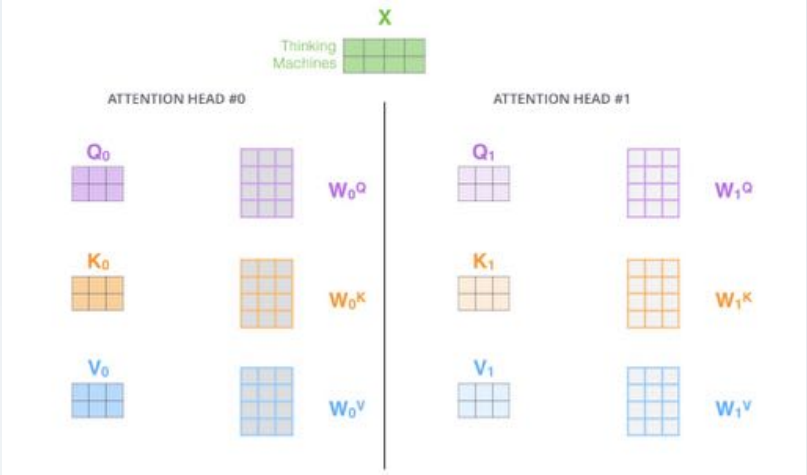
Transformer vs. Traditional CNN

- Traditional CNNs uses simpler “head” (e.g. NMS via Softmax) after the CNN “backbone”
  - More advanced models like YOLO or SSD are more sophisticated, but still feasible on host CPU
- Transformer’s computational complexity far exceeds what host processor can deliver

First the CNN output (X) are multiplied by 3 sets of matrices (W^Q, W^K, W^V)



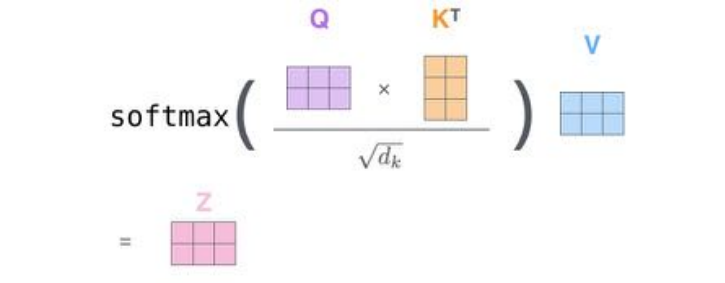
Or in reality, there’s multiple “attention heads” so there’s multiple sets of (W_0^Q, W_0^K, W_0^V) to (W_N^Q, W_N^K, W_N^V)

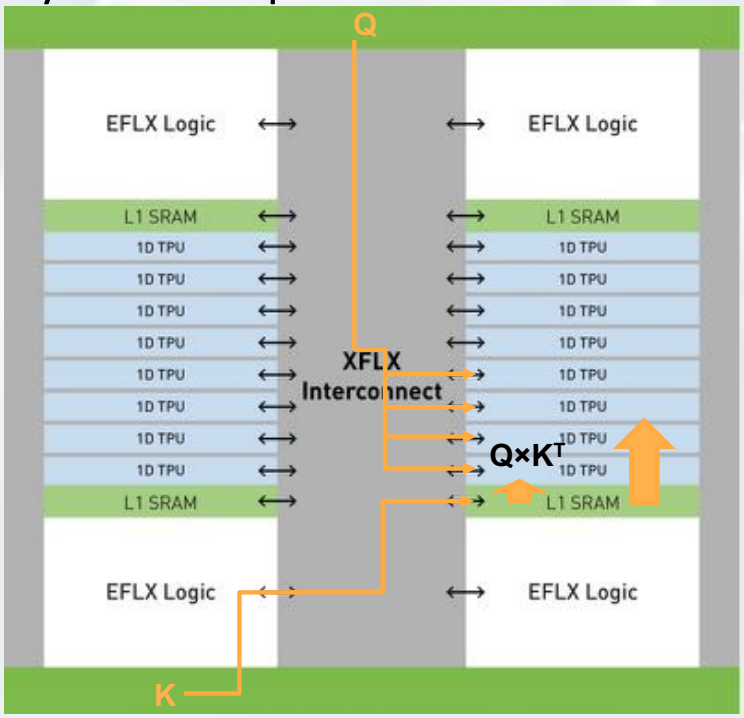


*https://jalammar.github.io/illustrated-transformer

Transformer vs. Traditional CNN (2)

- Transformer's complexity of HW accelerator mostly occur in the second step
- Intermediate results Q, K, and V are all activations, but they are multiplied with **each other**



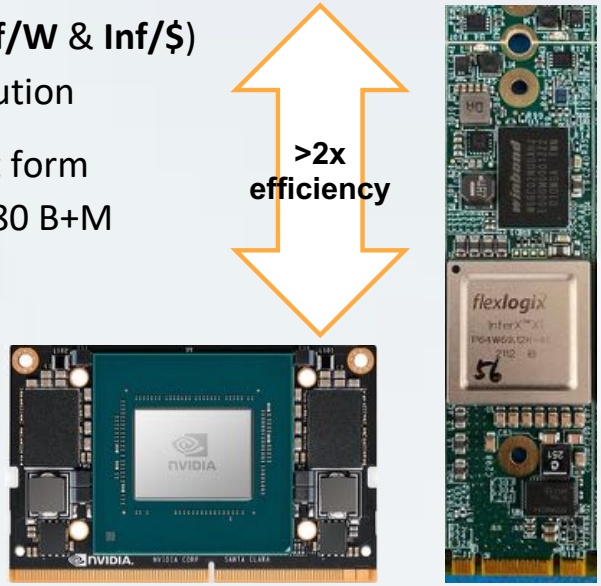


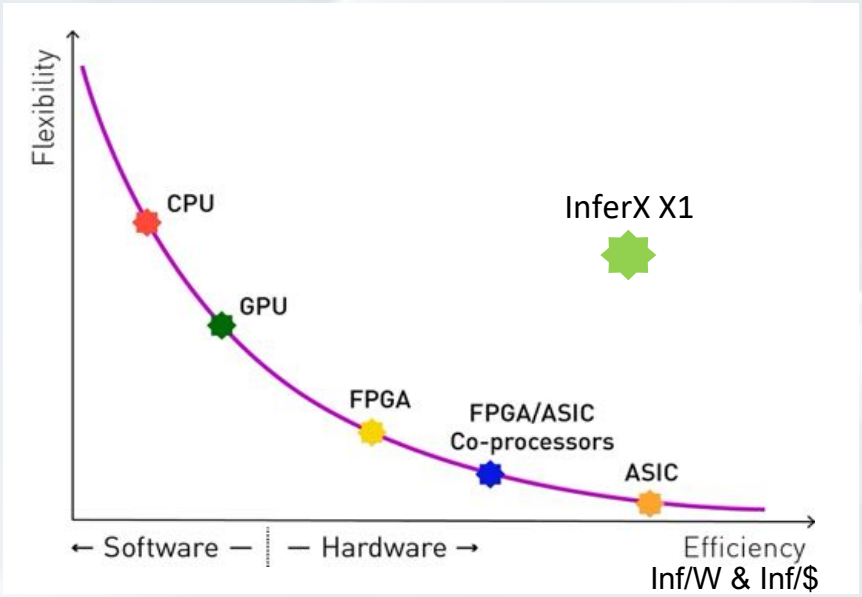
- X1's reconfigurability is ideal for transformers:
  - Dynamically load activation data into weight memory
  - Broadcast of activation into multiple 1D TPU for parallel compute
  - EFLX Logic useful for Softmax and Layer-norm functions, which run poorly on most accelerators but efficiently on X1
- Efficient transformer implementation allows for even more complex transformers to trade off for simpler CNN backbone

*https://jalammar.github.io/illustrated-transformer

The InferX X1 value proposition

- X1 provides **ASIC performance**/efficiency with flexibility of software
- InferX SDK directly converts TensorFlow graph model to **dynamic InferX hardware instance**
- Much more flexible & future proof vs ASIC solutions
- Much higher efficiency (**Inf/W** & **Inf/$**) vs CPU and GPU based solution
  - Thus enabling compact form factors such as M.2 2280 B+M





2019: Xin Feng, Computer vision algorithms and hardware implementations: A survey

Thank You

flex-logix.com

Flex Your Computing