PHASE 2: INITIAL DEPLOYMENT

Team 13, 19

Project Team 8

Raidah Fairooz 862168324

Rania Saber 862228874

Muneeb Ghori 862190576

Uzair Memon 862311538
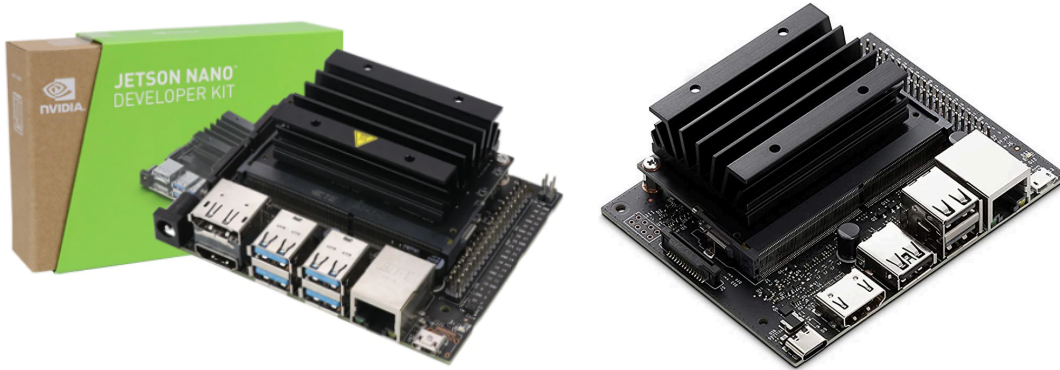
HAND GESTURE RECOGNITION

Detailed Network Diagram

CLOUD LAYER
Device: MacBook Pro

Task: Upload User data on hand gestures (accuracy, confidence, runtime) to Google Cloud. Use Cloud Storage API to store. Provide model Feedback

FOG LAYER
Device: Router

Task: Send metadata(user hand signs) of Nano 1 to Nano 2

EDGE LAYER
Device: Jetson Nano 1 (4GB) & Jetson Nano 2 (2GB)

Task: Real time scan and detect hand signs with Alexnet. Interact via Fog to the other Nano

Device Catalog

1) JETSON NANO (2 GB and 4GB)

Features / Specs	Jetson Nano Dev Kit 2GB (without 802.11ac Wireless Adapter)	Jetson Nano Dev Kit B01 (4GB)
Release Date	October 2020	January 2020
CPU	Quad-core ARM A57	
GPU	128-core Maxwell™ GPU	
Memory	2GB 64-bit LPDDR4 | 25.6 GB/s	4GB 64-bit LPDDR4 | 25.6 GB/s
Storage	MicroSD (not included)	
Connectivity	Gigabit Ethernet	Gigabit Ethernet, M.2 Key E
Video Encode	4Kp30| 4x 1080p30| 9x 720p30 (H.264/H.265)	
Video Decode	4Kp60| 2x 4Kp30| 8x 1080p30| 18x 720p30| (H.264/H.265)	
Power	• USB Type-C 5V 3A	• Micro-USB 5V 2A • DC power adapter 5V 4A • PoE (Power over Ethernet)
Display	HDMI	HDMI, DisplayPort
Camera	1x MIPI-CSI camera connector	2x MIPI-CSI camera connector
I/O Ports	GPIOs, I2C, I2S, SPI, UART	
USB Ports	1x USB 3.0, 2x USB 2.0, 1x USB 2.0 Micro-B	4x USB 3.0, 1x USB 2.0 Micro-B
Others	Fan connector	
Dimensions	100mm x 80mm x 29mm	

2) MACBOOK PRO 13 INCH

Chip

- Apple M1 chip
  8-core CPU with 4 performance cores and 4 efficiency cores
  8-core GPU
  16-core Neural Engine

Battery and Power1

- 61W USB-C Power Adapter

Memory

- 8GB
  8GB unified memory
  Configurable to: 16GB

Storage

- 256GB SSD

Wireless

- Wi-Fi
  802.11ax Wi-Fi 6 wireless networking
  IEEE 802.11a/b/g/n/ac compatible
- Bluetooth
  Bluetooth 5.0 wireless technology

Camera

- 720p FaceTime HD camera

Operating Requirements

- Frequency: 50Hz to 60Hz
- Operating temperature: 50° to 95° F (10° to 35° C)
- Storage temperature: -13° to 113° F (-25° to 45° C)

3) LOGITECH C922 1080p WEBCAM

Resolution, Frame Rate and Field of View
  ● Supports Full HD 1080p Video at 30 fps/ 720p at 60 fps
  ● Wide 78 degree Diagonal Field of View

System Connection
  ● USB-A 2.0

Power
  ● Bus Powered

Compatibility
  ● Includes Microphone
  ● Windows 7 and up, MacOS 10.10 and up, Android 5 and up, Chrome OS

Detailed Network Report + Progress

Plan:

      Our goal for Phase 2 is to set up the base project so we can expand upon it with our data and additional features. The base project is a rock-paper-scissors game with image and hand gesture recognition using Alexnet. We will be implementing a Cloud Storage API on GCP to store and update the user's hand gestures and their respective symbols. We will also add in another device to extract the metadata from one jetson nano to the other via our Fog layer.

Challenges and Solutions:

      During the initial deployment of the project, we ran into major issues with our device since the project is not custom made for the cameras that we are using. Without the camera being recognized, we could not deploy the project at all. The original project was made using the CSI cameras and is using Alexnet for image classification that we are not familiar with. We flashed another SD card with the project to keep things separate and have enough space for our program to run. We used that card to set up the environment on our Nano 1 (4GB). To resolve the issues with the camera, we had to be well versed in the existing code. After successfully deploying the jupyter notebook on our secondary device (laptop), we were able to make changes to the code in real time. We edited the *train_model.ipynb, test_game.ipynb* to be compatible with our Logitech USB webcam (connected to 4gb nano).

      Our project uses *AlexNet* instead of Detectnet for image classification as it offers superior performance in that aspect. While AlexNet doesn't inherently address security and privacy concerns, we have chosen to store our data on a trusted and secure cloud platform in order to protect the privacy of user data. By taking advantage of the cloud infrastructure, we can benefit from the robust security measures and protocols already implemented by the cloud service provider to keep our data secure.

Name and Description of Services

      We will be using *AlexNet*, a convolutional neural network architecture. It consists of eight layers, five are convolutional and three are fully connected. The convolutional layers are used for extracting hierarchical features from input images. The last three layers of AlexNet are fully connected layers. The first two fully connected layers consist of 4,096 neurons each, followed by a final fully connected layer with 1,000 neurons, representing the classes in the ImageNet dataset.

      We will also be using Google Cloud Storage API. The services included in Google Cloud can be used for storage, compute, machine learning, natural language processing, and more.

These APIs enable developers to build powerful and scalable applications on Google Cloud infrastructure while leveraging Google's expertise in various domains.