Choose the Right Hardware

For the scenarios, refer to
https://docs.google.com/document/d/1MhUL2eMknBYC7SAr8swcspw3fBzZFjB6eyhuZvi0bNc/edit?usp=sharing

Scenario 1: Manufacturing

Client Requirements and Potential Hardware Solution

Personal Note:

- To check the supported device specs refer to https://docs.openvinotoolkit.org/2019_R3/_docs_IE_DG_supported_plugins_Supported_Devices.html
- **CPU**: Intel® Xeon® with Intel® AVX2 and AVX512, Intel® Core™ Processors with Intel® AVX2, Intel® Atom® Processors with Intel® SSE
- **GPU**: Intel® Processor Graphics, including Intel® HD Graphics and Intel® Iris® Graphics
- **FPGA**: Field Programmable Gate Arrays, are able to be further configured by a customer after manufacturing. Hence the "field programmable" part of the name
- **VPU**: Vision Processing Units, are going to be like the Intel® Neural Compute Stick. They are small, but powerful devices that can be plugged into other hardware, for the specific purpose of accelerating computer vision tasks
- To understand floating point **FP32, FP16, INT8, and their trade-off**: http://www.principledtechnologies.com/benchmarkxprt/blog/2019/09/05/understanding-the-basics-of-aixprt-precision-settings/

Which hardware might be most appropriate for this scenario? (CPU / IGPU / VPU / FPGA)
FPGA

Requirement Observed (Include at least two.)	**How does the chosen hardware meet this requirement?**
Each camera records video at 30-35 FPS	Before testing, FPGA is expected to have a high bandwidth of data as input.
The images processing task to be completed five times per second	FPGA is expected to superior to CPU and GPU because it yields the shortest time between input and response time.
Repurpose the system to address a second issue, the semiconductor chips being packaged for shipping have flaws	FPGA allows the customer to further configured afterward. The engineering cost can be high since it used machine languages.

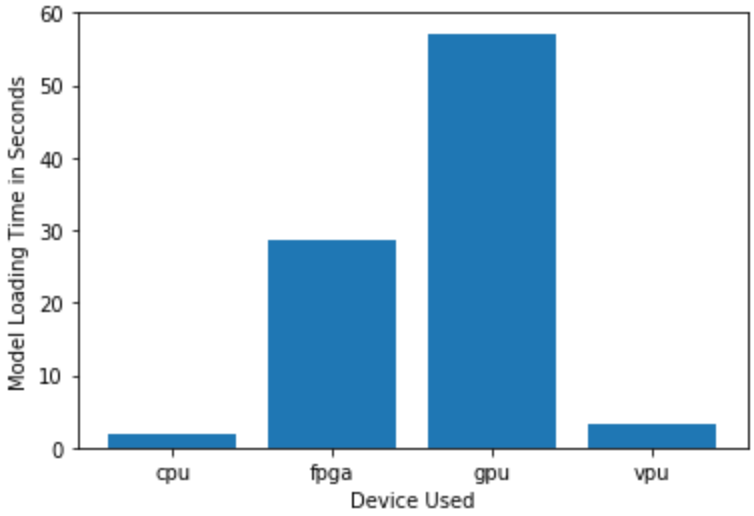
To detect the flaw, the system would need to be able to run inference on the video stream very quickly	FPGA is expected to yield low inference time because of its high performance computing

Source: https://blog.esciencecenter.nl/why-use-an-fpga-instead-of-a-cpu-or-gpu-b234cd4f309c
GPU vs FPGA: https://pqdtopen.proquest.com/doc/1844966839.html?FMT=ABS

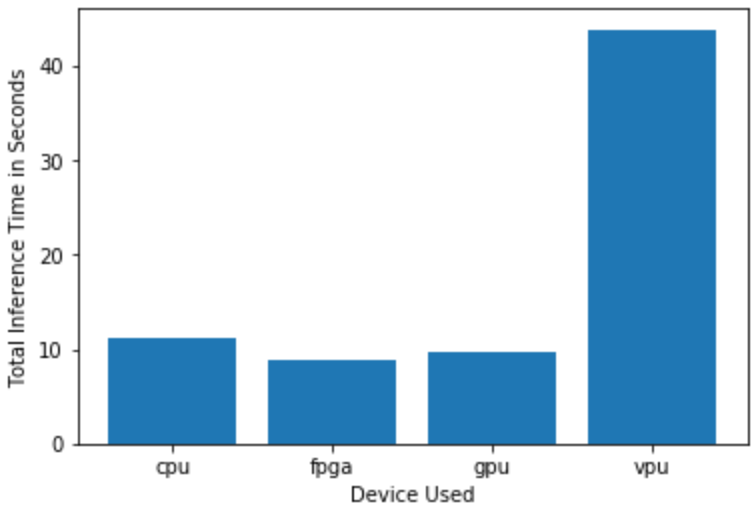
Queue Monitoring Requirements

Maximum number of people in the queue	5 (Up to customer)
Model precision chosen (FP32, FP16, or Int8)	FP16

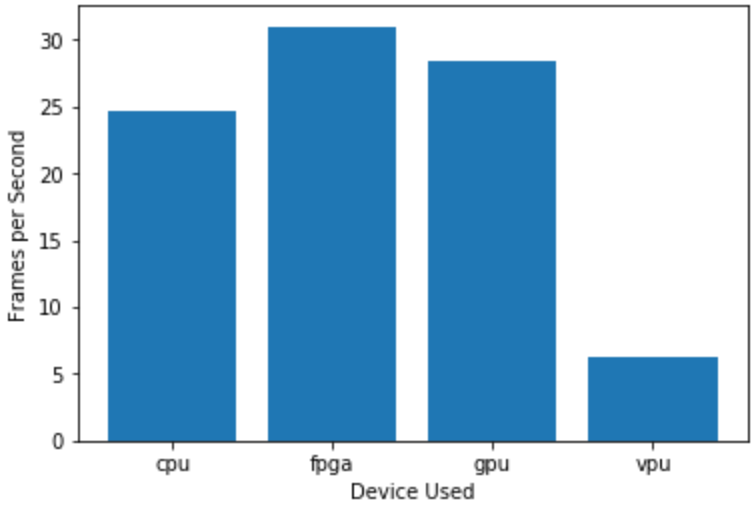
Test Results



Model Load Time



Inference Time



FPS

Final Hardware Recommendation

Write-up: Final Hardware Recommendation
From the test results, FPGA can load the model in a decent time. But the most important is it can run at 30 FPS and the Inference time is 10. These results proved that FPGA is a good candidate for this situation. Also, FPGA has a long life-span and ability to reconfigure to a specific purpose of computing. FP16 is a decent choice for input. If the customer switches to the second task, the preference is FP32.

Scenario 2: Retail

Client Requirements and Potential Hardware Solution

Which hardware might be most appropriate for this scenario? (CPU / IGPU / VPU / FPGA)
CPU

Requirement Observed (Include at least two.)	How does the chosen hardware meet this requirement?
The customer wants to optimize the cost of hardware infrastructure and electric consumption	Using CPU from the customer so the customer does not need to pay for hardware. The electric consumption stays merely the same
The average time spent is 40 mins at the store and 350-400 seconds at the checkout line.	CPU can get the job done just in time since there is a lot of time to inference

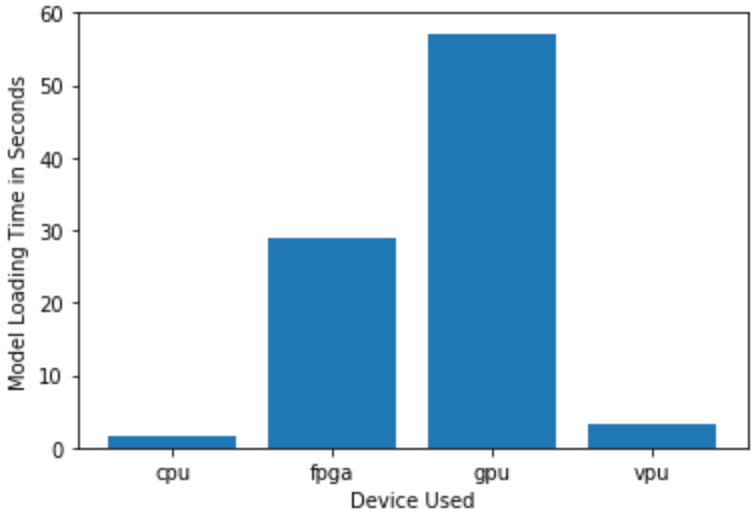
Most of the store's checkout counters already have a modern computer, each of which has an Intel i7 core processor. Currently these processors are only used to carry out some minimal tasks that are not computationally expensive.	One CPU is good enough to control the counter and the CPU has enough resources to get the job done.

Queue Monitoring Requirements

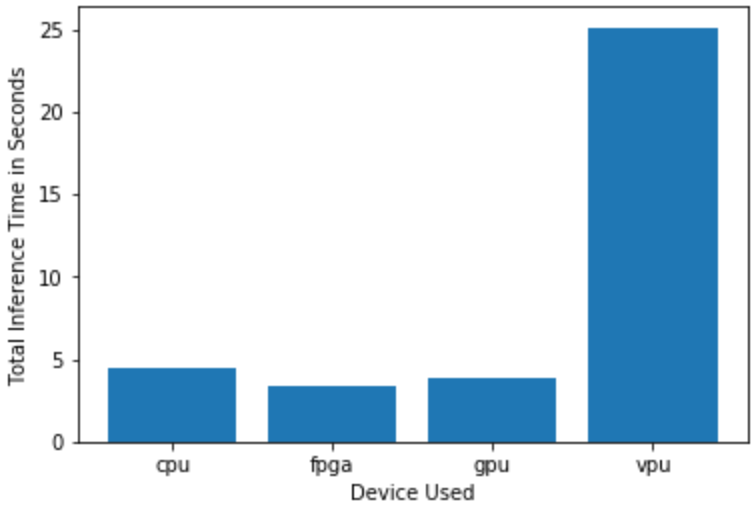
Maximum number of people in the queue	5
Model precision chosen (FP32, FP16, or Int8)	FP16

Test Results

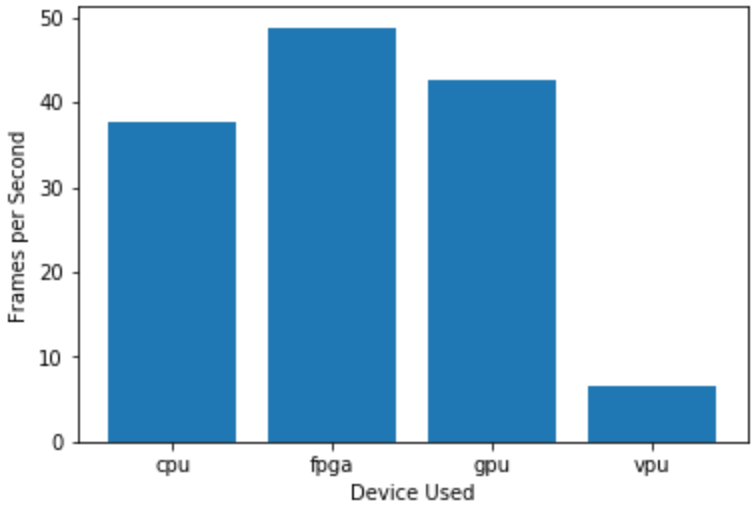
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).



Model Load Time



Inference Time



FPS

Final Hardware Recommendation

Write-up: Final Hardware Recommendation
According to the test result, the CPU has the shortest loading model time. With the requirement that the customer wants to use the CPU, the CPU can get the job done in this case. Also, the inference time of the CPU is not too bad, this fits the requirement that the system does not need to be very responsive to the situation. Lastly, the CPU can process up to 35 FPS. The FPS is very good compared to the requirement.

Scenario 3: Transportation

Client Requirements and Potential Hardware Solution

Look through the scenario and find any relevant client requirements. Then, suggest a potential hardware type and explain how this hardware would satisfy each of the requirements.

Which hardware might be most appropriate for this scenario? (CPU / IGPU / VPU / FPGA)
VPU (with customer's CPU)

Requirement Observed (Include at least two.)	How does the chosen hardware meet this requirement?
The budget allows for a maximum of $300 per machine with saving as much as possible both on hardware and future power requirements.	The latest Raspberry Pi with a Neural Compute Stick and other add-ons should cost around $150 to

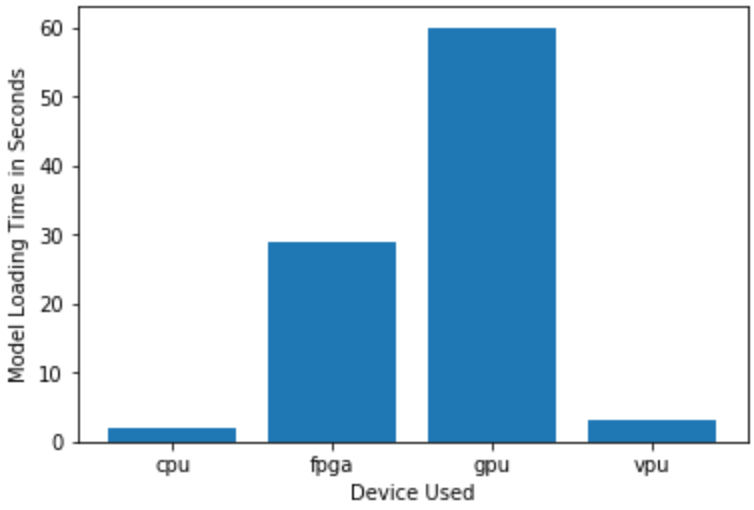
	$200. It only requires engineer to reprogram the purpose or upgrade the discrete components.
The CPUs in these machines are currently being used to process and view CCTV footage for security purposes and no significant additional processing power is available to run inference	The Neural Compute Stick is compatible with CPU Intel-chip based. More work required if it is not Intel. The CPU should have enough or need an add-on to handle the inference task.
In peak hours they currently have over 15 people on average in a single queue outside every door in the Metro Rail. But during non-peak hours, the number of people reduces to 7 people in a single queue. On office hours there is a train every 2 mins. However, on the weekends the time increases to up to 5 mins since some of their drivers work only 5 days a week.	The inference time requirement is not too high or too low. Low latency is recommended but it is not too strict.

Queue Monitoring Requirements

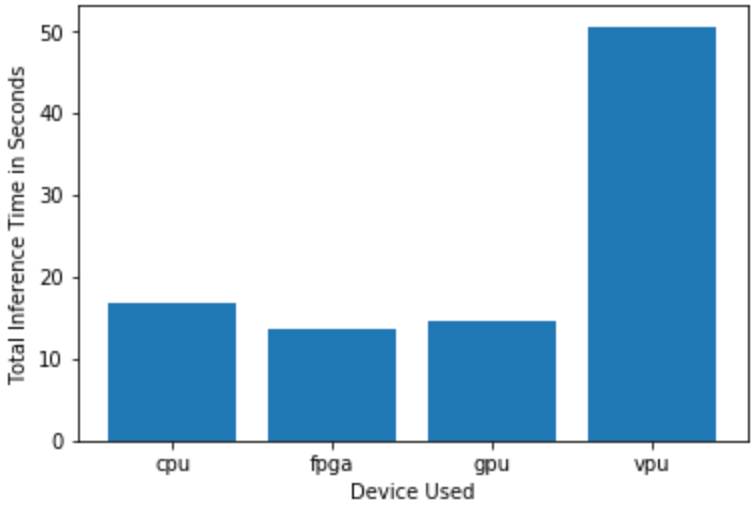
Maximum number of people in the queue	15
Model precision chosen (FP32, FP16, or Int8)	FP16

Test Results

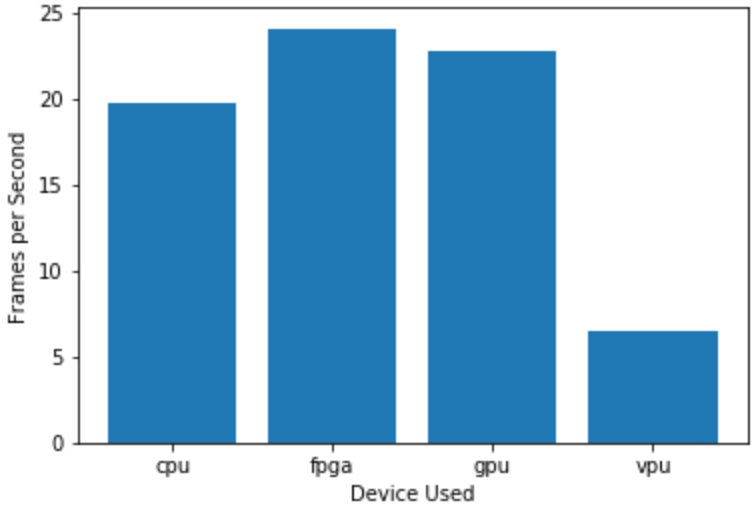
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).



Model Load Time



Inference Time



FPS

Final Hardware Recommendation

Write-up: Final Hardware Recommendation
According to the test result, VPU has a very low loading model time. Thus, the VPU can make it in time. With the inference time being very long, it is acceptable since the system does not need to be very responsive but an upgrade is optional for the future. However, the VPU has the lowest FPS, the VPU still can get the job done. In conclusion, because there is a strict budget to follow so the choice is VPU with its ability can get the job done.