# Choose the Right Hardware

*Proposal Template*

---

## Scenario 1: Manufacturing

### 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)** |
|---|
| *CPU and FPGA* |

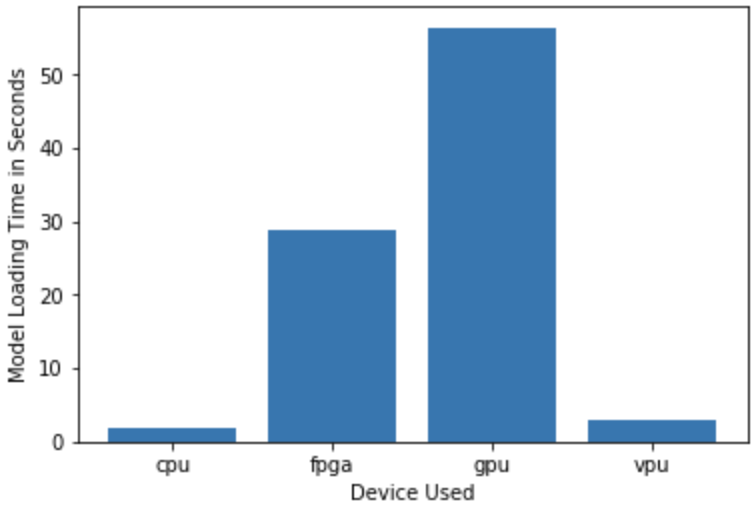
| **Requirement Observed (Include at least two.)** | **How does the chosen hardware meet this requirement?** |
|---|---|
| Performance | *The client wants image processing tasks to be completed 5 times per seconds and be able to run inference on the video stream very quickly. Also the performance metrics shows that FPGA has the fastest inference time than other devices. However CPU provides better results than FPGA in terms of frames per seconds. So therefore, it is recommended to use CPU and FPGA for this application.* |
| *Reprogrammable* | *The client wants the system to be flexible such that it can be reprogrammed and optimized to quickly detect flaws in different chip designs. FPGA provides a better option for this since it is more flexible and can easily be reprogrammed.* |
| *Long Term Solution* | *The client also wants the system to last for at least 5-10 years.* |
| *Economic Constraint* | *The client wants to save on this investment since it is a significant investment* |

### Queue Monitoring Requirements

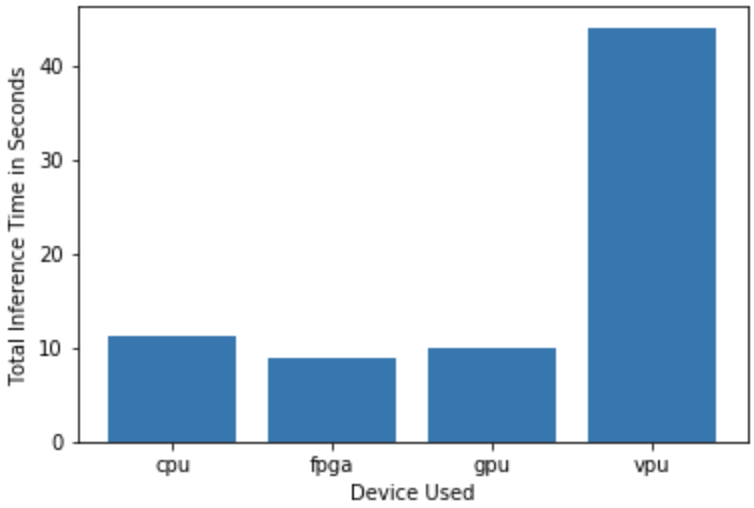
| | |
|---|---|
| **Maximum number of people in the queue** | *2* |
| **Model precision chosen (FP32, FP16, or Int8)** | *FP16 (It is preferred to use FP32 for CPU but I use FP16 since FPGA uses model precision FP16)* |

## Test Results

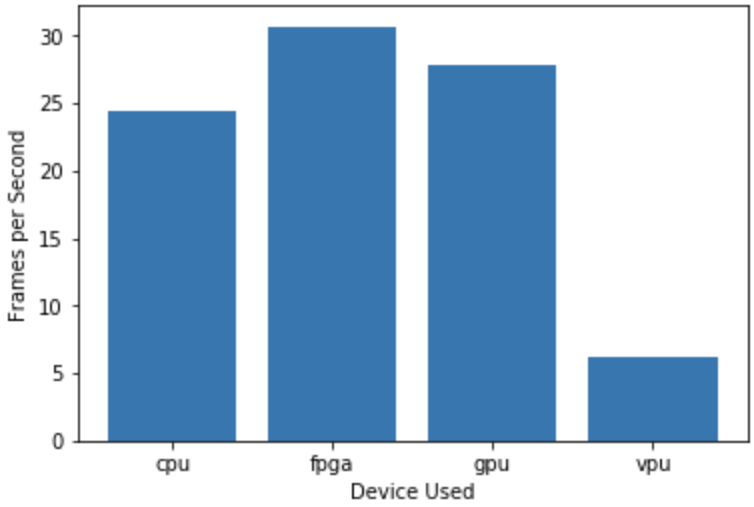
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).




***Model Load Time***




***Inference Time***




***FPS***

## Final Hardware Recommendation

Now synthesize your points from above and provide a brief write-up describing why the chosen hardware is the best choice for this scenario. Be sure to discuss the client's requirements, the test results, and how these relate to one another (e.g., perhaps one of the devices performed better than the rest, but does not meet one of the client's requirements).

| **Write-up: Final Hardware Recommendation** |
| --- |
| *Based on client requirements and experimental results, I recommend CPU and FPGA for the following reasons. The FPGA provided the fastest inference time of 9.0 seconds compared to other devices and since one of the client's requirements was to be able to run inference on video stream very quickly. However, CPU tends to perform better in terms of frames per seconds (FPS) and model loading time which is also an important requirement for the client. Therefore I recommend CPU and FPGA as the best devices to deploy the Smart Queueing System for the client.* |

---

# Scenario 2: Retail

## 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?<br>(CPU / IGPU / VPU / FPGA)** |
| --- |
| *CPU and GPU* |

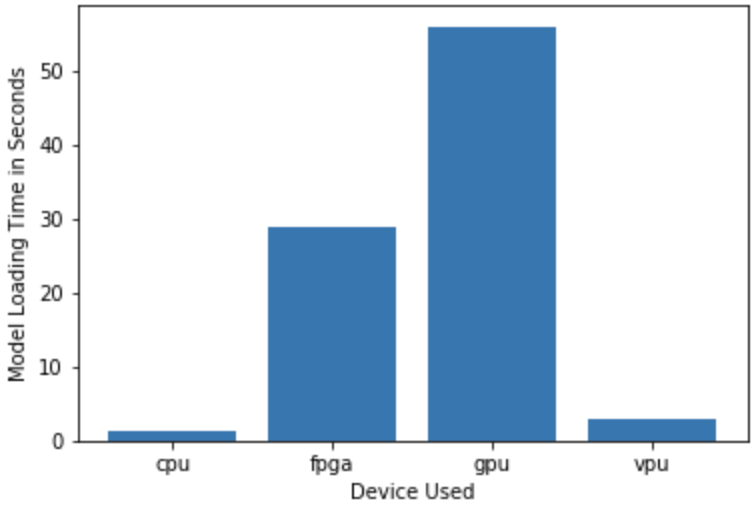
| Requirement Observed (Include at least two.) | How does the chosen hardware meet this requirement? |
|---|---|
| *Cost constraint* | *Integrated GPU is highly recommended to the client for the following reasons: first the client has already purchased computers with Intel i7 core processor which is currently not used for computationally expensive tasks. So using the IGPU would offer the best result since the client does not have much money to invest in additional hardware.* |
| *Energy/power constraint* | *Another requirement of the client is the ability to save as much as possible on electricity bills. Therefore using the IGPU would allow the client not to incur additional electricity cost from what they currently pays.* |
| *Performance* | *From the experimental results, it was observed that the GPU has the highest model loading time compared to other devices, however since the clients requirement is mainly to save cost and the client have already Intel i7 core processor, it is recommended to use CPU for model loading and GPU for running inference on the video stream.* |
| *Long term* | *Using these devices will also allow the application to last longer* |

## Queue Monitoring Requirements

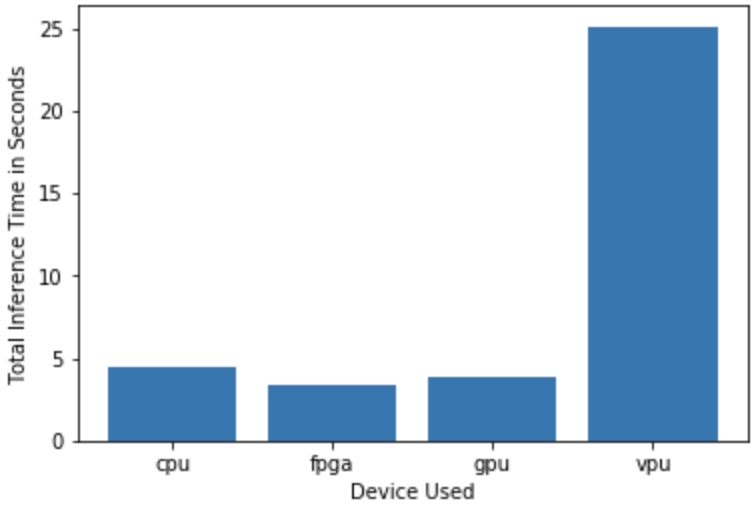
| | |
|---|---|
| **Maximum number of people in the queue** | *2* |
| **Model precision chosen (FP32, FP16, or Int8)** | *FP16 (It is preferred to use FP32 for CPU but I use FP16 since GPU uses model precision FP16)* |

## Test Results

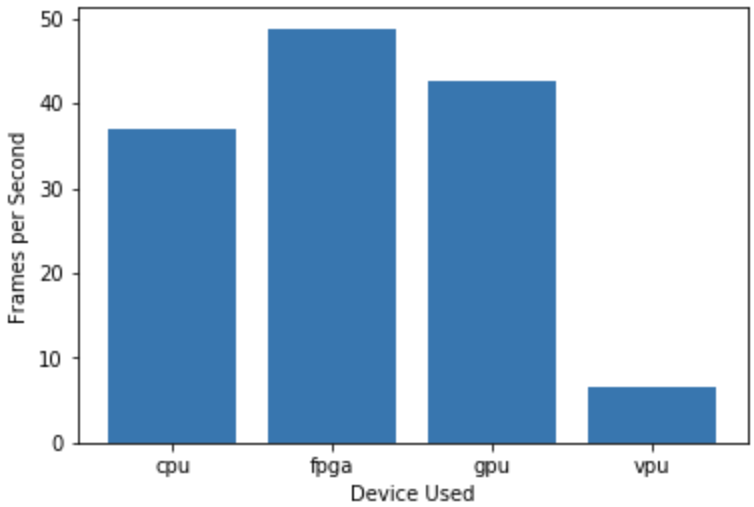
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).




***Model Load Time***




***Inference Time***




***FPS***

## Final Hardware Recommendation

Now synthesize your points from above and provide a brief write-up describing why the chosen hardware is the best choice for this scenario. Be sure to discuss the client's requirements, the test results, and how these relate to one another (e.g., perhaps one of the devices performed better than the rest, but does not meet one of the client's requirements).

| Write-up: Final Hardware Recommendation |
| --- |
| *The most important requirement of the client is to be able to redirect customers to less-congestion queues in the store at the least minimal cost. Since the client has already purchased Intel i7 core processors that are currently used for less computationally expensive tasks, I recommend using the CPU and GPU for the deployment of a smart queueing system. These devices recommended have been validated by the experimental results. In the experimental results above, it is observed that using FPGA will provide the fastest inference time on the video streams and using the CPU will provide the fastest model loading time.* |

---

# 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?<br>(CPU / IGPU / VPU / FPGA) |
| --- |
| *CPU and VPU* |

| Requirement Observed<br>(Include at least two.) | How does the chosen hardware meet this requirement? |
| --- | --- |
| *Processing Power and Space Requirement* | *The first choice of device to be recommended to the client is the VPU for the following reasons. The client mentioned that the 7 CCTV cameras on the rail station are connected to All-In-One Pcs and that there is no significant additional processing power available to run inference. Therefore it will be wise to use an external device (VPU) to run inference on the video streams as this will free up space from the All-In-One Pcs.* |
| *Low Cost* | *Another constraining factor mentioned by the client is the limited budget available and they want to save as much as possible. Again the VPU device is much cheaper device to deploy the smart queueing system* |

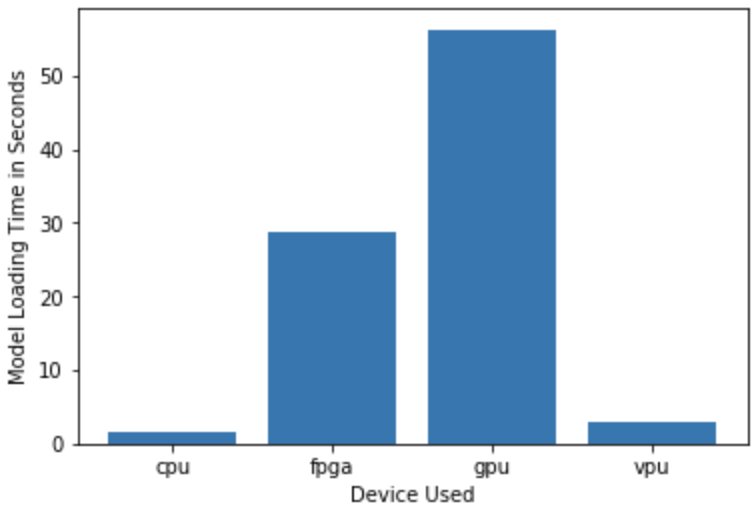
| | |
|---|---|
| | *since it costs from about $39 to $69 compared to the rest which are more costly in the range above hundreds of dollars.* |
| *Low Power Usage* | Also the client wants to save on future power requirements. VPU is an extremely low power device that fits into the client's requirement perfectly, however this can come at some cost to performance compared to other devices. |
| *Performance* | *As observed, performance is one of the least requirements for the client, therefore using VPU to deploy the smart queueing system for the client will save substantial cost for the client but at a low performance in terms of running inference on the video stream as can be seen in the experimental graph below. Therefore having the CPU as a fallback is common practice.* |

## Queue Monitoring Requirements

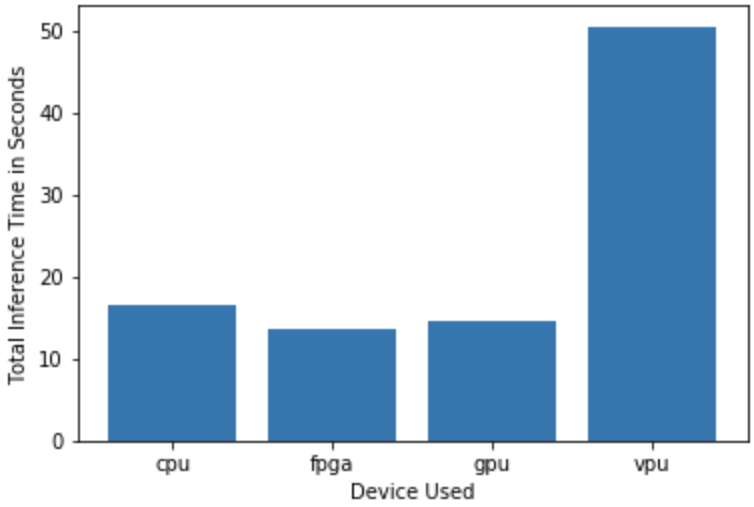
| | |
|---|---|
| **Maximum number of people in the queue** | *5* |
| **Model precision chosen (FP32, FP16, or Int8)** | *FP16 (It is preferred to use FP32 for CPU but I use FP16 since VPU uses model precision FP16)* |

## Test Results

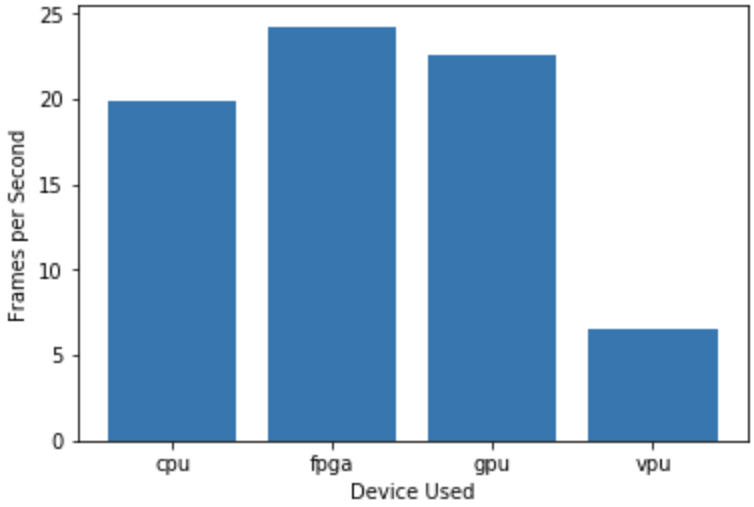
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).




***Model Load Time***




***Inference Time***




***FPS***

## Final Hardware Recommendation

Now synthesize your points from above and provide a brief write-up describing why the chosen hardware is the best choice for this scenario. Be sure to discuss the client's requirements, the test results, and how these relate to one another (e.g., perhaps one of the devices performed better than the rest, but does not meet one of the client's requirements).

| **Write-up: Final Hardware Recommendation** |
| --- |
| *One major requirement of the client is the ability to deploy the smart queueing system without consuming additional processing power while running inference since the 7 CCTV cameras are already connected to closed All-In-One PCs. Also the client would like to deploy the application with a limited budget, save on hardware and future power requirements. Based on these main requirements, the best device to recommend to the client is the VPU for many reasons. VPU is an extremely cheap, low power external device that can be used to run inference on the video stream, thereby freeing up space from the All-In-One PCs. Secondly VOPU is extremely cheap ranging from $39 to $69 per device and utilities extremely low power consumption. However, it is noteworthy* |

*that this can come at some cost in terms of performance when compared to other devices. For in the experimental results above, VPU takes the longest inference time to process the video streams, so as a precautionary measure it is a common practice to indicate CPU as a fallback and also for supported layers.*