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)** |
| Actually, Mr. Vishwas wants two problems to be addressed. The first, and most urgent according to specifications, is the fact that there is a significant drop in production during the shift transition periods. In order to address this and since the requirements are not that strict, one solution could be to use a VPU stick, like the NCS2. In this case the budget will be kept low, and the requested task could be achieved, without any significant changes to be done in the current infrastructure.  Another option, since the company is actually making CPU chips, could be to manufacture one CPU chip 6th generation (or above), which is compatible with OpenVINO (according to the documentation <https://software.intel.com/content/www/us/en/develop/tools/openvino-toolkit/system-requirements.html>, OpenVINO requires at least 6th gen intel core CPU, but the one they are about to manufacture does not fulfil that requirement <https://ark.intel.com/content/www/us/en/ark/products/27497/intel-pentium-4-processor-supporting-ht-technology-3-00-ghz-1m-cache-800-mhz-fsb.html>), or if it is not possible to manufacture such chip, just buy one. Having in mind that they want an inference rate of about 5FPS, the whole inference could be done in the CPU alone, and even if some extra speed was required the IGPU could be used. That extra one time overhead that is required is not that crucial, since the inference could be started during a shift, in which the production rate is ok, the only problematic part is the one during shift transitions. Of course a cloud environment which could provide them with such a hardware through VM is an alternative solution, although this solution requires a constant bill to the cloud service provider.  The second problem is to detect chips that have flaws. According to Mr. Vishwas this must be done extremely quick and also that it should be able to adapt to changes (since there are multiple chip designs and newer model will rise). In this case, the best solution could be to use an FPGA, which can perform inference really fast, has a support period of 10 years and is able to be reprogrammed per customer needs, as it is requested. Of course, in order to achieve that the best case scenario could be to get that extra 6th gen CPU, compatible with OpenVINO, which could provide the extra benefit of the IGPU. |

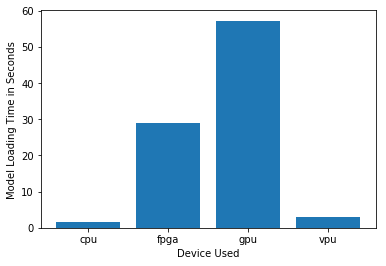
|  |  |
| --- | --- |
| **Requirement Observed**  **(Include at least two.)** | **How does the chosen hardware meet this requirement?** |
| *Example requirement:*  The client requires a tiny device to be connected to their CPU—and their budget is only about $100 for each device. | *Example explanation:*  VPU or NCS2 is only about 27.40 mm in size and would fit in the price range. |
| Monitor number of people in the factory line | Use a NCS2 with the current infrastructure. Client only needs inference speed of 5 FPS, which could probably be easily achieved |
| Monitor number of people in the factory line | Manufacture, or obtain, a 6th generation chip, to run inference on it. Could better be used with the below requirement |
| Check for flaws in the chips, during manufacturing | Use an FPGA, with the 6th gen CPU of the previous step, to run inference and check for flaws. The high inference speed could be achieved, in conjunction with IGPU and CPU utilization. The FPGA have a support of 10 years and can be reconfigured per client needs, as it is requested |

## Queue Monitoring Requirements

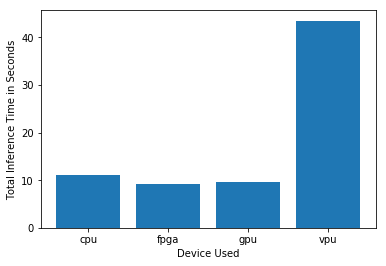
|  |  |
| --- | --- |
| **Maximum number of people in the queue** | Not specified, shift size (since we are checking the shift transition) [From video, each had 5 people] |
| **Model precision chosen (FP32, FP16, or Int8)** | FP32 in case of CPU only, FP16 in the other cases |

## Test Results

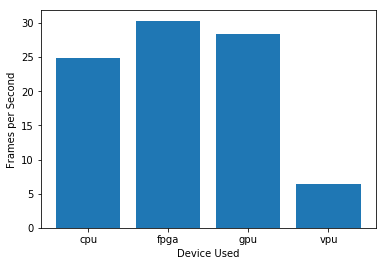
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).

**

***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** |
| Adding a VPU to the current system seems that can cope with the initial requirements, since it can handle an FPS of at least 5 secs, as it was requested, and the budget is kept low. But, when Mr. Vishwas would like to further expand to checking the CPU chips, it won’t be able to cope with it and adding a modern CPU would be the way around. |

# 

# 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?**  **(CPU / IGPU / VPU / FPGA)** |
| Since Mr. Lin already has a computer with a modern I7 CPU, and given the fact that the task that are currently running on them are not that computational heavy, the best choice would be to use either the CPU itself, or the CPU with the boost of the IGPU can provide. I would suggest using MULTI plugin in this case scenario, since the utilization is low and we could benefit from the many free resources. |

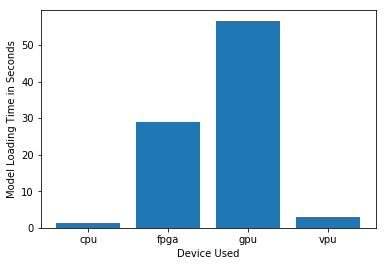
|  |  |
| --- | --- |
| **Requirement Observed**  **(Include at least two.)** | **How does the chosen hardware meet this requirement?** |
| *Example requirement:*  The client requires a tiny device to be connected to their CPU—and their budget is only about $100 for each device. | *Example explanation:*  VPU or NCS2 is only about 27.40 mm in size and would fit in the price range. |
| Low cost in terms of extra infrastructure / equipment | Using what Mr. Lin already possesses does not add any extra cost |
| Save on other extra costs (such as electric bill) | Not adding any new equipment will not add to the current costs (apart the extra power the CPU - with or without the IGPU - will require dues to its utilization) |
| *[TODO: Type your answer here]* | *[TODO: Type your answer here]* |

## Queue Monitoring Requirements

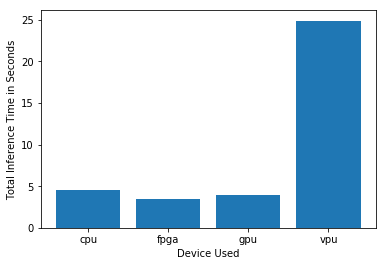
|  |  |
| --- | --- |
| **Maximum number of people in the queue** | Over 5 at peak hours - ideal 3 |
| **Model precision chosen (FP32, FP16, or Int8)** | FP32 in case of CPU only, FP16 if we use the IGPU |

## Test Results

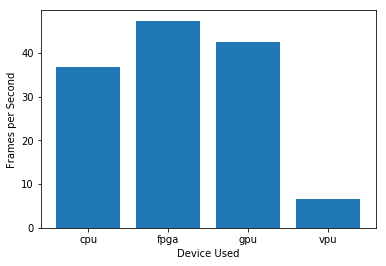
After you've tested your application on all four hardware types (CPU, IGPU, VPU, and FPGA), copy the matplotlib output showing the comparison into the spaces below. You should have three graphs (for model load time, inference time, and FPS).

**

***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** |
| As it was expected CPU alone can handle the requested task, and fulfill the client’s requirements about keeping the cost low and using the current infrastructure. |

# 

# 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)** |
| Since the current infrastructure is highly utilized, and their budget is sort of limited, we could try using, one or probably two, VPUs per machine, alongside with IGPU. We could also add the CPU for inference as a last resort, mainly if there are unsupported layers in the other devices. I would suggest using the HETERO plugin, since we want mainly inference to run on VPU, due to low free resources of CPU. The reason i suggested two VPU’s is that because of the suggestion of about 4 inference request per usb device (in order to hide the usb transfer loss), and each PC is controlling footage of 7 CCTV’s. |

|  |  |
| --- | --- |
| **Requirement Observed**  **(Include at least two.)** | **How does the chosen hardware meet this requirement?** |
| *Example requirement:*  The client requires a tiny device to be connected to their CPU—and their budget is only about $100 for each device. | *Example explanation:*  VPU or NCS2 is only about 27.40 mm in size and would fit in the price range. |
| Current PCs are highly utilized | Adding a VPU to run the inference will not add up to current CPU workload |
| Each PC is responsible for 7 CCTVs | Adding two VPUs will hide the extra usb cost |
| Keep cost and power requirements low | Adding one, or even two, VPUs is in between the client’s budget, and their power requirements are low |

## Queue Monitoring Requirements

|  |  |
| --- | --- |
| **Maximum number of people in the queue** | Over 15 at peak hours - ideal 7 |
| **Model precision chosen (FP32, FP16, or Int8)** | FP16 which is compatible with VPU |

## 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** |
| *[TODO: Type your answer here]* |

# 