

人臉與口罩偵測系統硬體加速架構設計與實現

Hardware Accelerator Implementation for Mask and Face Detection on PYNQ FPGA

組員：電機四 黃志棋、電機四 張耀文

摘要

此專題中我們選用FPGA做為加速器。在此專題中，我們結合影像處理的方法，針對串流影像做處理，經FPGA中的數學運算後，得到結果資料，再用PS端做最後判斷且框出人臉與口罩偵測。並比較「經FPGA運算」與「全部經CPU運算」之所需時間，達成加速之成效。藉由Vivado、Vivado HLS等工具將模型轉化成FPGA可應用之形式，並且利用Python，控制FPGA板之運作。本系統在720P的解析度下(1280x720)，使用hdmi影像格式輸入，實際的fps可達17、18，可以實現多人臉的系統。

系統方法與設計

此次使用的FPGA版本為Xilinx TUL PYNQ™-Z2 board。其內部核心晶片 型號為ZYNQ XC7Z020-1CLG400C，是一塊以Python + Zynq為概念的開發板。板上可以分為PS和PL兩個部分。

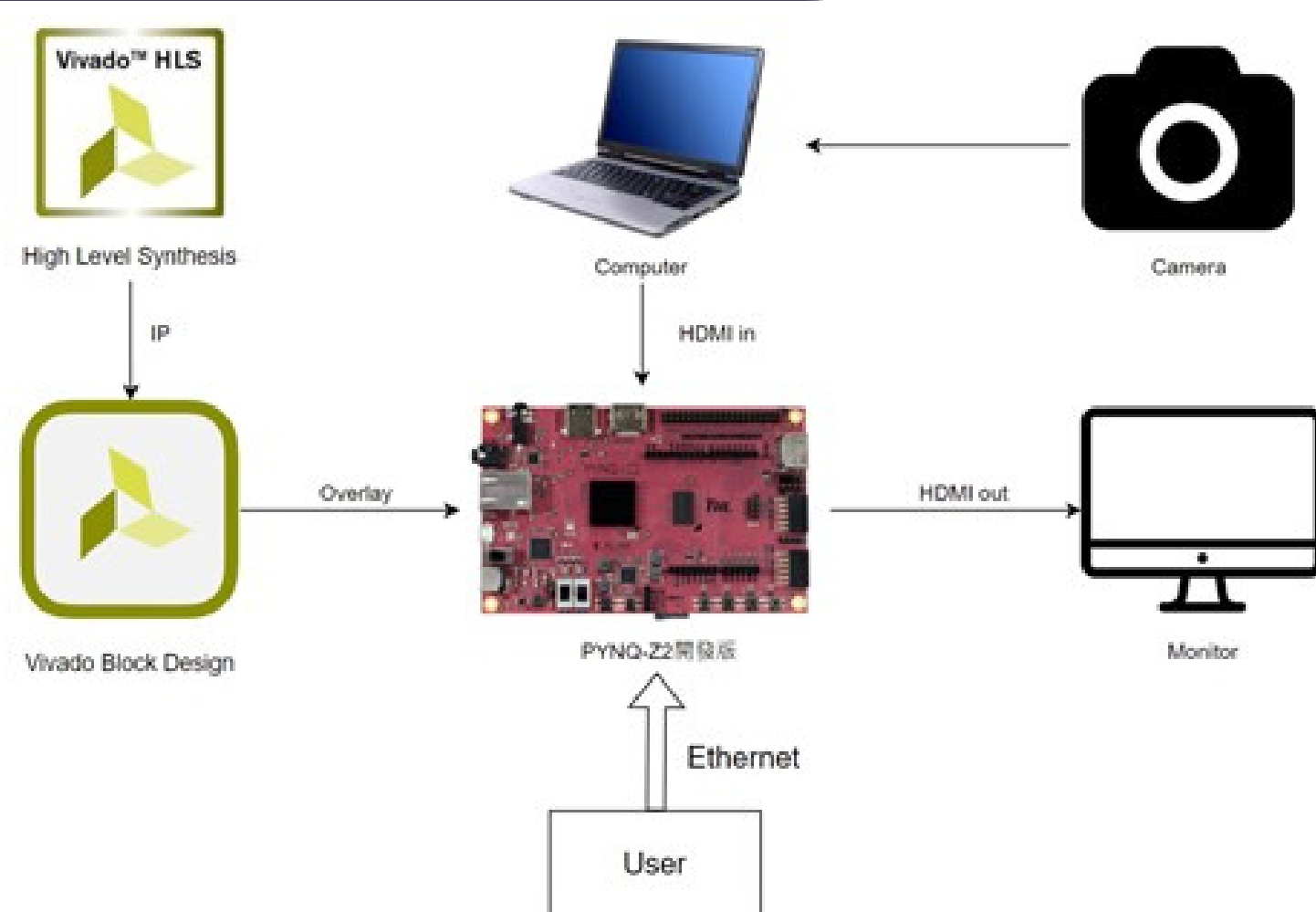
在PL端設計：使用可程式化邏輯硬體資源，由Vivado HLS設計IP後，將需要的演算法設計完成並透過Vivado將IP整合成系統(Block design)，確認無誤後製作bitstream，放入開發版即可調用硬體資源。

在PS端設計：使用ARM架構進行運算，由Ubuntu系統、Python構成，我們在PS端將資料做前處理、後處理。
前處理：將影像資料讀取至FPGA版上，設定好演算法IP的MMIO。
後處理：將在FPGA版上所得出來的結果，進行座標定位，且利用python的套件將座標框起來。

在設計演算法時，我們需要不同影像處理的模組，切分區塊可讓系統明確分工，不僅容易除錯且有助於開發。

系統架構

使用Vivado HLS製作演算法IP，再由Vivado做系統設計(Block design)，製作成Overlay提供PS端調度，PS端部分是負責製作API與控制流程，搭配周邊I/O完成系統架構。



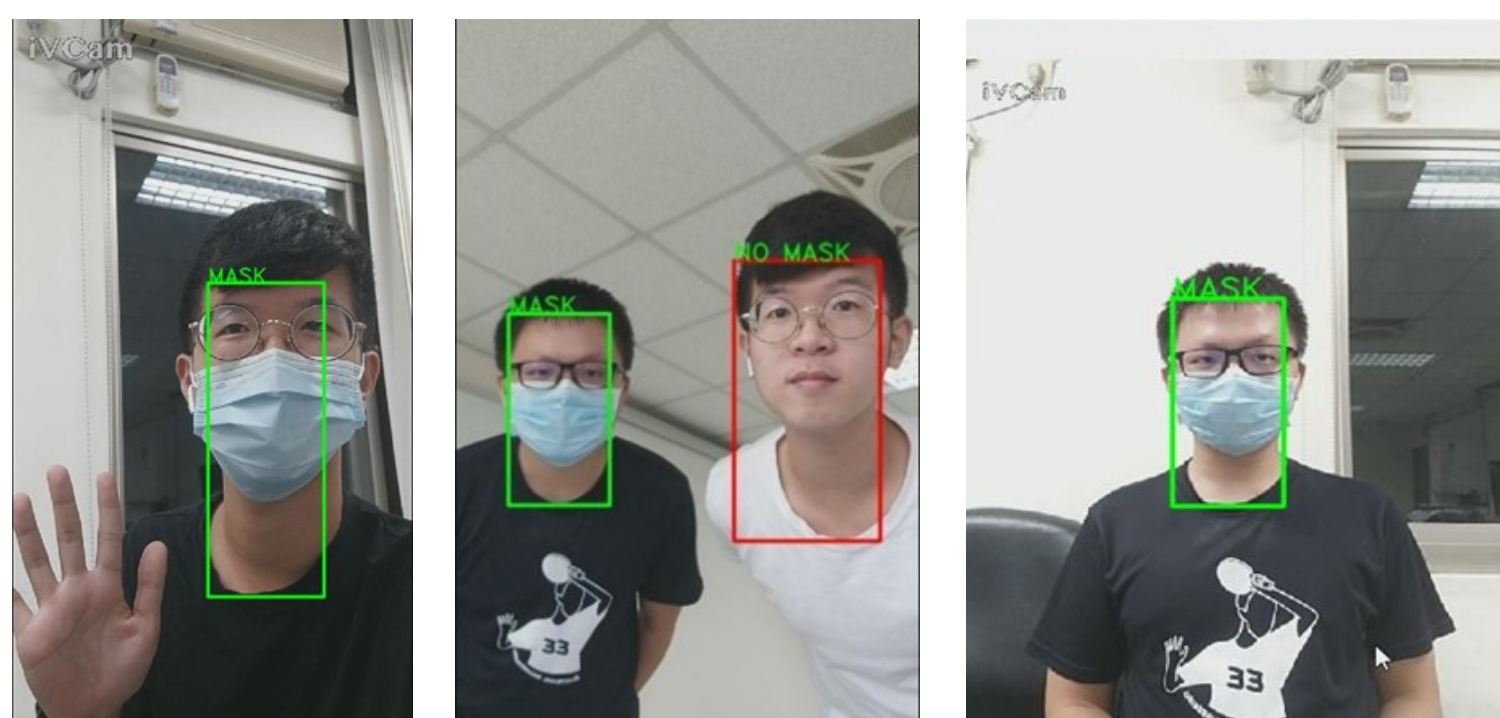
實作成果

硬體使用率

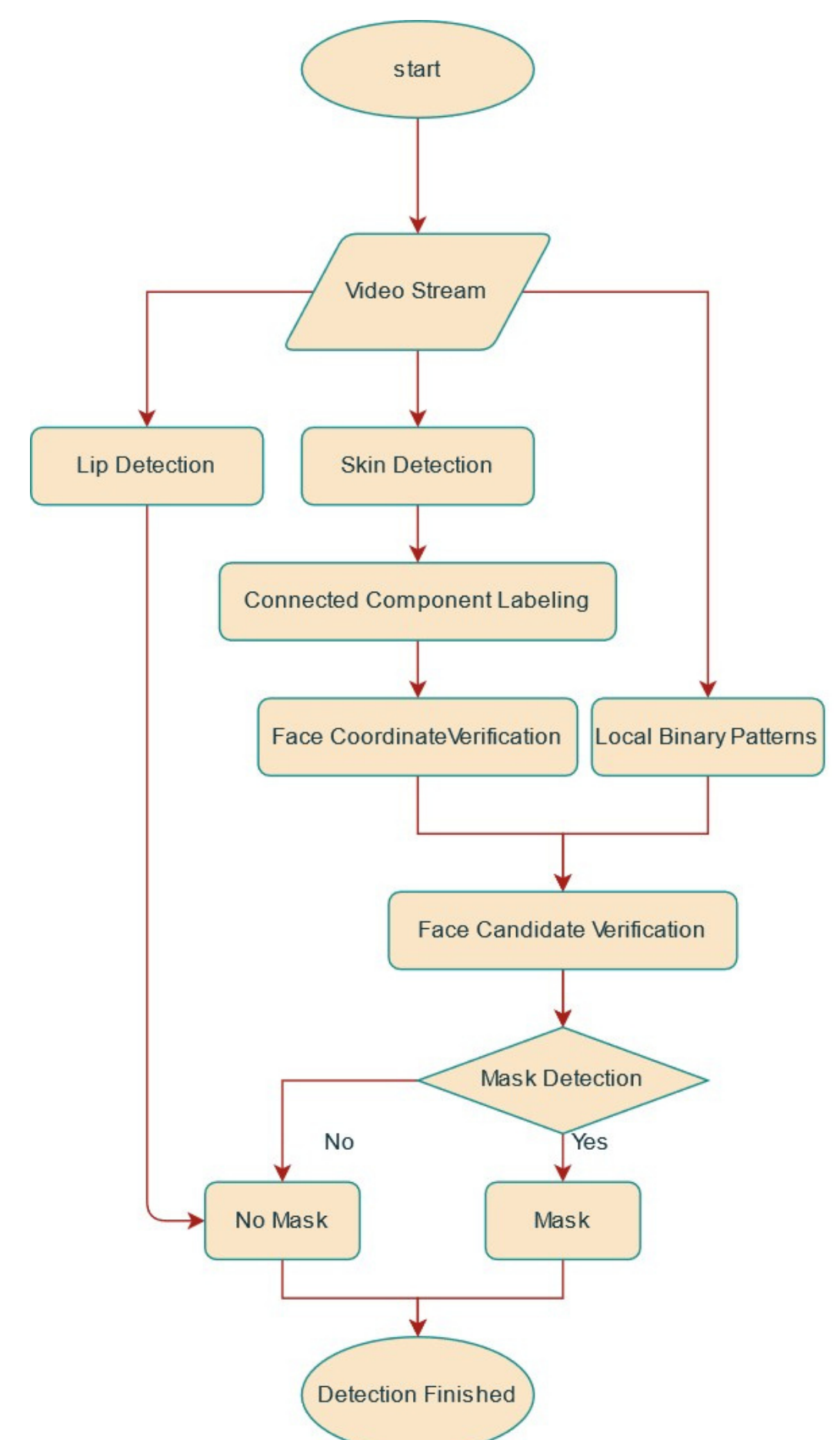
因為處理的資料為720P的串流，在LUT這邊的使用率較高，演算法有使用到侵蝕與膨脹，會使用到BRAM，基本上演算法做的運算不會太複雜，都是乘加器與判斷式較多，DSP使用率僅7.27%。

Resource	Utilization	Available	Utilization %
LUT	41292	53200	77.62
LUTRAM	3472	17400	19.95
FF	62097	106400	58.36
BRAM	62.50	140	44.64
DSP	16	220	7.27
IO	20	125	16.00
BUFG	3	32	9.38
MMCM	2	4	50.00

實驗結果



在距離為半公尺的地方，高1.5公尺的地方做量測，可將人臉偵測出，進一步偵測是否有戴口罩，如左圖所示。在解析度為720P時，每秒可跑17、18幀(FPS)。



結論

在此專題中實作了即時偵測人臉與口罩系統在嵌入式系統上，了解軟硬體架構且相互合作，以Xilinx的PYNQ的框架下運用python控制硬體Overlay，硬體負責運算影像的資料，達到加速的效果。在演算法中遇到了蠻多光線上的困難，也許在未來可以在搭載AI的硬體上，發展更成熟的即時系統，希望可以提高正確率。最後我們去限定應用的範圍，去提高正確率，在門口設置量體溫的地方，除了量測體溫之外，可以順便偵測是否有戴上口罩，達到防疫的目的。