

# 1 Results

## 1.1 Hardware Latency

为了对本设计相对软件图像预处理的延迟进行衡量，设置单帧尺寸为200\*200的图像以进行基准延迟测试，软件测试平台为Google Colab免费版，Python版本为 Python 3.11.13 (main, Jun 4 2025, 08:57:29) [GCC 11.4.0] on linux 。采用目前业界流行的OpenCL库对三大滤波模块进行软件重写，经过10000轮重复实验，其对单帧图像的处理延时约为  $0.4584ms - 0.5610ms$ 。

采用同样尺寸的单帧图像作为硬件滤波模块的输入激励以对硬件模块图像预处理的延迟进行度量，测试EDA为Vivado 2023.2，FPGA为Xilinx Artix-7 XC7A100T。由于该工作采用全流水线设计，其流水线延迟为8个时钟周期，因此在处理单帧200\*200图像时，其总的时钟开销为40,008个时钟周期。在目标5纳秒时钟周期（@200MHz）下，单帧处理时间约为 $0.20ms$ ，此时系统的时钟裕度为 $0.105ns$ ；在目标10ns时钟周期（@100MHz）下，单帧处理时间约为 $0.40ms$ ，此时系统的时钟裕度为 $3.734ns$ 。

因此，本设计基于FPGA进行硬件图像预处理，可将传统的软件图像预处理处理单帧图像延迟降低约12.74%（@100MHz，软件端处理延时取最小值）至28.70%（@100MHz，软件端处理延时取最大值）。在极限情况下，可将处理延迟降低64.35%（@200MHz，软件端处理延时取最大值）。

## 1.2 Power Consumption

在@200MHz时钟频率下，除去外围电路、时钟发生电路等模块的功耗后，片上图像预处理总功耗约为 $0.117W$ 。该功耗受时钟频率影响极小。对于200MHz时钟频率，在处理尺寸为200\*200的图像时，其吞吐量约为 $200,000,000/40,000 = 5000 FPS$ ，其每瓦性能指标（Performance-per-watt）约为 $42735FPS/watt$ ，远远高于绝大多数PC端的每瓦性能指标。

## 1.3 Resource Utilization

在Xilinx Artix-7 XC7A100T FPGA上，经过综合、布局、布线后，该三大滤波模块对硬件的资源消耗如下表所示：

Resource	Utilization	Available	Utilization(%)
LUT	754	63400	1.19

Resource	Utilization	Available	Utilization(%)
LUTRAM	228	19000	1.20
FF	387	126800	0.31
IO	33	285	11.58
BUFG	1	32	3.13

由上表可见，该设计非常紧凑，大部分硬件资源消耗都在10%以下，因此该滤波模块可进一步适配小型低成本FPGA，以在实际的交通监控场景中实现较低的部署成本。

## 1.4 Bandwidth Reduction

除去延迟的降低与功耗、成本的优势以外，该设计还具备降低数据传输带宽的特性。该设计将摄像头输入的RGB888信号压缩为单通道的二值黑白信号，在标准的30FPS监控场景下，假设输入视频尺寸为200\*200，不采用图像预处理时所需传输带宽约为 $Bandwidth@30FPS = 200 \cdot 200 \cdot 24bits/pixel \cdot 30FPS = 28.8Mbps$ ，而在经过FPGA预处理后，传输带宽降低至 $Bandwidth@30FPS = 200 \cdot 200 \cdot 1bit/pixel \cdot 30FPS = 1.2Mbps$ ，约为原始数据的 $\frac{1}{24}$ 。这直接提升了系统的可扩展性，使多路摄像头的视频流能够通过单一网络链路聚合传输而不会造成带宽饱和。

## 2 Conclusion

本设计通过将传统的由软件端执行的图像预处理模块固化至FPGA硬件端，实现了延迟、功耗、传输带宽的显著降低，具有将多路视频流通过单一网络链路聚合传输的可拓展性，并保证了较低的延迟与功耗，可大大降低PC端人工智能推理负载，具备实际场景下的可行性。