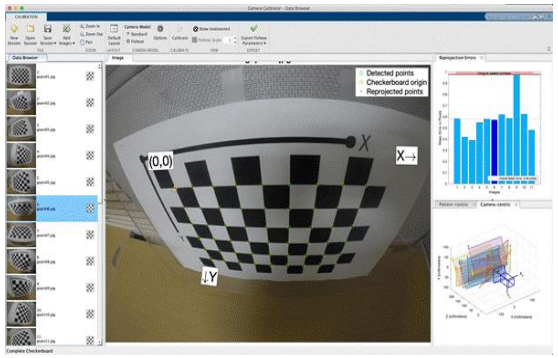
MATLAB EXPO

将图像处理算法和AI模型快速部署到CPU和FPGA

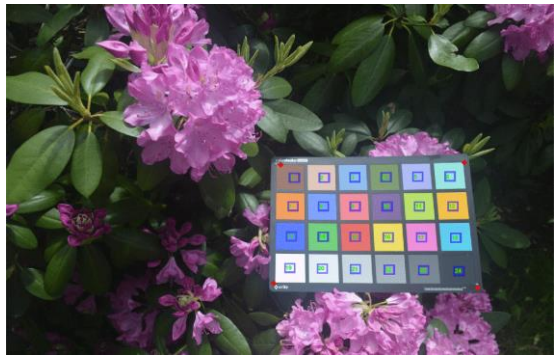
严小商 MathWorks

MathWorks

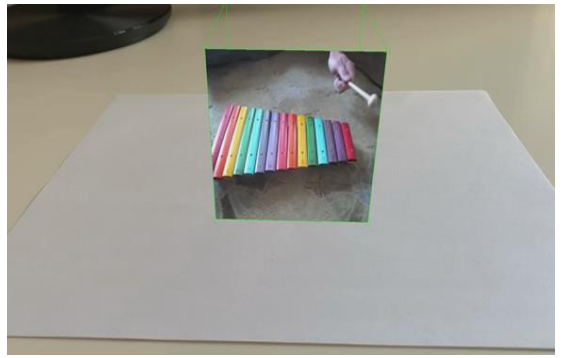
MATLAB图像处理和计算机视觉



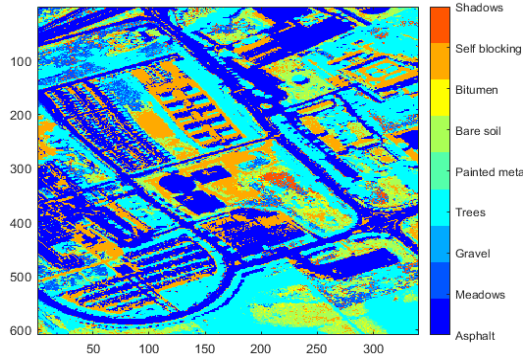
相机标定



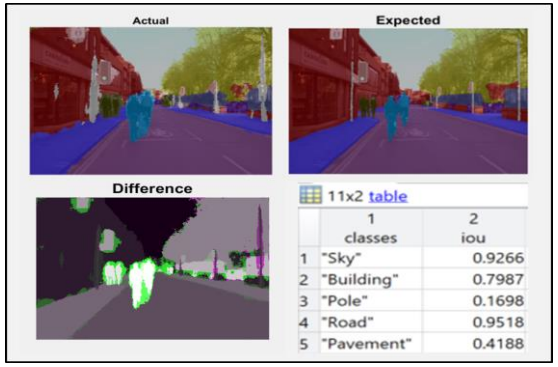
相机图像处理流程



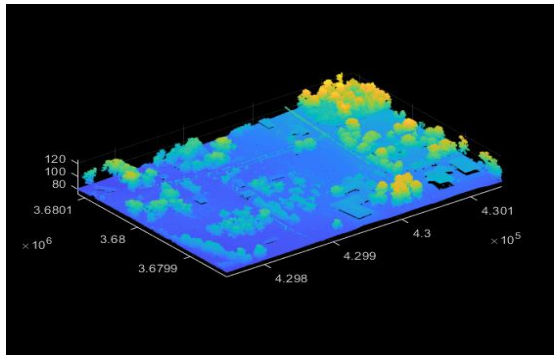
AR / VR



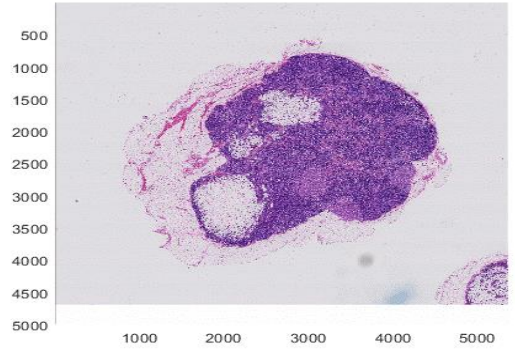
高光谱图像



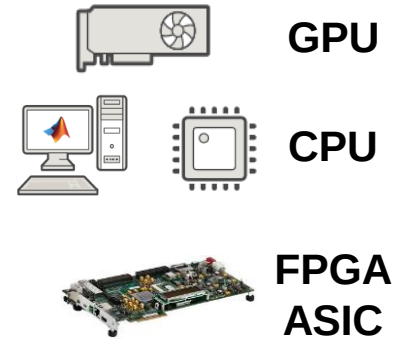
深度学习



激光雷达图像处理



超大幅图像

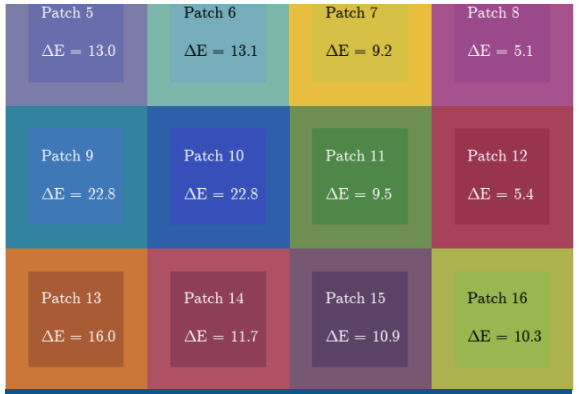


代码生成

MATLAB图像处理和计算机视觉

相机图像处理

- RAW 到 RGB 转换
- 图像增强
- 色彩空间变换
- HDR图像处理
- 从低分辨率图像创建高分辨率图像



图像质量评估

- 测试图
- 色彩精度
- 图像质量指标

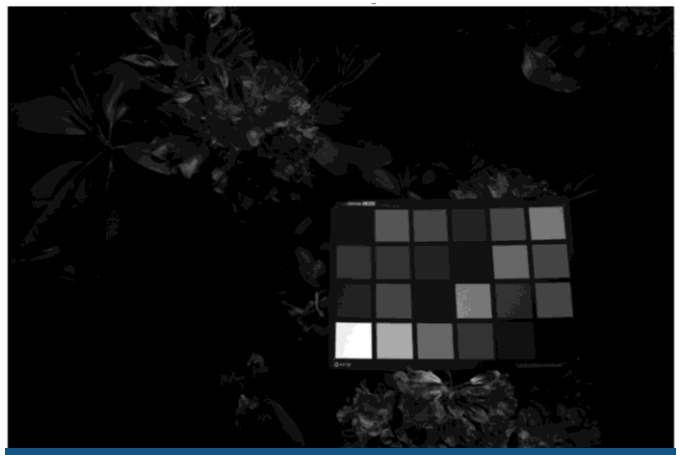
深度学习

- RAW 到 RGB 转换
- RAW低光照度图像恢复
- 去噪

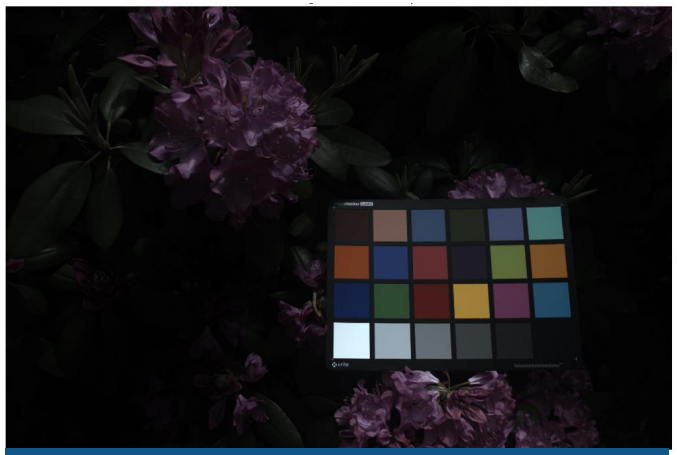
算法部署

- 算法到C/C++代码
- 算法到HDL代码
- 部署到CPU/FPGA

例子：RAW 转 RGB 相机工作流



线性CFA图



线性色彩空间渲染图

sRGB色彩空间渲染图

Functions	
rawinfo	Read information about color filter array (CFA) images in RAW files
rawread	Read CFA images from RAW files
demosaic	Convert Bayer pattern encoded image to truecolor image
lin2rgb	Apply gamma correction to convert linear sRGB to sRGB color space
raw2planar	Separate a Bayer-patterned CFA image into individual, sensor-element
planar2raw	Combine planar sensor images into a full Bayer-pattern CFA image
raw2rgb	**Convert a RAW file into an RGB file in one step**

Example: End-to-End Implementation of Digital Camera Processing Pipeline

白平衡

自动白平衡算法的两个步骤:

- 步骤 1: 光源估计
- 步骤 2: 色彩平衡校正

Original Image Before and After Gamma Correction

Montage of Best White-Balanced Images: White Point, Gray World, Cheng

Algorithms to estimate scene illuminant	
illumwhite	Estimate illuminant using White Patch Retinex algorithm
illumgray	Estimate illuminant using gray world algorithm
illumpca	Estimate illuminant using principal component analysis (PCA)

Example: Comparison of Auto White Balance Algorithms

低光照度图像增强

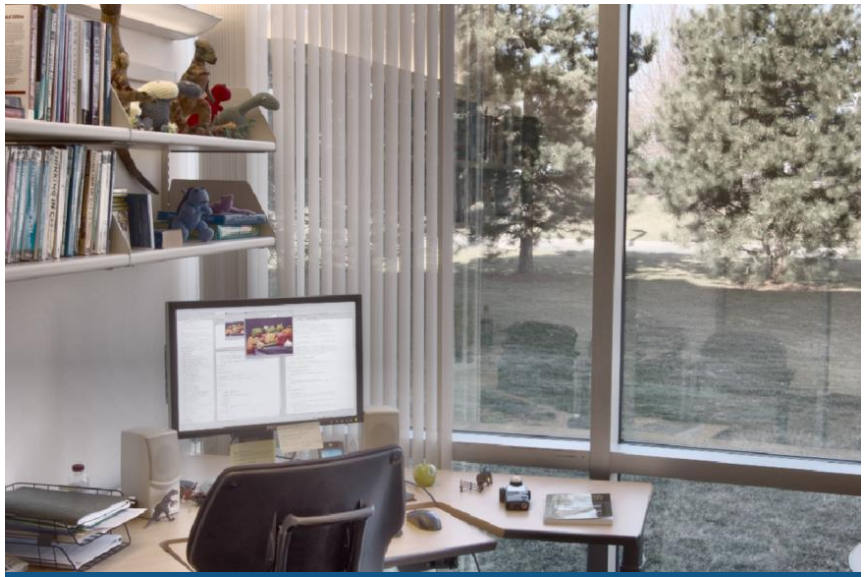
Function	
imlocalbrighten	Brighten low-light images

Example: Low-Light Image Enhancement

色彩空间转换

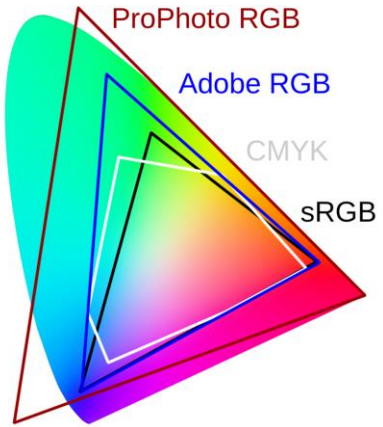
支持多色彩域

- BT.2020 (Ultra High Definition, UHD)
- BT.2100 (High Dynamic Range, HDR)
- ProPhoto (ROMM RGB) Color Space

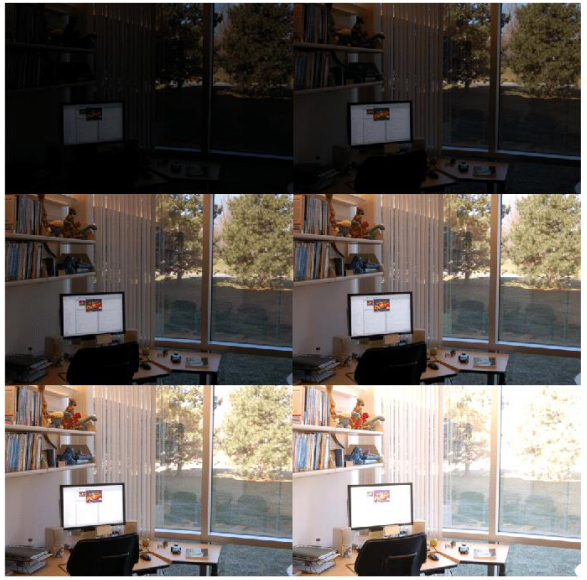


HDR

Function	
rgbwide2ycbcr	Convert wide-gamut RGB color values to YCbCr color values
ycbcr2rgbwide	Convert YCbCr color values to wide-gamut RGB color values
xyz2rgbwide	Convert CIE 1931 XYZ color values to wide-gamut RGB color values
rgbwide2xyz	Convert wide-gamut RGB color values to CIE 1931 XYZ color values

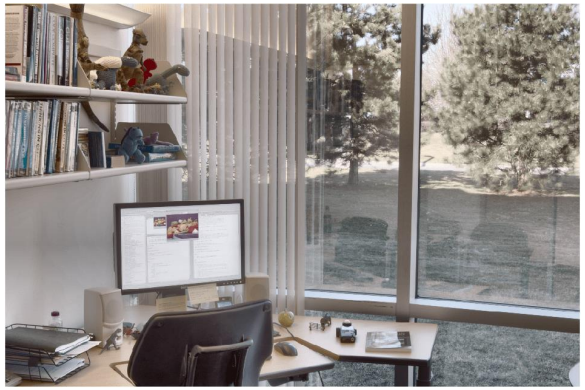


高动态范围(HDR)图像



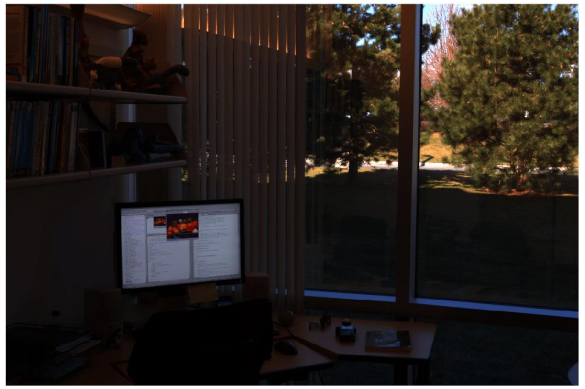
makehdr

从多个低动态范围(LDR)图像创建高动态范围(HDR)图像



tonemap

HDR 转 LDR图像



支持 EXR 和 HDR 文件格式

从低分辨率图像创建高分辨率图像

Set of Low-Resolution Burst Mode Images

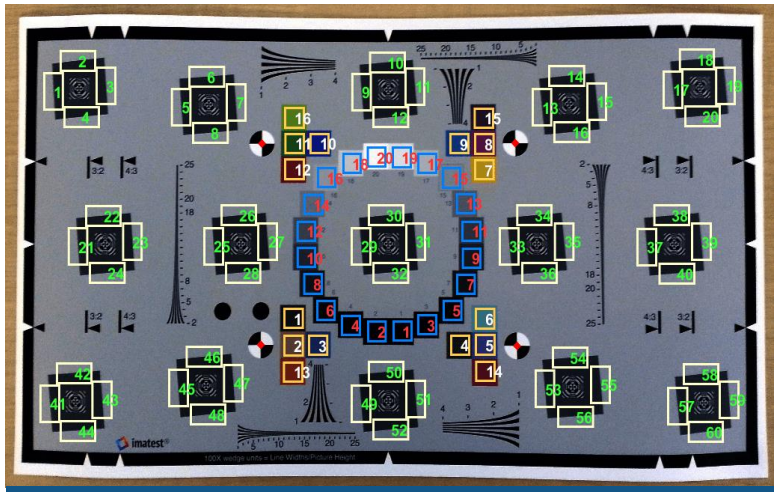
burstinterpolant

High-Resolution Image

图像质量：支持测试图

- Imatest：eSFR测试图(ISO 12233)的扩展、增强、楔形扩展和楔形增强版本的检测和分析
- X-Rite® (Gretag Macbeth®) ColorChecker ® 测试图

Function	
measureIlluminant	Measure scene illuminant of test chart
colorChecker	Identifies the color patch ROI in test chart
displayChart	Display test chart with ROI
measureColor	Measure colors in test chart
displayColorPatch	Display measured and reference color as color patches
plotChromaticity	



eSFR 测试图

X-Rite ColorChecker

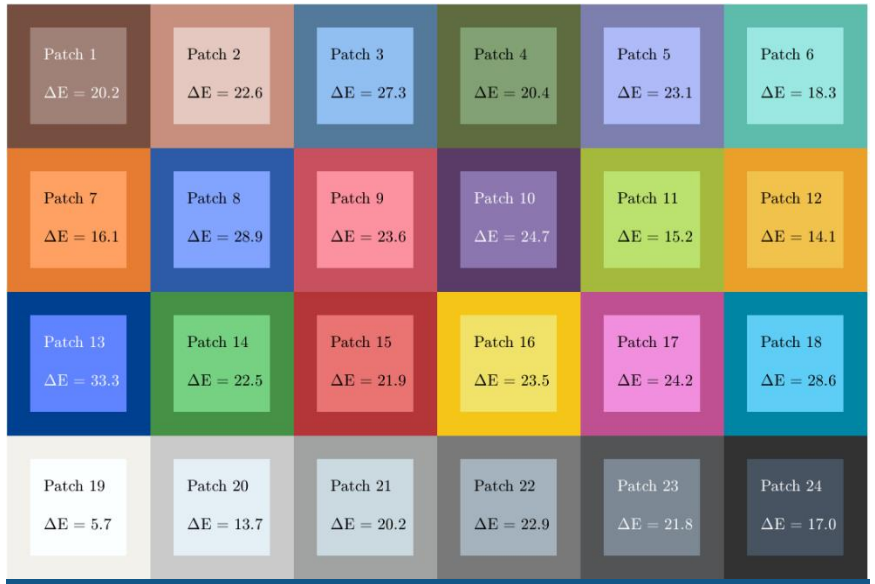
Example: Evaluate Quality Metrics on eSFR Test Chart

Example: colorChecker

图像质量：比较测量

- 测试图的色彩梯度测量
- RGB / L*a*b 色彩空间里色彩差异比较

Function	
deltaE	Color difference based on CIE76 standard
imcolordiff	Color difference based on CIE94/CIE2000 standard



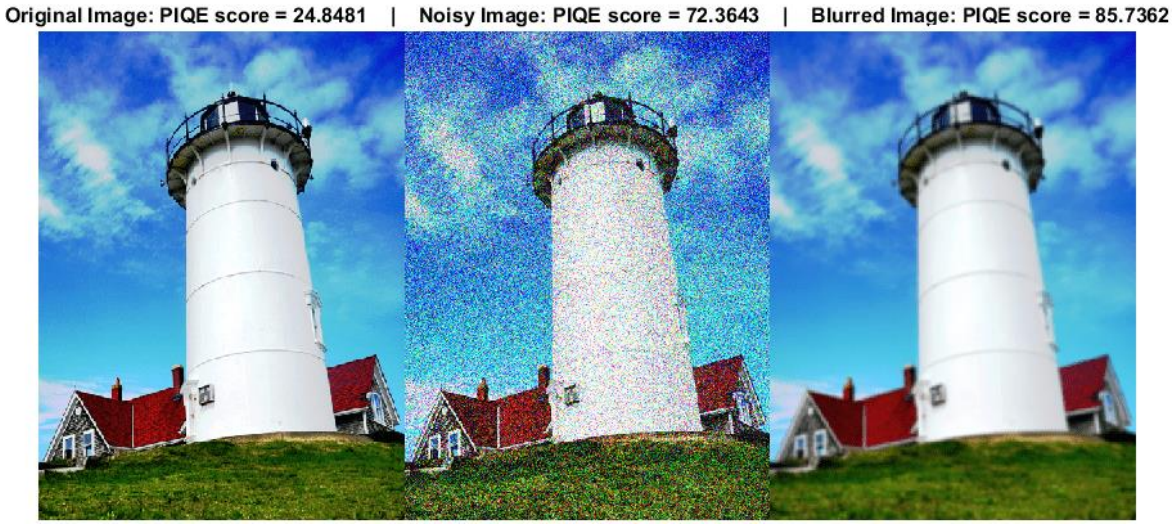
Measured vs Reference Color Difference


Original vs Local Color Difference

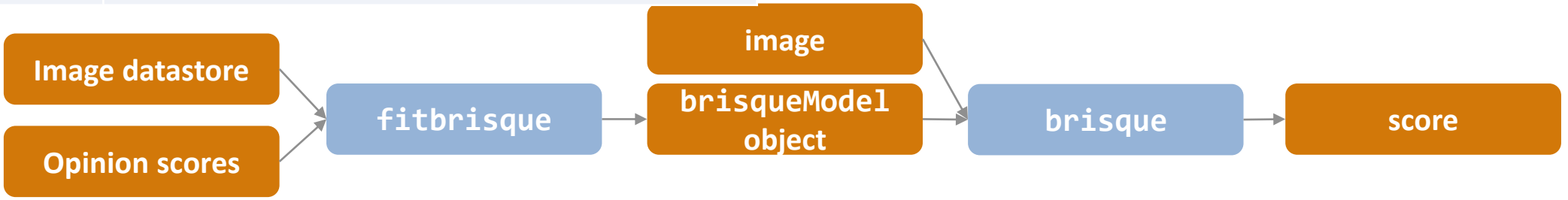
图像质量指标

Reference-Based Techniques Function	
immse	Mean squared error
psnr	Peak signal-to-noise
ssim	Structural similarity metric
multissim	MS-SSIM index for image quality
multissim3	MS-SSIM index for volume quality

No-Reference Techniques Function	
niqe	Naturalness Image Quality Evaluator
brisque	Blind Reference-less Image Spatial Quality Evaluator
piqe	Perception-based Image Quality Evaluator
nima	Analyze the aesthetic quality of images



PIQE No-Reference Techniques



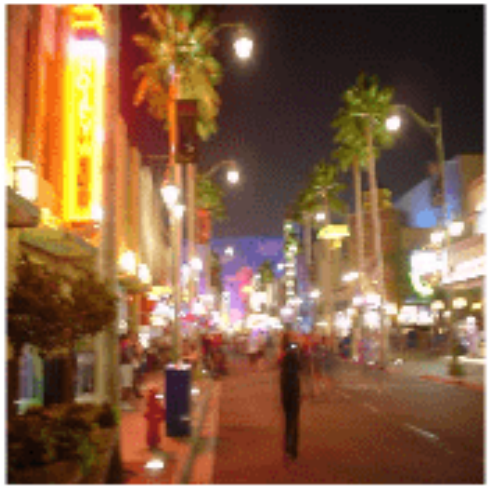
Example: Train and Use a No-Reference Quality Assessment Model

用NIMA卷积神经网络评估图像质量

Image 780.JPG
Mean Score: 92.432
Std Dev: 12.038

Image 278.bmp
Mean Score: 28.1042
Std Dev: 16.6824

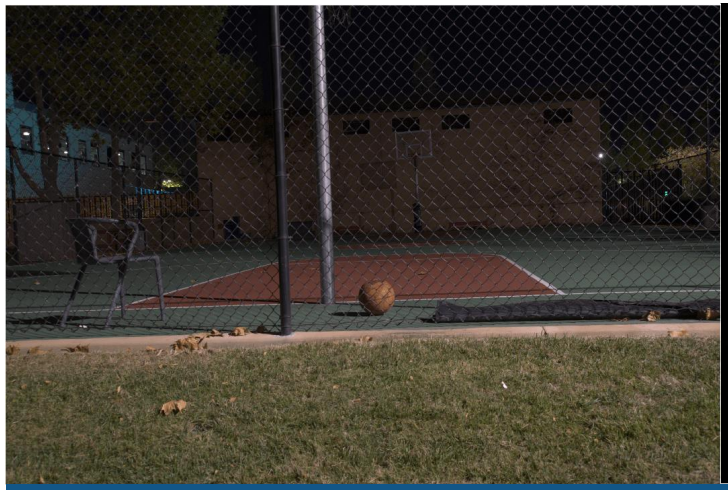
Image 772.JPG
Mean Score: 12.7964
Std Dev: 13.7164



Example:Evaluate Image Quality Using Trained NIMA Model

用深度学习恢复低光照度图像

- 用深度学习网络恢复低光照度下的短曝光RAW图像
- 预训练的U-Net网络用于低光照度图像恢复



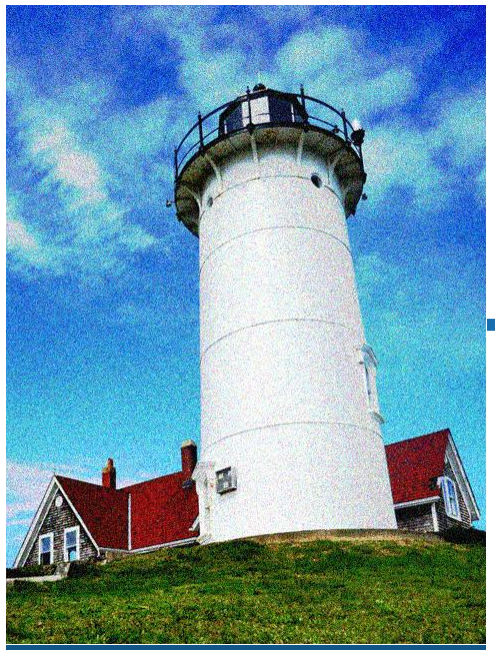
低光照长曝光

低光照图短曝光

用深度学习网络恢复的低光照度短曝光图像

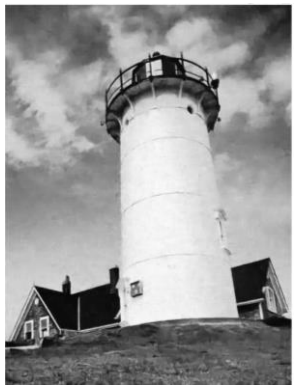
Example: Recover Images from Extreme Low-Light Conditions Using Deep Learning

用预训练的卷积神经网络降噪

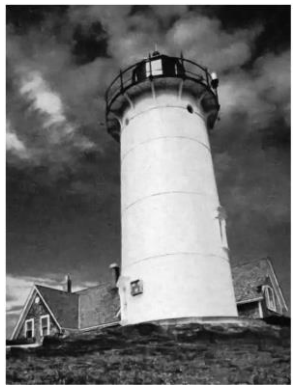


有噪图像

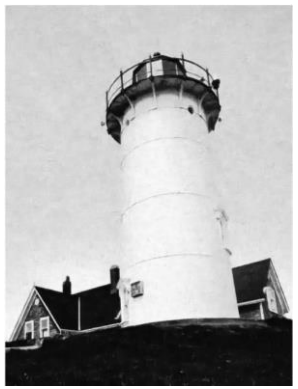
Pretrained
DcCNN



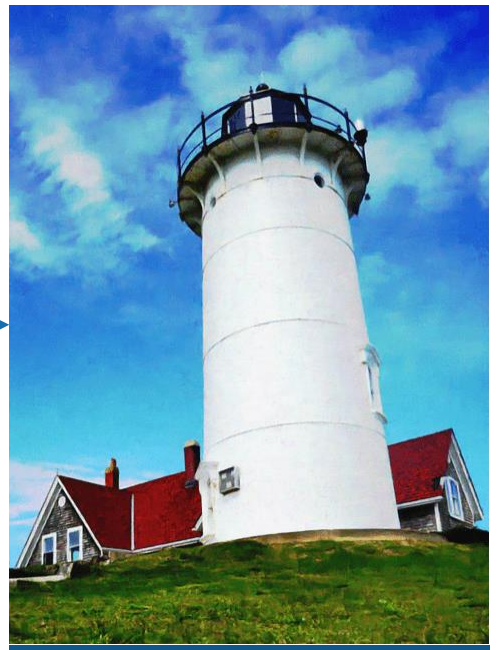
Red



Green



Blue

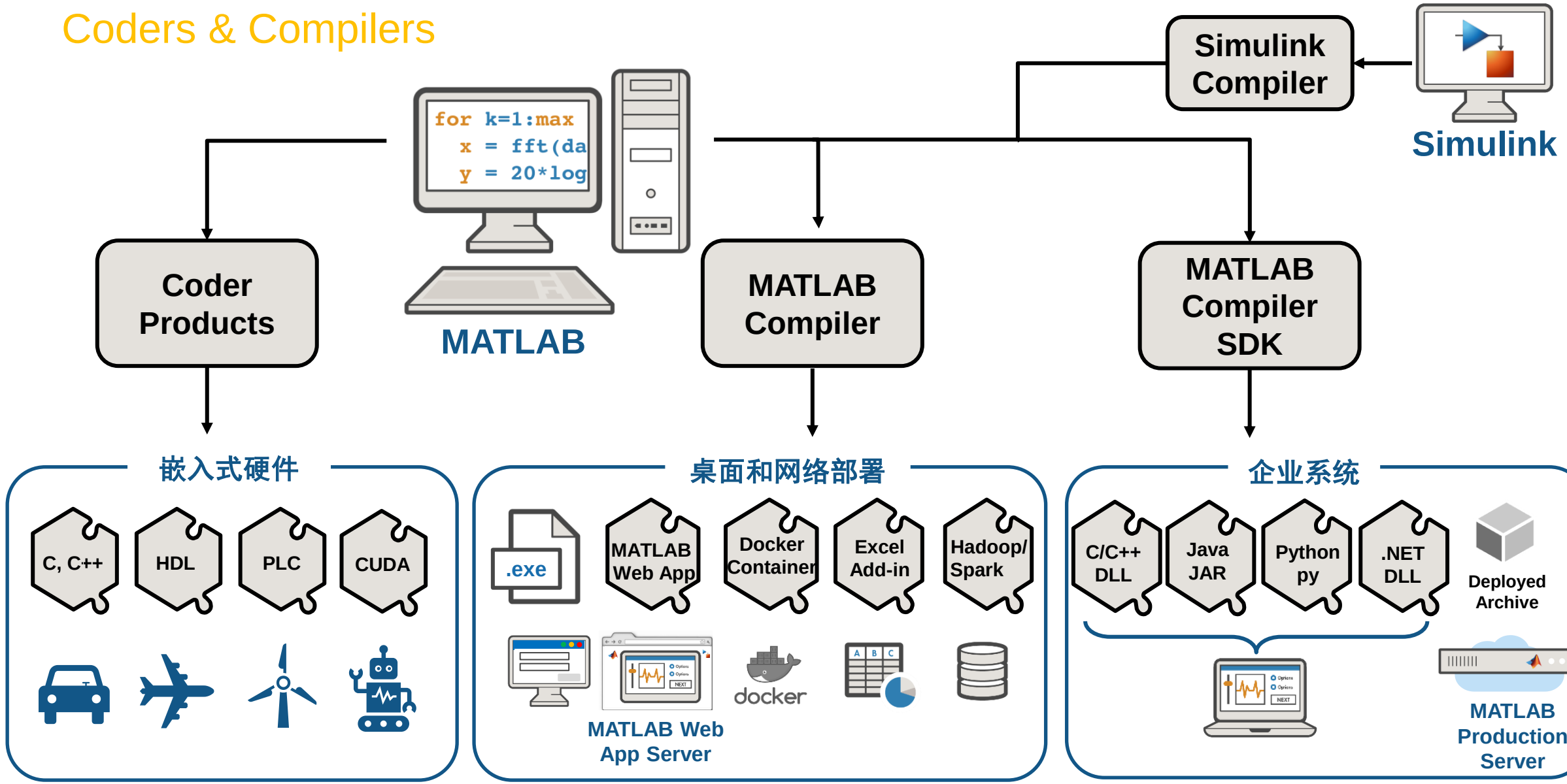


降噪后图像

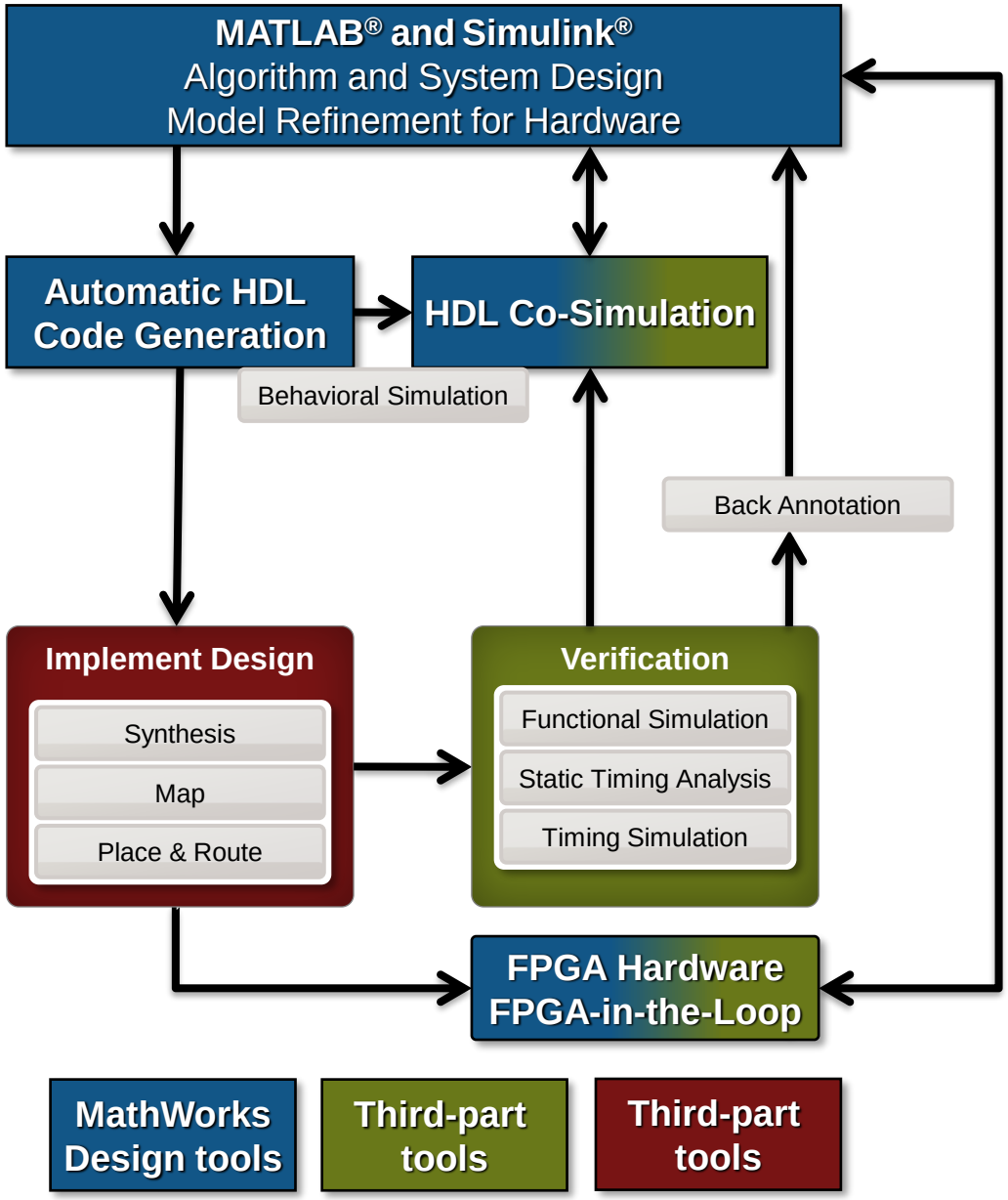
参考图像

[Remove Noise from Color Image Using Pretrained Neural Network]

代码生成和部署
Coders & Compilers

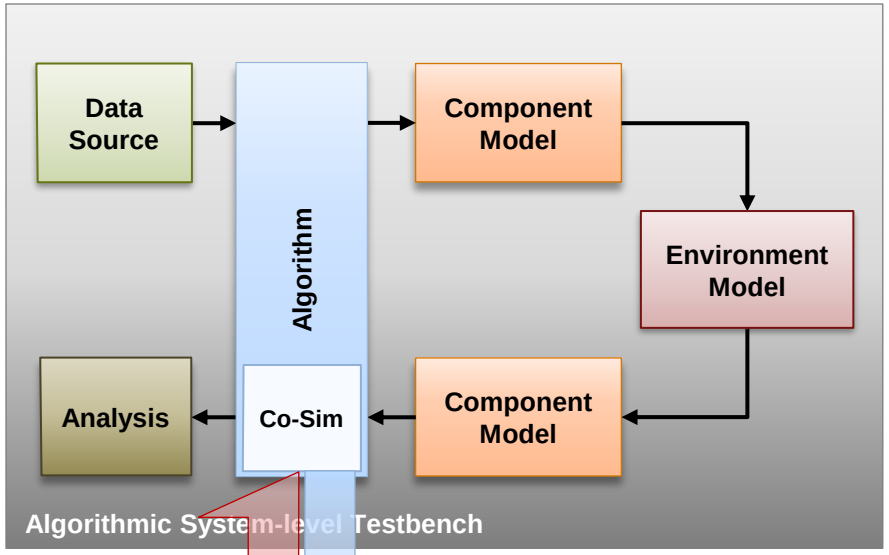


基于模型的FPGA设计流程

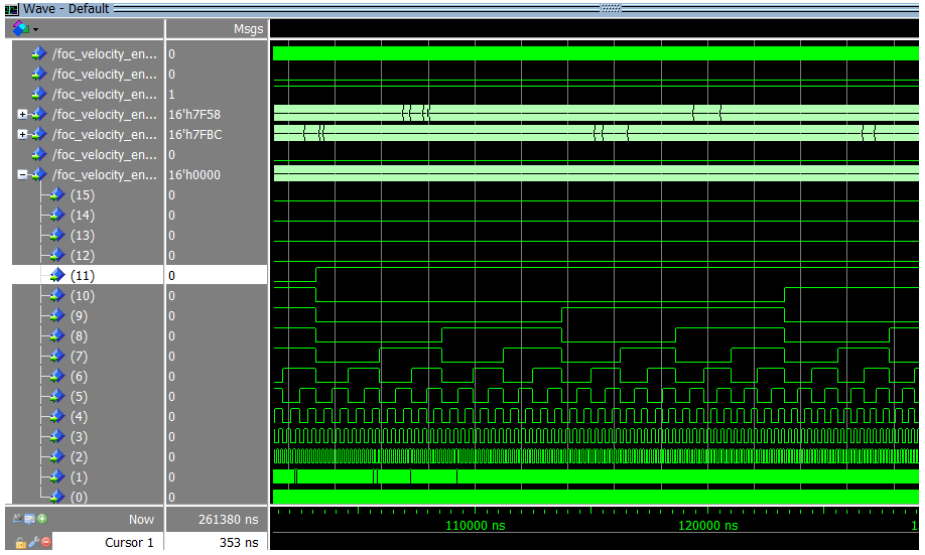


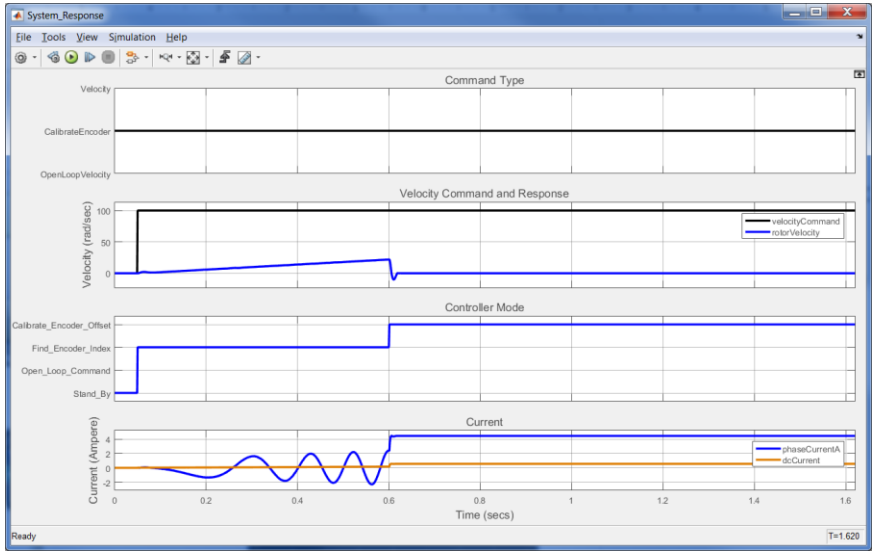
- 自动代码生成
  - 自动定点化
  - 与目标器件无关的 HDL 代码
    - VHDL-1993 (IEEE® 1076-1993) or later
    - Verilog-2001 (IEEE 1364-2001) or later
- 验证
  - 自动生成HDL test-bench
  - 与HDL 仿真器联合仿真
- 设计自动化
  - 集成Xilinx、Altera综合工具接口
  - 面积、速度优化
  - Xilinx、Altera器件编程下载

Co-simulation 用于 HDL 代码验证



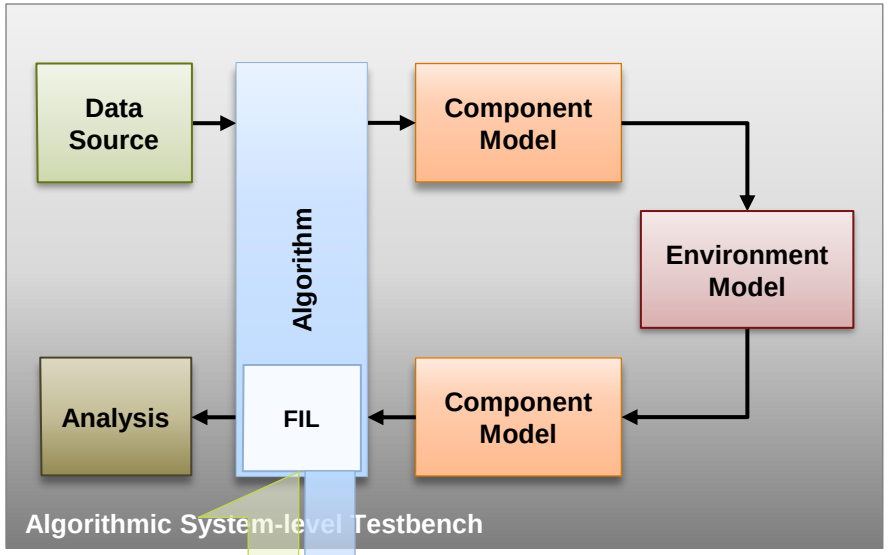
- 第三方HDL仿真器联合仿真:
  - 复用ML/SL testbench
  - HDL 代码在3rd-party HDL仿真器里运行
  - 可集成手写代码/生成代码
  - 在MATLAB 和 HDL仿真器里分析



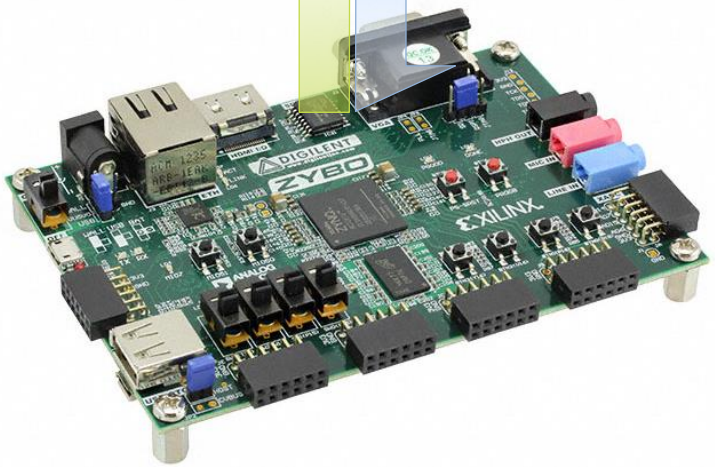


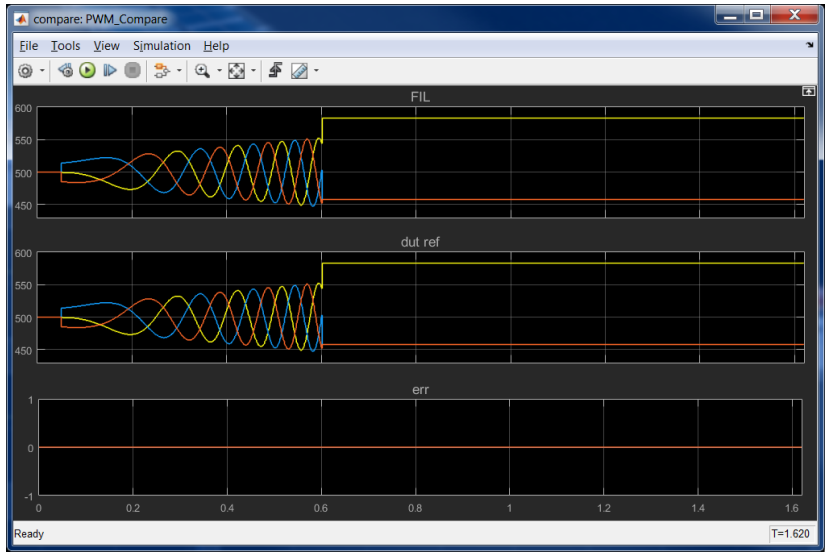
* Mentor Graphics® ModelSim® or Questa ®
Cadence ® Incisive ®

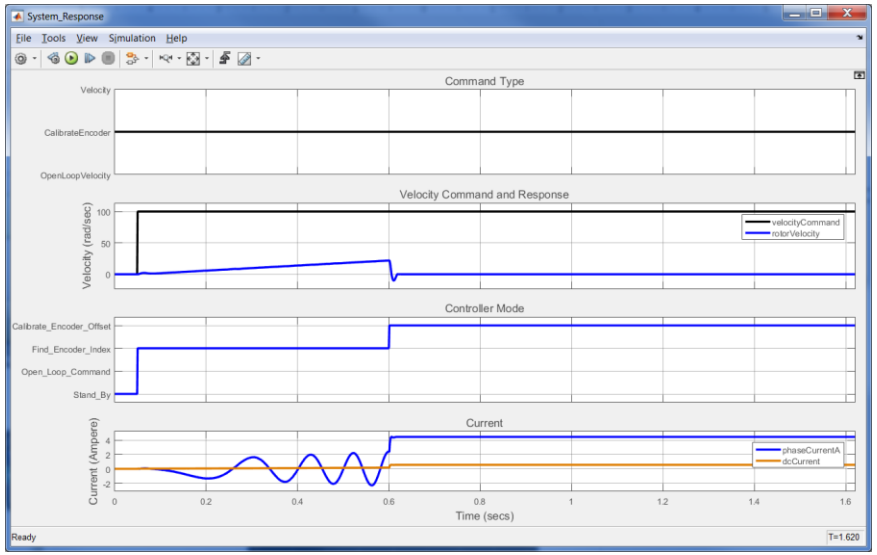
FPGA-in-the-Loop 验证



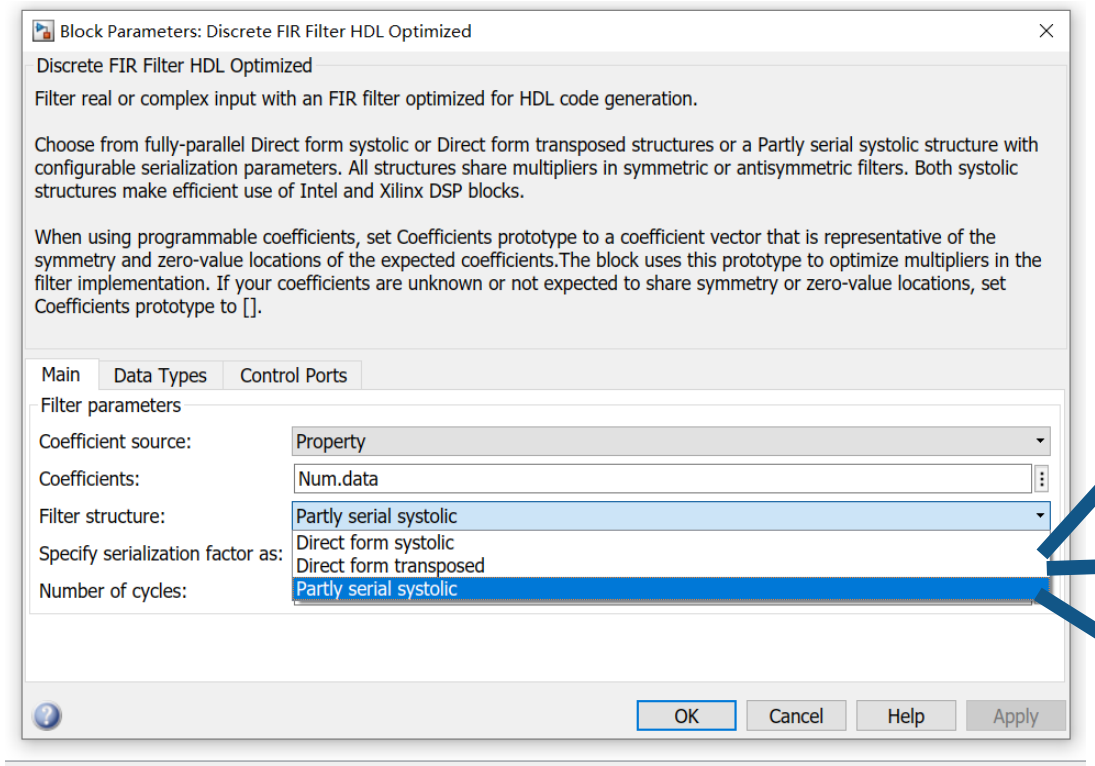
- 用开发板进行FIL验证:
  - 复用ML/SL testbench
  - HDL 代码在FPGA里运行
  - 可集成手写代码/生成代码
  - 自动插入协调仿真组件 (Ethernet, JTAG, PCIe)

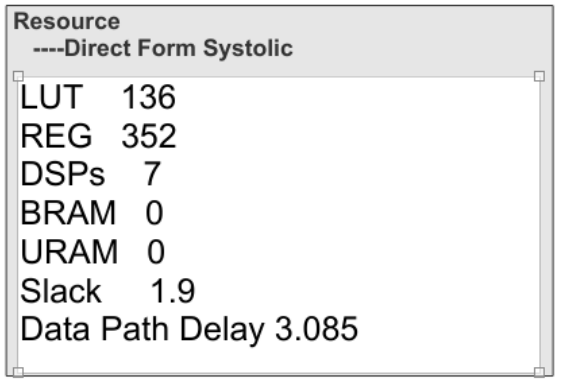


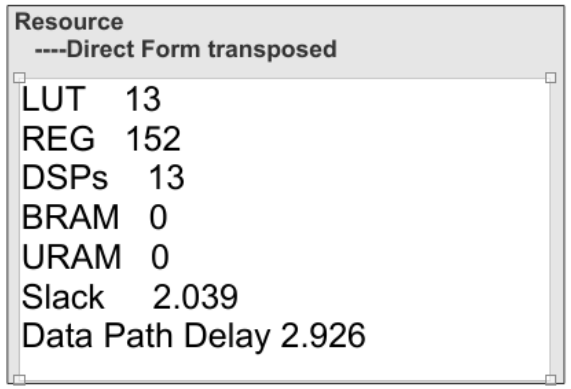


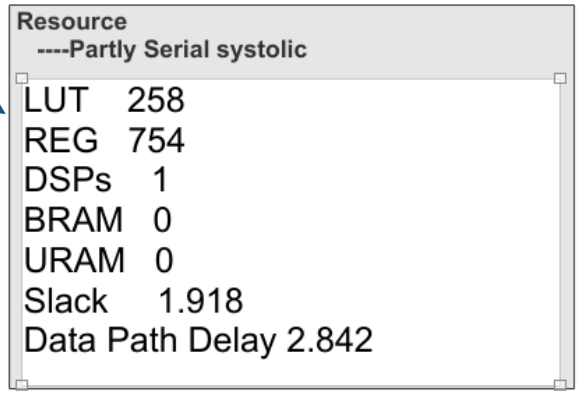


资源优化：结构选择

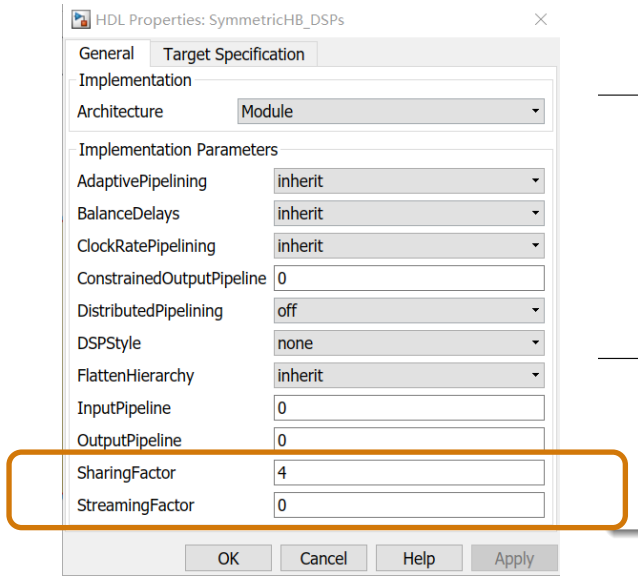

Block Parameters: Discrete FIR Filter HDL Optimized
Discrete FIR Filter HDL Optimized
Filter real or complex input with an FIR filter optimized for HDL code generation.
Choose from fully-parallel Direct form systolic or Direct form transposed structures or a Partly serial systolic structure with configurable serialization parameters. All structures share multipliers in symmetric or antisymmetric filters. Both systolic structures make efficient use of Intel and Xilinx DSP blocks.
When using programmable coefficients, set Coefficients prototype to a coefficient vector that is representative of the symmetry and zero-value locations of the expected coefficients.The block uses this prototype to optimize multipliers in the filter implementation. If your coefficients are unknown or not expected to share symmetry or zero-value locations, set Coefficients prototype to [].
Main
Data Types
Control Ports
Filter parameters
Coefficient source:
Property
Coefficients:
Num.data
Filter structure:
Partly serial systolic
Specify serialization factor as:
Number of cycles:
Direct form systolic
Direct form transposed
Partly serial systolic
OK
Cancel
Help
Apply

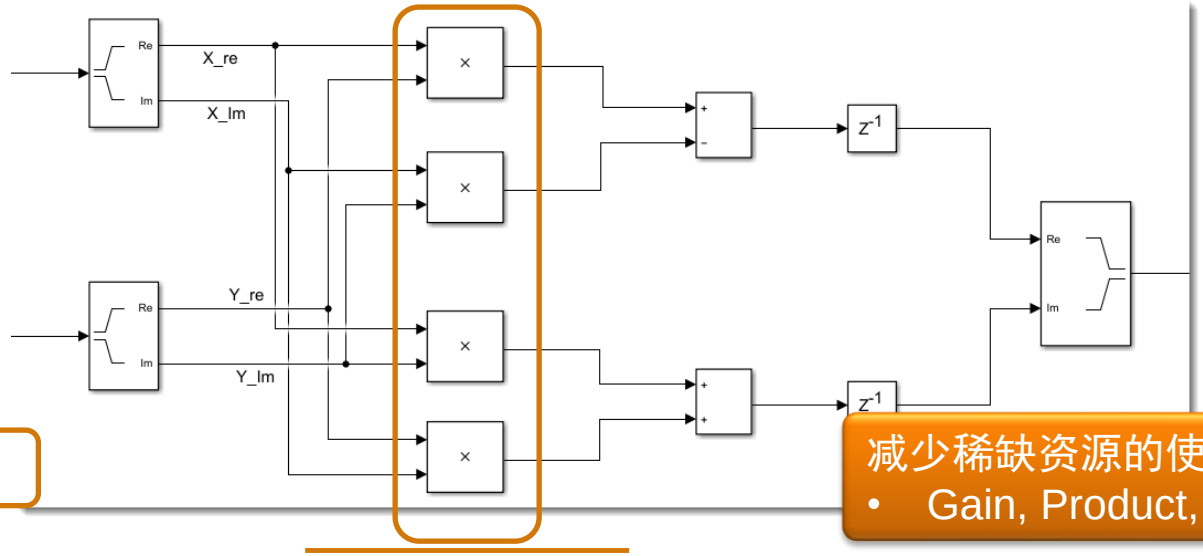

Resource
----Direct Form Systolic
LUT 136
REG 352
DSPs 7
BRAM 0
URAM 0
Slack 1.9
Data Path Delay 3.085


Resource
----Direct Form transposed
LUT 13
REG 152
DSPs 13
BRAM 0
URAM 0
Slack 2.039
Data Path Delay 2.926


Resource
----Partly Serial systolic
LUT 258
REG 754
DSPs 1
BRAM 0
URAM 0
Slack 1.918
Data Path Delay 2.842

资源优化：复用



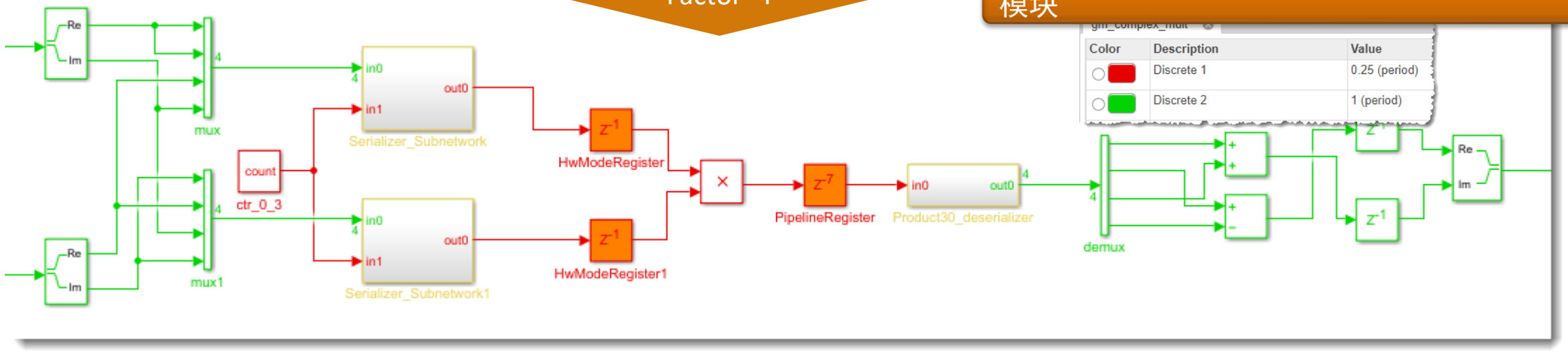


减少稀缺资源的使用，如乘法器

- Gain, Product, Multiply-Add

Resource Sharing Factor=4

提升复用块采样率，自动插入分时复用转换模块



时序优化：时钟流水线和自适应流水线

Tool and Device

Synthesis Tool: Xilinx Vivado

Family: Virtex7

Package: ffg1761

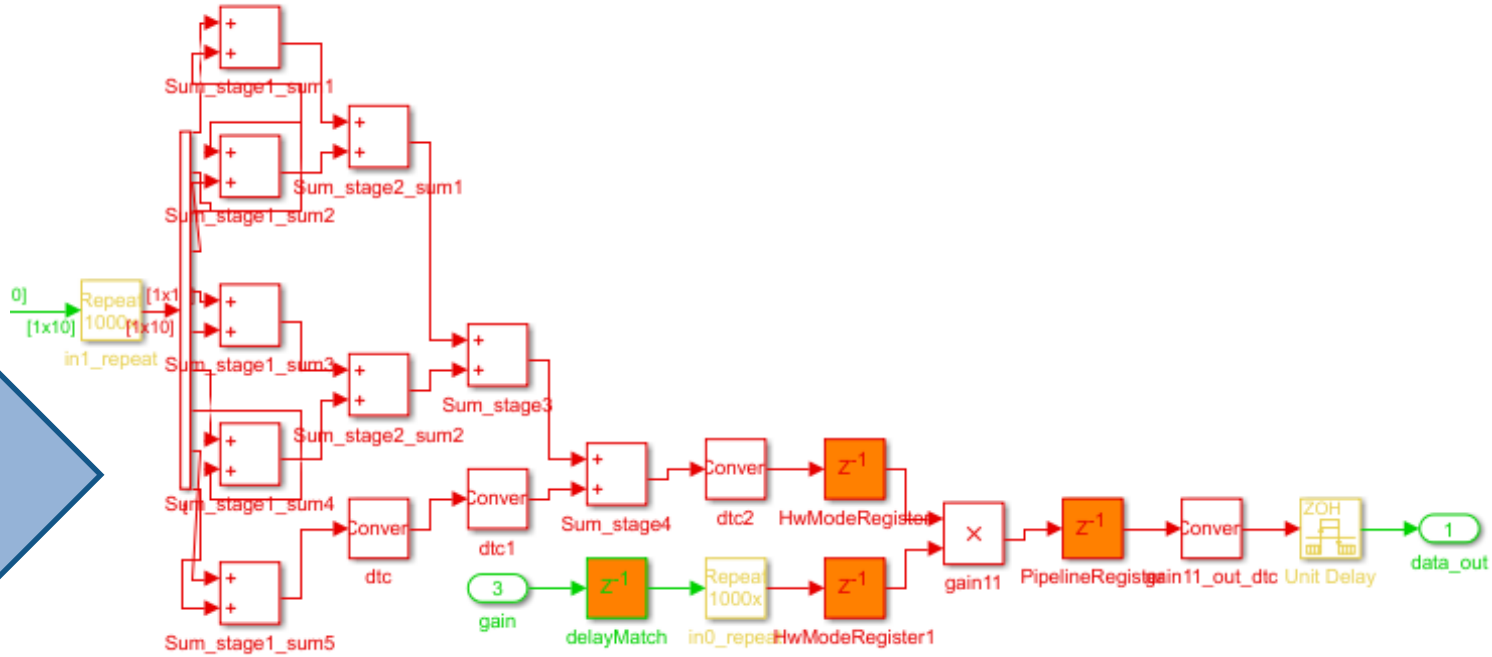
Objectives Settings

Target Frequency (MHz) 100

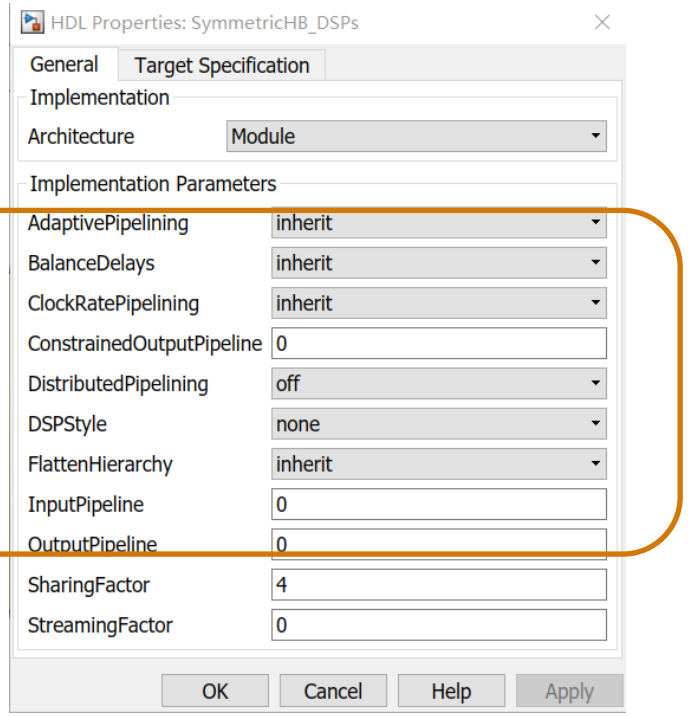
Optimizations

General | Pipelining | Resource sharing

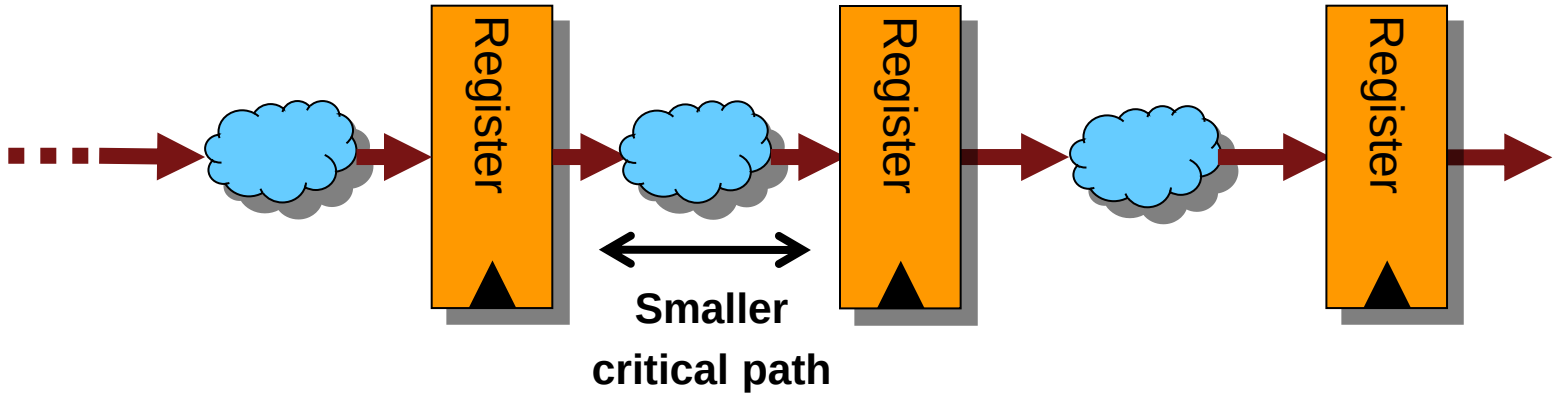
- [ ] Hierarchical distributed pipelining
- [x] Clock-rate pipelining
- [ ] Allow clock-rate pipelining of DUT output ports
- [ ] Preserve design delays
- [x] Adaptive pipelining



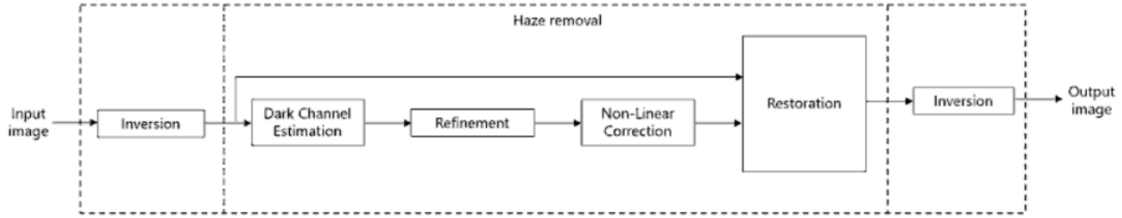
时序优化：自动/半自动流水线

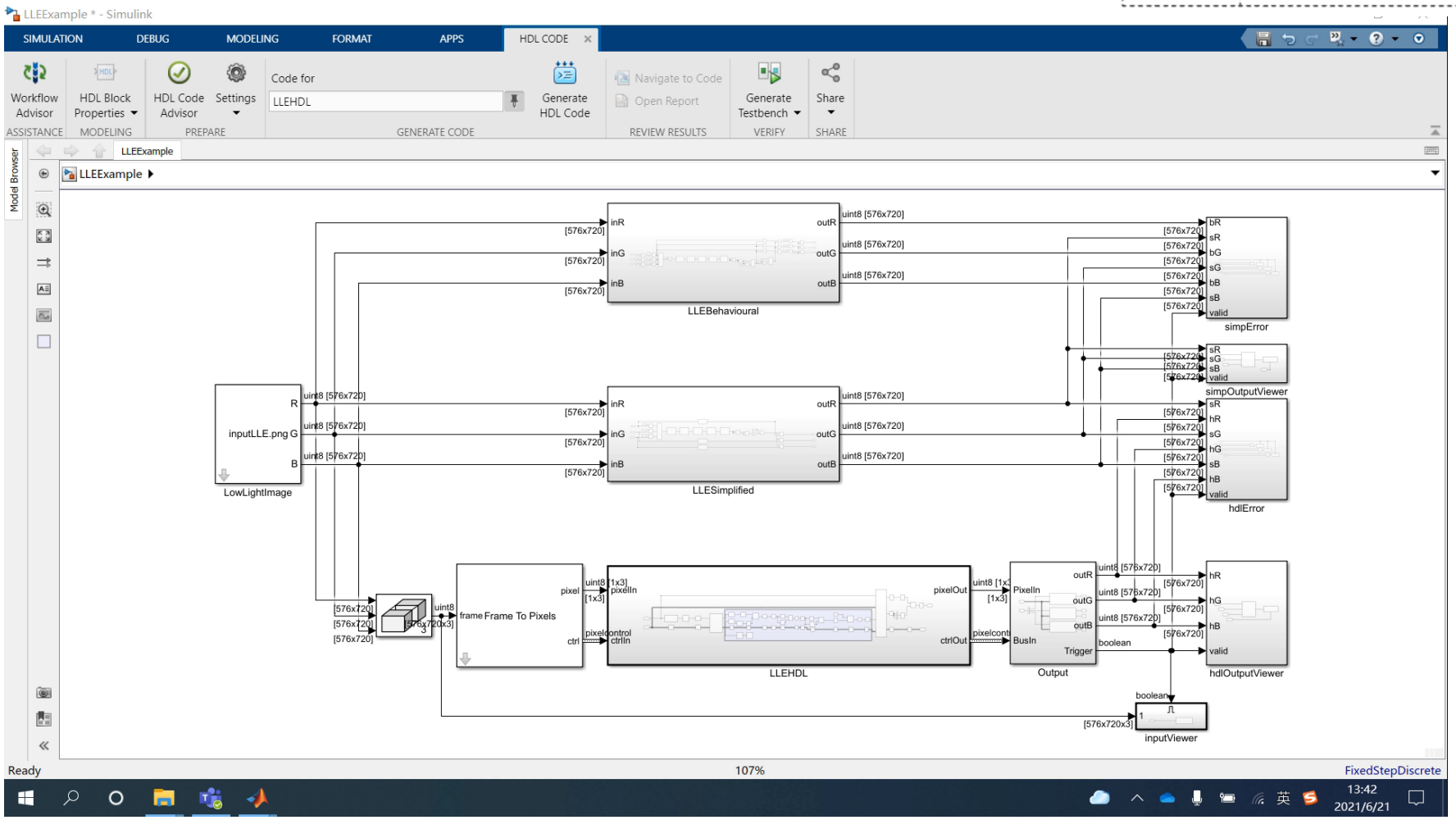


- ✓ 自动/半自动流水线
- ✓ 满足时序目标



HDL代码生成：LLE

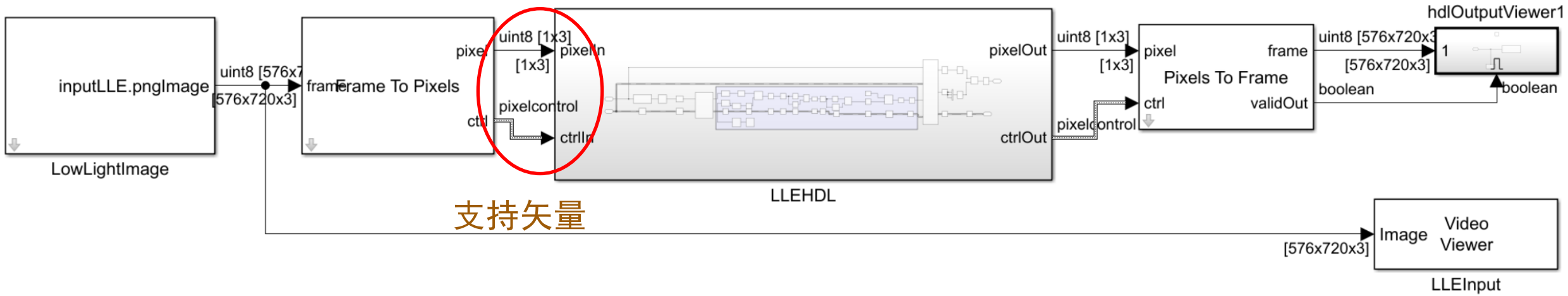




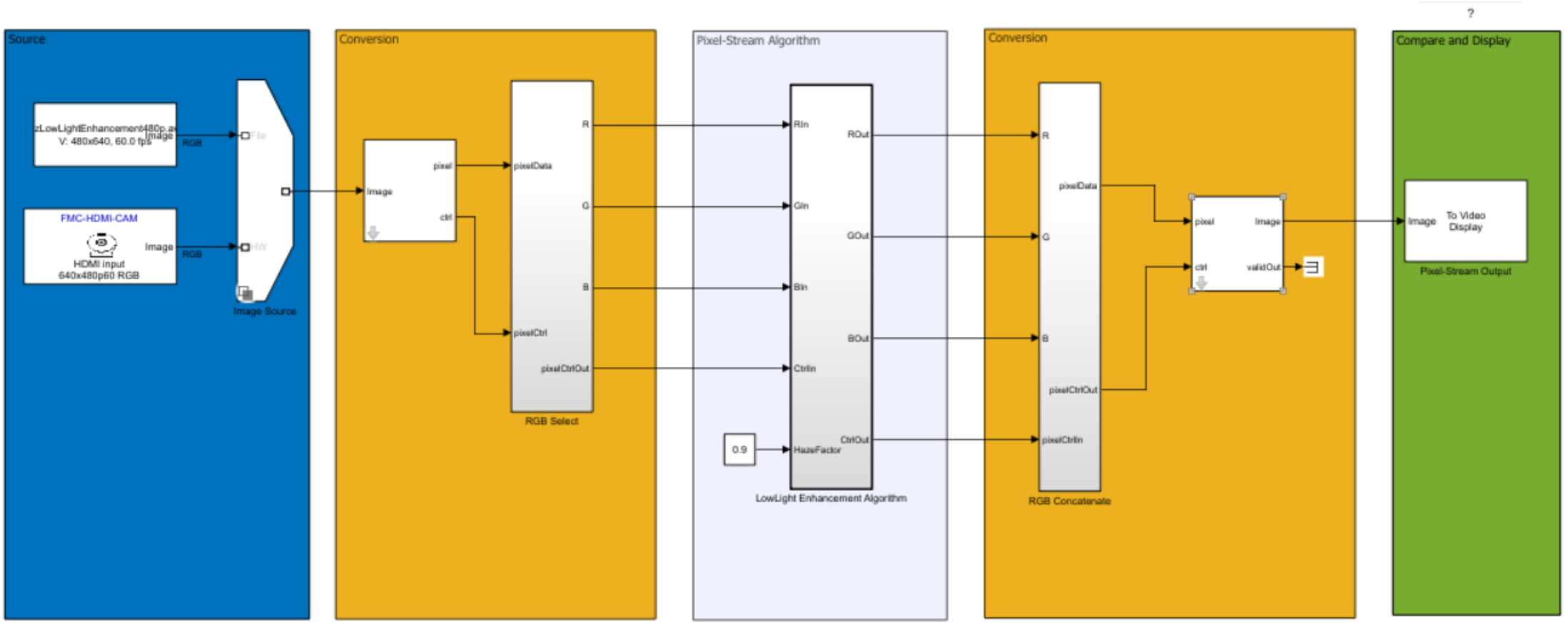
[Example: Low Light Enhancement]

HDL代码生成：LLE

读取图像 | 帧 – 流转换 | HDL系统 | 流 – 帧转换 | 显示

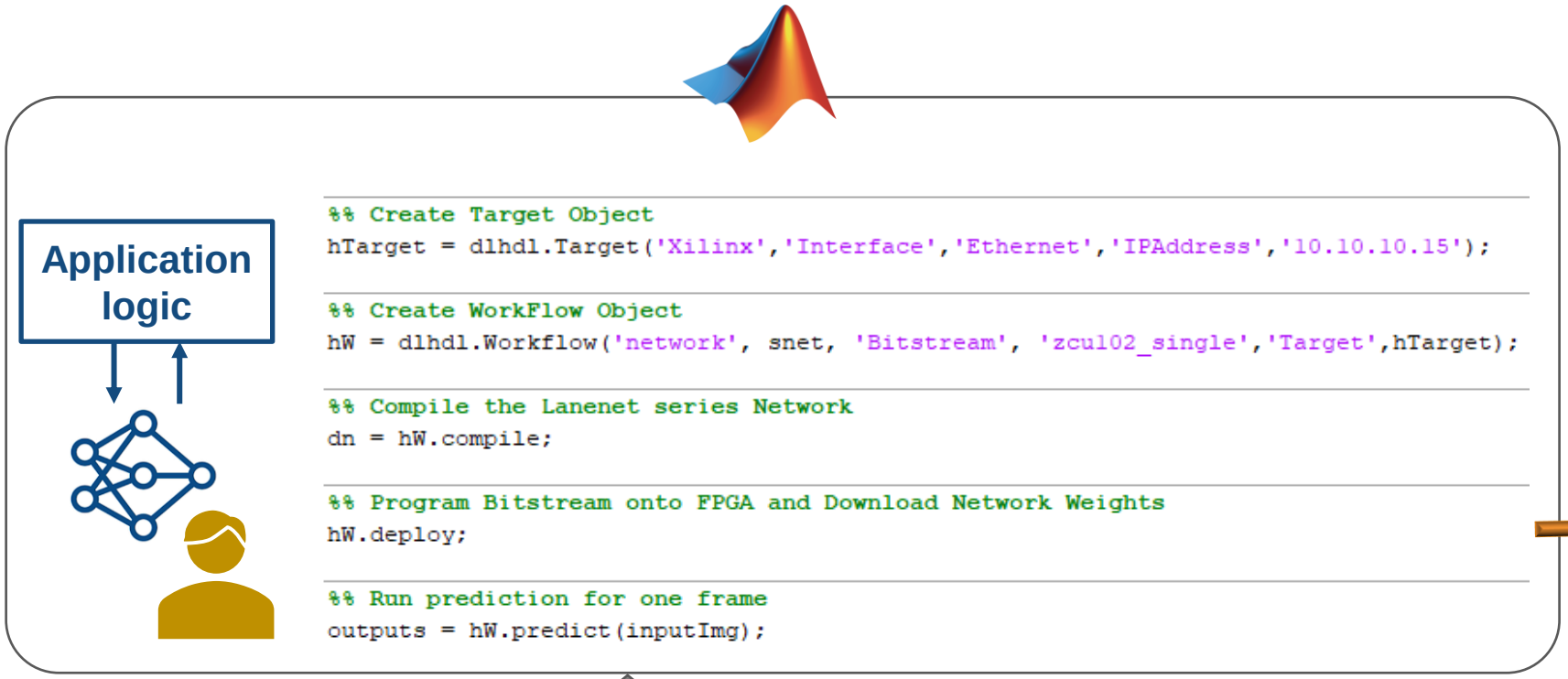


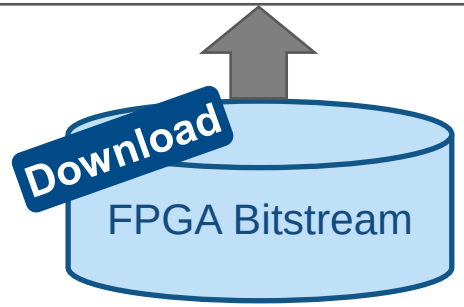
HDL代码生成：LLE 与 FPGA在环验证



Example: Low Light Enhancement With Zynq-Based Hardware

从深度学习网络到HDL/C

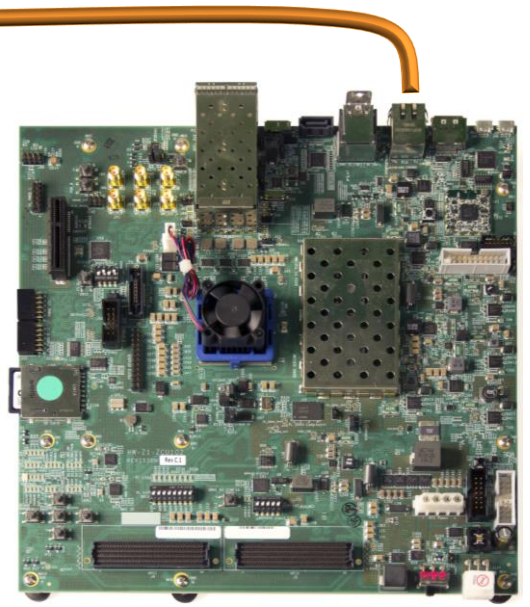




Hardware support package
Deep learning processor with I/O and external memory interfaces

Layer control instructions

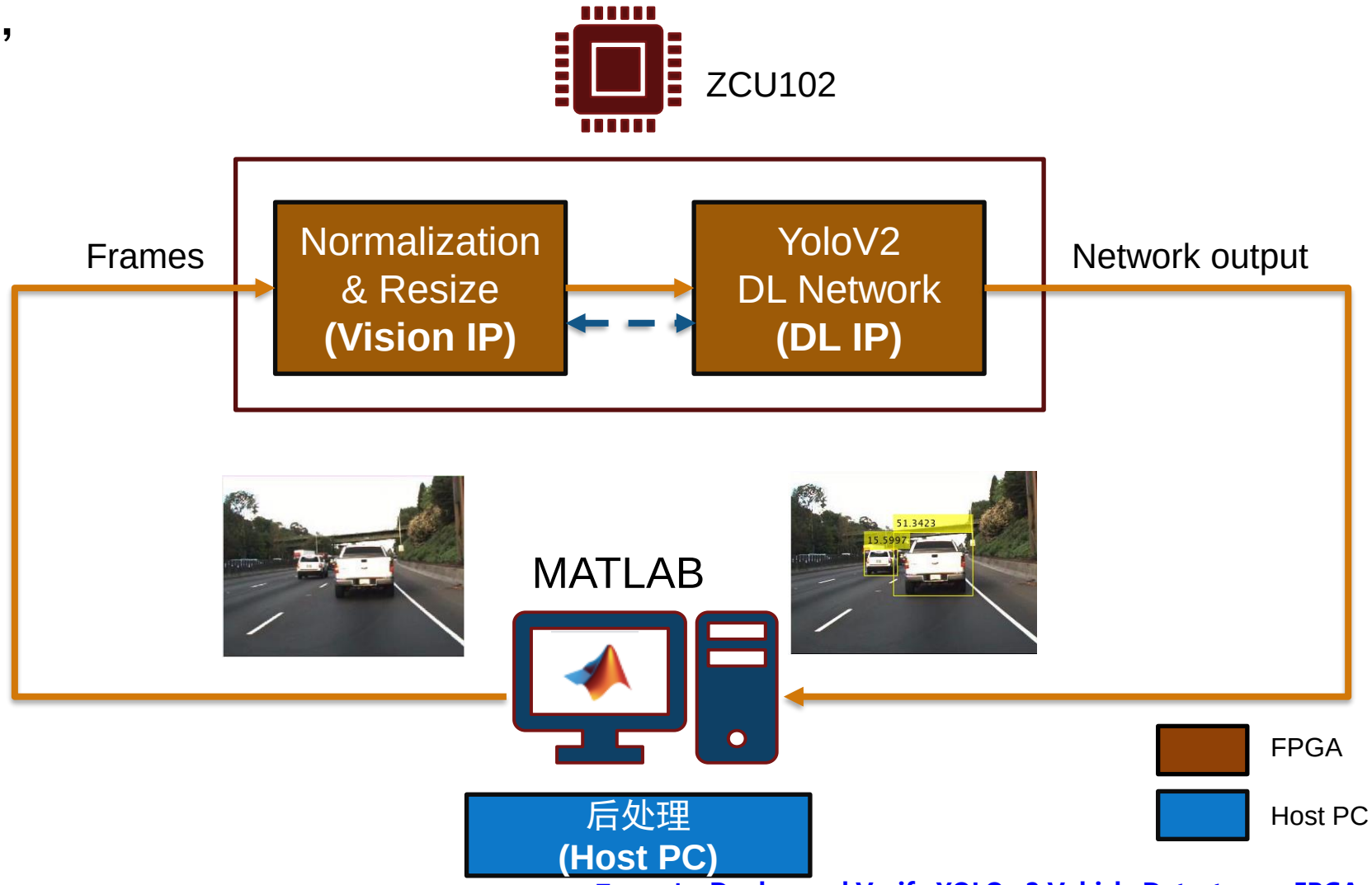
Weights & Activations



支持的网络/层/板卡

YOLOv2 目标检测：MATLAB在环验证

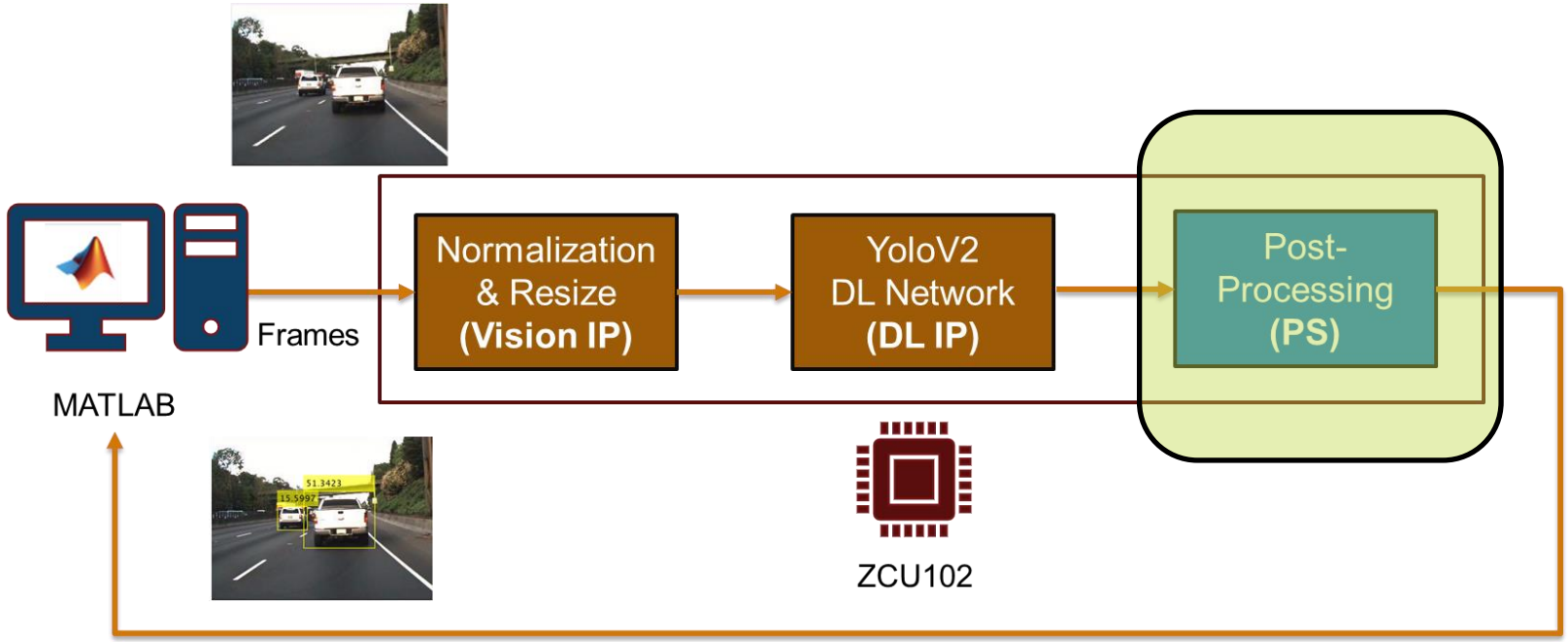
- DL 网络用于视觉处理应用，的仿真示例
- 基于YOLO v2 目标检测



Example: Deploy and Verify YOLO v2 Vehicle Detector on FPGA

YOLOv2 目标检测：SoC验证

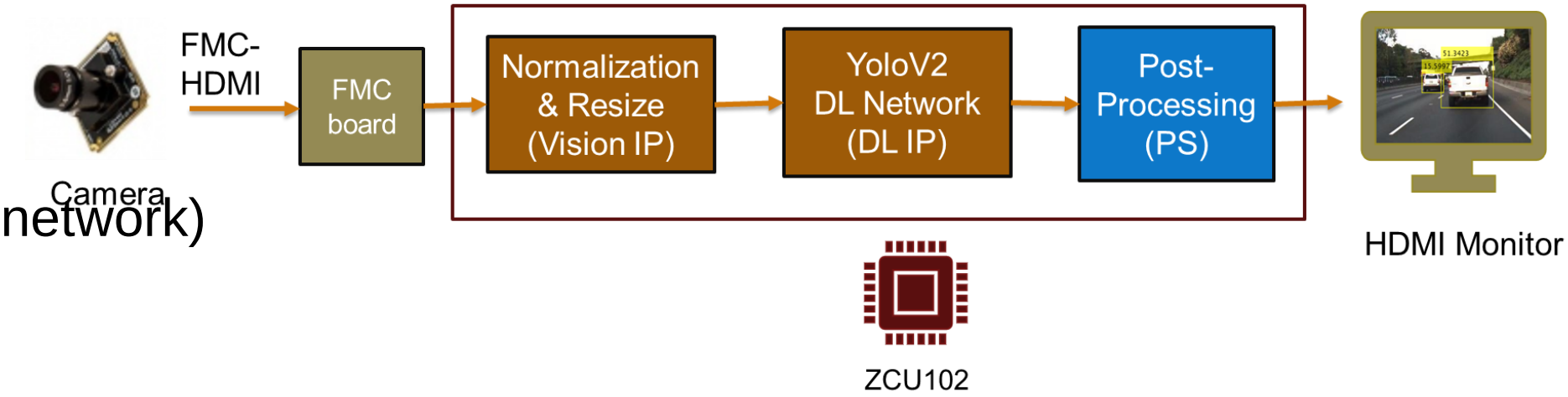
- 在PL上的前处理及PS上的后处理
- 可以仿真PL/PS握手



Example: Integrate YOLO v2 Vehicle Detector System on SoC example link

YOLOv2 目标检测：Camera实时图像流

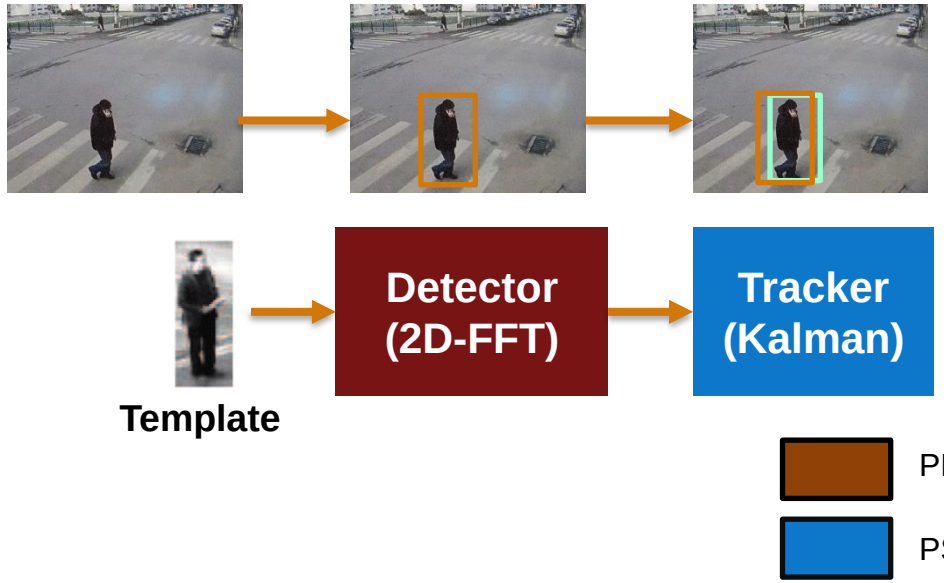
- 添加对相机实时图像的支持
- 支持1080p 输入
- 23fps （32 layer network)

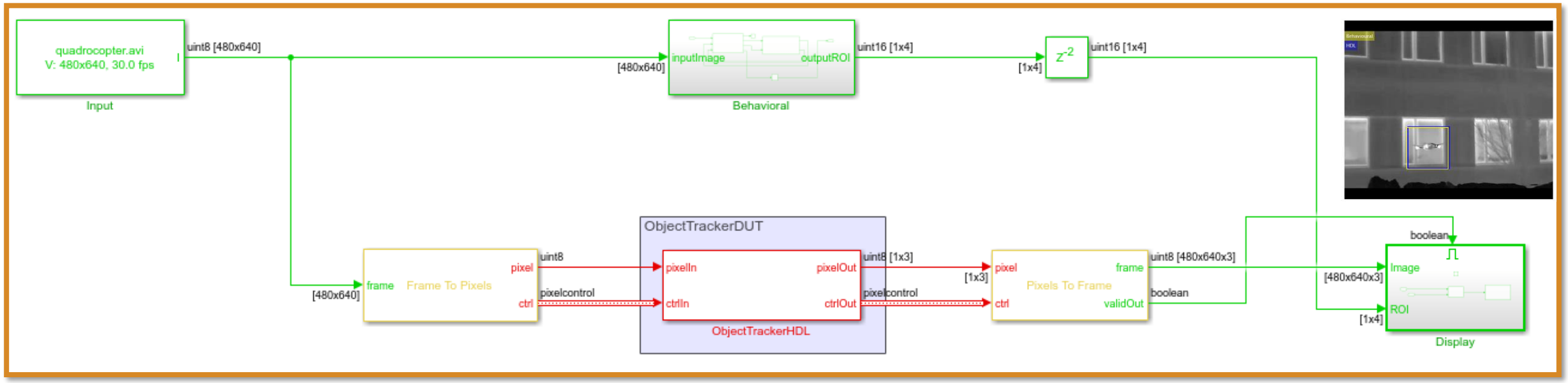


Example :YOLO v2 Vehicle Detector with Live Camera Input on Zynq-Based Hardware

高速目标检测和跟踪：2D-FFT

- 可配置参数高达：1080p@120 fps
- 模板匹配
- 检测：2D-FFT
- 跟踪：Kalman





Example :Object Tracking using 2-D FFT

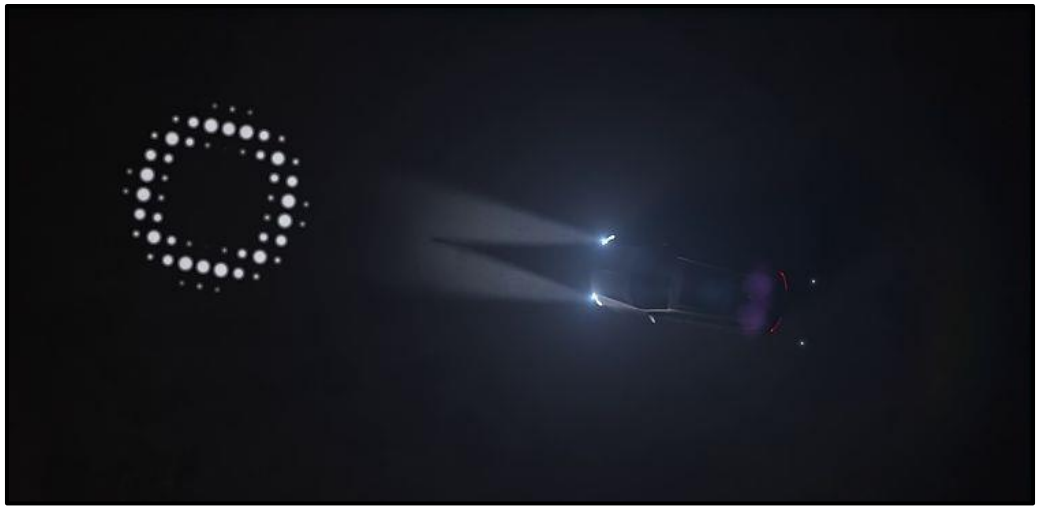
LG 电子使用快速原型工作流开发智能前照灯

挑战

为下一代智能前照灯开发视频处理和控制算法

解决方案

将基于模型的设计与 MATLAB、Simulink 和 HDL Coder 结合使用，以建立原型工作流，用于对基于 FPGA 的视频处理设计进行建模、仿真和实现



LG 电子使用 MATLAB、Simulink 和 HDL Coder 开发下一代智能前照灯。

结果

- 在原型硬件上快速实现了需求
- 缩短了编码和验证时间
- 提高了代码一致性和可重用性

“MATLAB 和 Simulink 将开发阶段所需的时间减少了一半。这些工具支持自定义函数设计，从而使我们能够轻松响应 OEM 客户的需求。”

- Jiyoung Jeong，LG 电子

Link to user story

MATLAB EXPO

谢 谢