# VLSI and Embedded (E&C)
# Presentation
# on
# "ENHANCEMENT OF NIGHT TIME VIDEO IP ACCELERATOR USING DCP"

Presented By:
Rakotojaona Andrianoelisoa Nambinina,PATEL Monal
, Electronics and Communication

Supervised by:

Prof. Carlos Valderrama (External Guide)
Head of Microelectronics Department,
University of Mons, Belgium

04/06/2020

# Outline

- Introduction
- DCP Algorithm
- Vitis unified software
- Required Component
- Programming languages
- Library used  for hardware implementation
- software/hardware implementation of algorithm
- Result
- Conclusion
- references

# Introduction

- Aim:-
  - Enhanced night input frame
  - Accelerate the video processing speed

Input frame

Enhancement

Enhanced frame

# Introduction

## Hardware available for implementation

| Criteria | CPUs | GPUs | FPGAs | ASICs |
|---|---|---|---|---|
| Processing peak power | Moderate | Very High | High | Highest |
| Power Consumption | High | Very High | Low | Very Low |
| Flexibility | Highest | Very High | Medium | Lowest |

- Parallelism design
- Consume less power
- Reconfigurable hardware

# DCP Algorithm

- Enhancement: Computer system modeled to enhanced the quality of input frame
- DCP algorithm: enhanced night input frame based on the minimum pixel value of RGB
- WHY DCP
  - Simple algorithm
  - Less component on hardware
  - Fast

# DCP Overview

- Model mathematical of McCartney

$$I(x) = J(x)\,t(x) + A\,(1 + t(x)) \quad [1]$$

I (x) : Input frame with x : coordinate of pixel (i,j)
A : Highest pixel value of I (x)
t (x) : Medium transmission
J (x): Output Frame (Enhanced frame)

- Output frame (J(x))

$$J(x) = I(x) - A\,(1 + t(x)) / t(x) \quad [2]$$

- Medium transmission (t(x))

$$t(x) = 1 - w\,I_{dark}(x) \quad [3]$$

with w: control parameter / w = 0.8

- Dark channel of Input frame (I dark (x))

$$I_{dark}(x) = \min_{(R,G,B)} I(x) \quad [4]$$

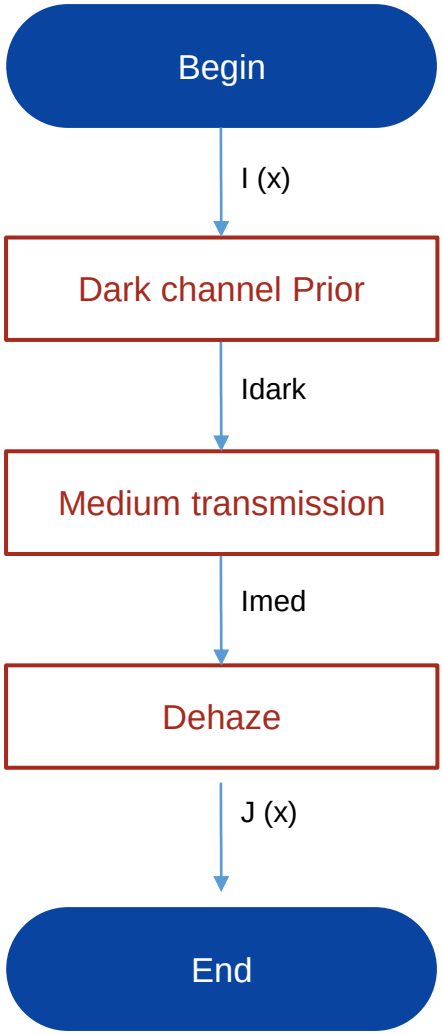
# DCP Algorithm

I (x) : Input frame

Idark : Dark input frame

Imed : medium transmission of I dark

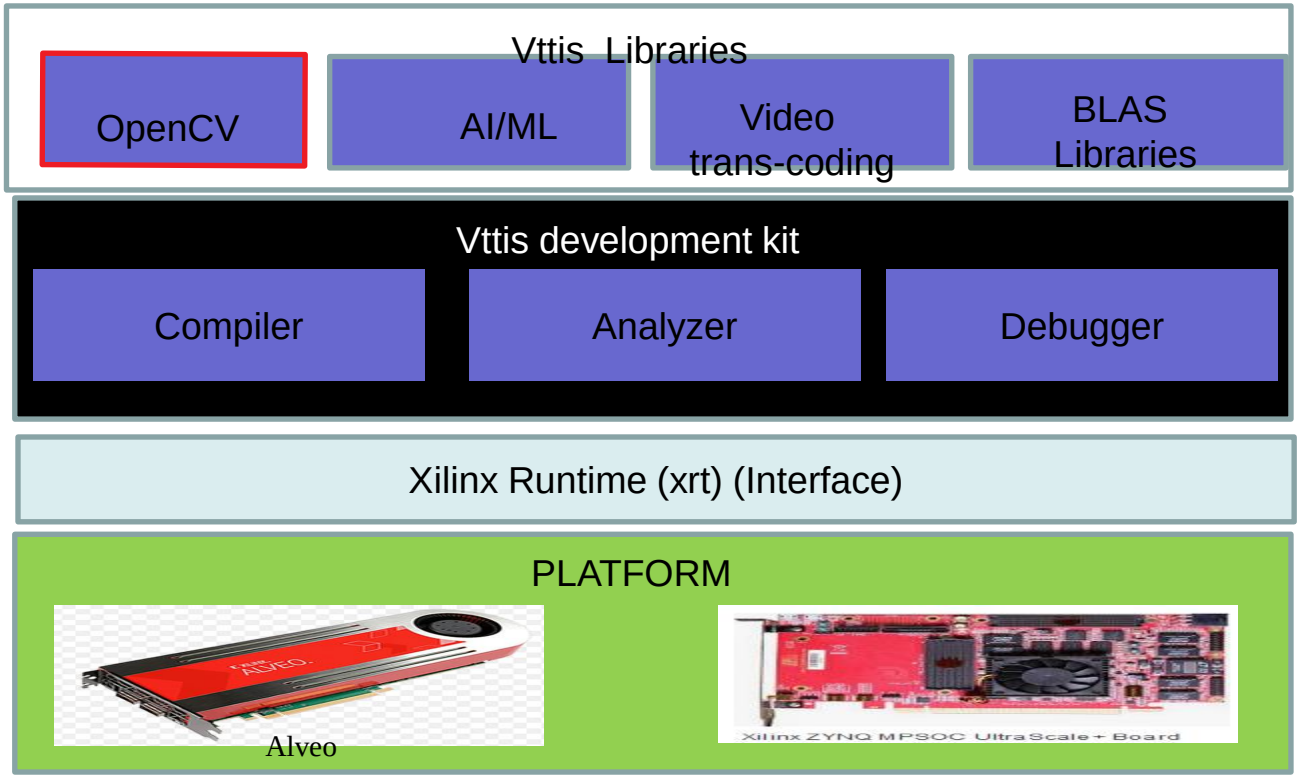
J(x) : Recover Frame)



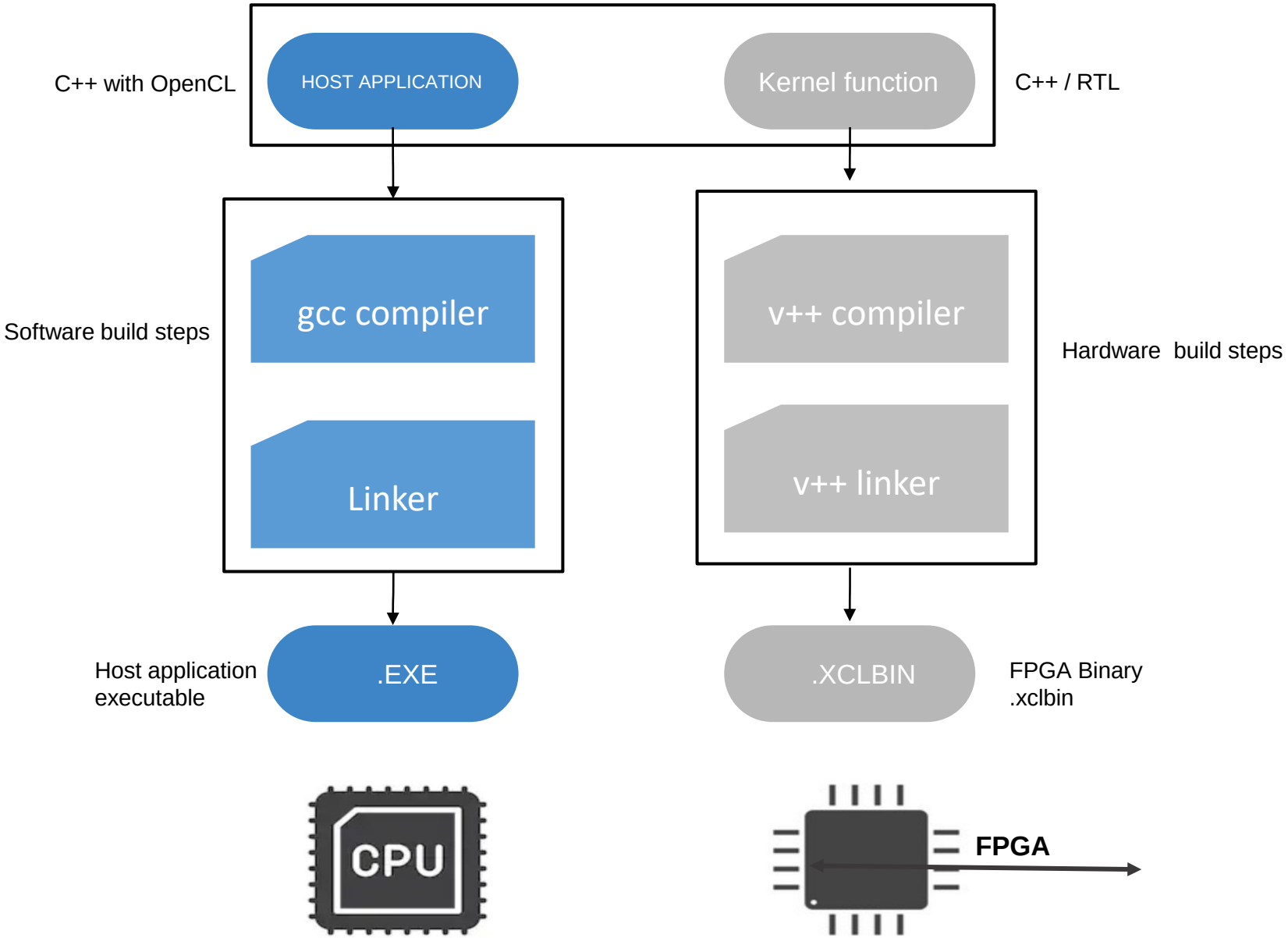
# Vitis software environment /Architecture

## Vitis unified software:

- Vitis software: Build an application software run on hardware.
- Xilinx



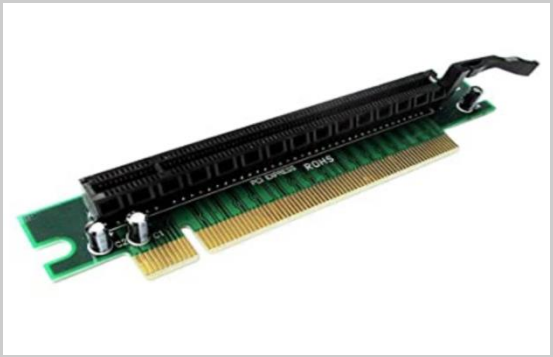
# Flow of Building application program on Vitis



# Component used to build a project


Alveo u-200

Core i-7 2.7Ghz , 16 Go Ram



PCIe Gen 3 x 16

Power supply (225 W)

# FPGA USED

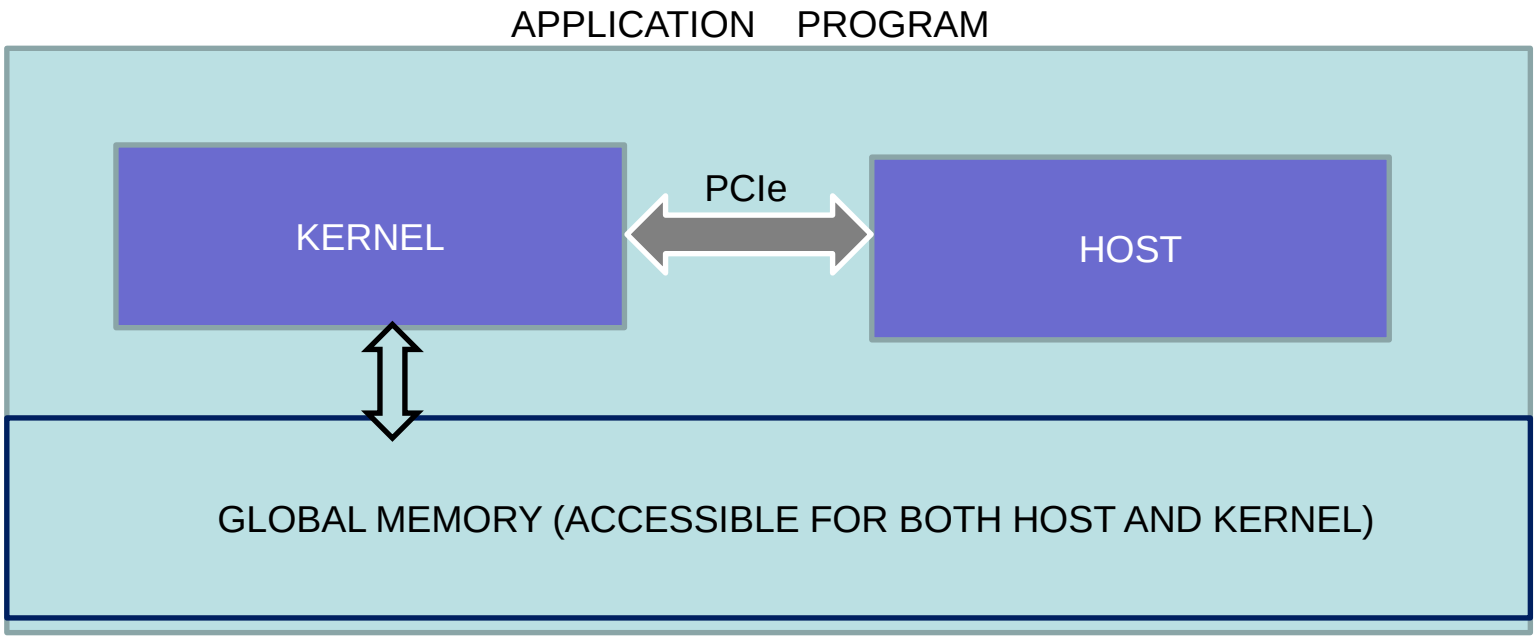
◆ALVEO CARD U200

Introduced by Xilinx
Open source
Environment : Vitis unified software

Why :

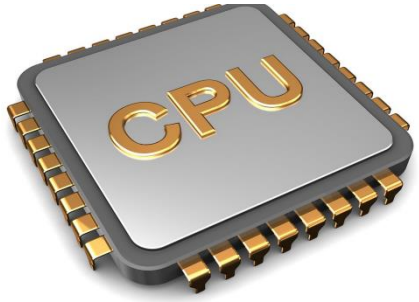
- Prototype
- Using multiple languages and Libraries, designers can benefits of programmable logic and CPU
- High frame rate video
  Low latency

# SYSTEM ARCHITECTURE



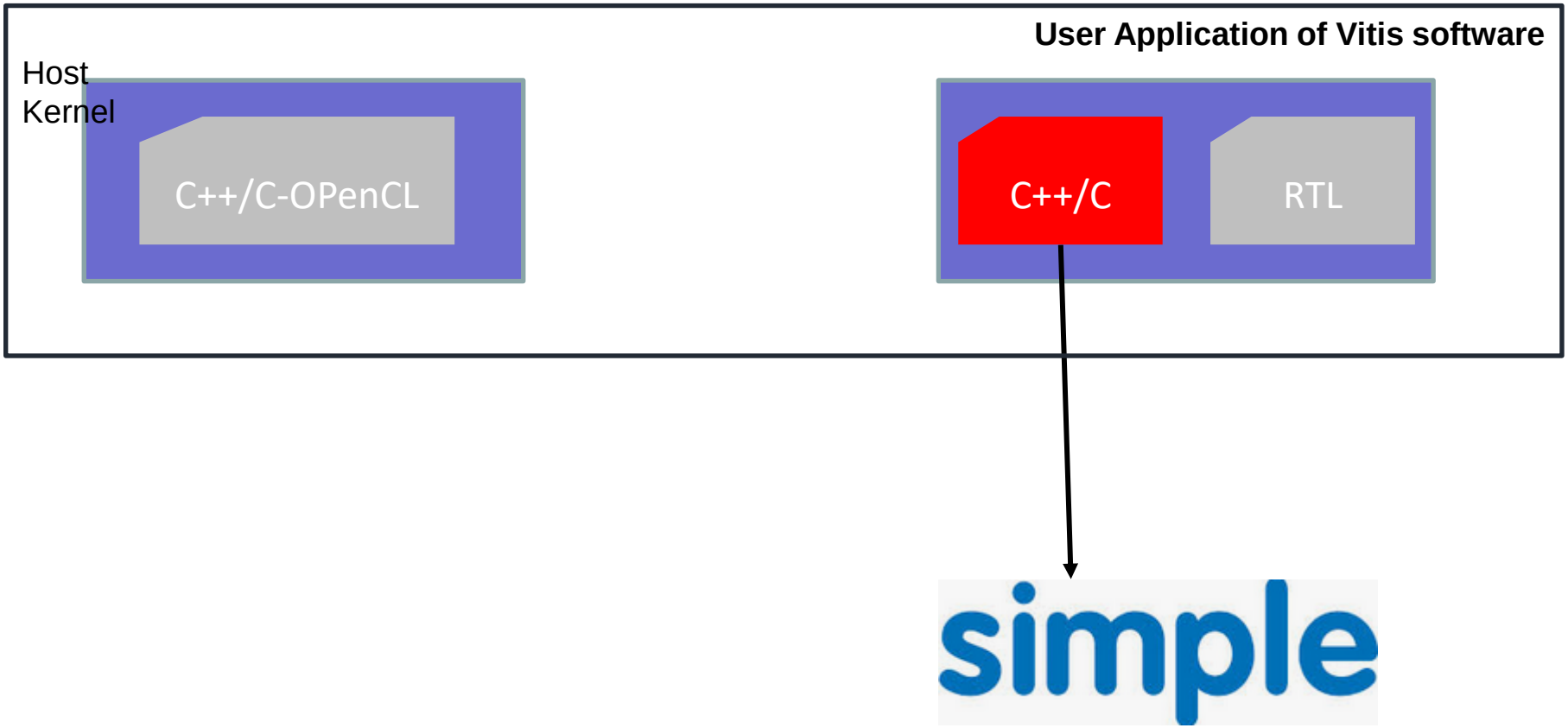


FPGA




HOST

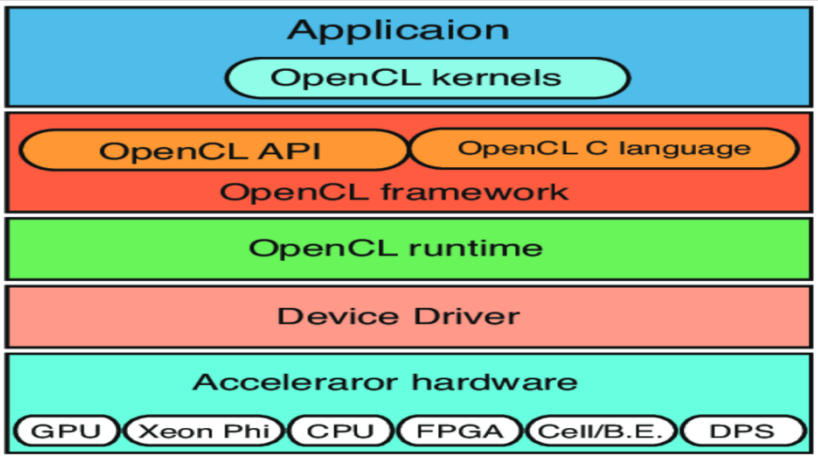
# Programming languages



# OpenCL (Host language)

- OpenCL : language used to target Hardware (FPGA/ GPU / CPU, ...)

- Command used on kernel program :
- set up the kernel
- Write/ Read data to/from Global memory
- Execute data , ...



# Library - Vitis vision

- computer vision library designed to work on Vitis software .
- Open source library
- Small architecture
- Easy and fast vs Open-CV
- Available for academic and business purpose for free

# DATA FLOW OF PROPOSED ALGORITHM

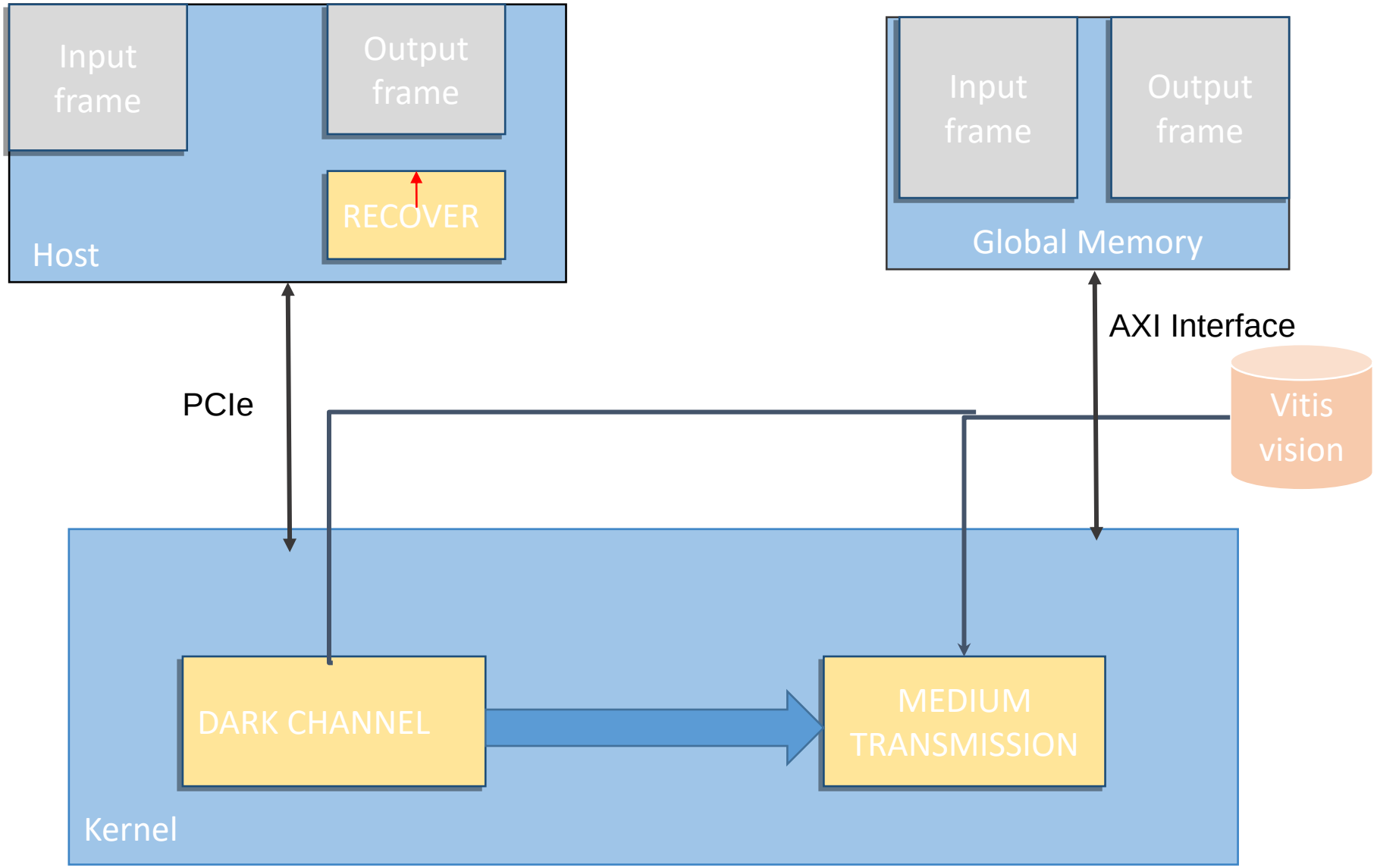
DATA FLOW

Dark channel Prior

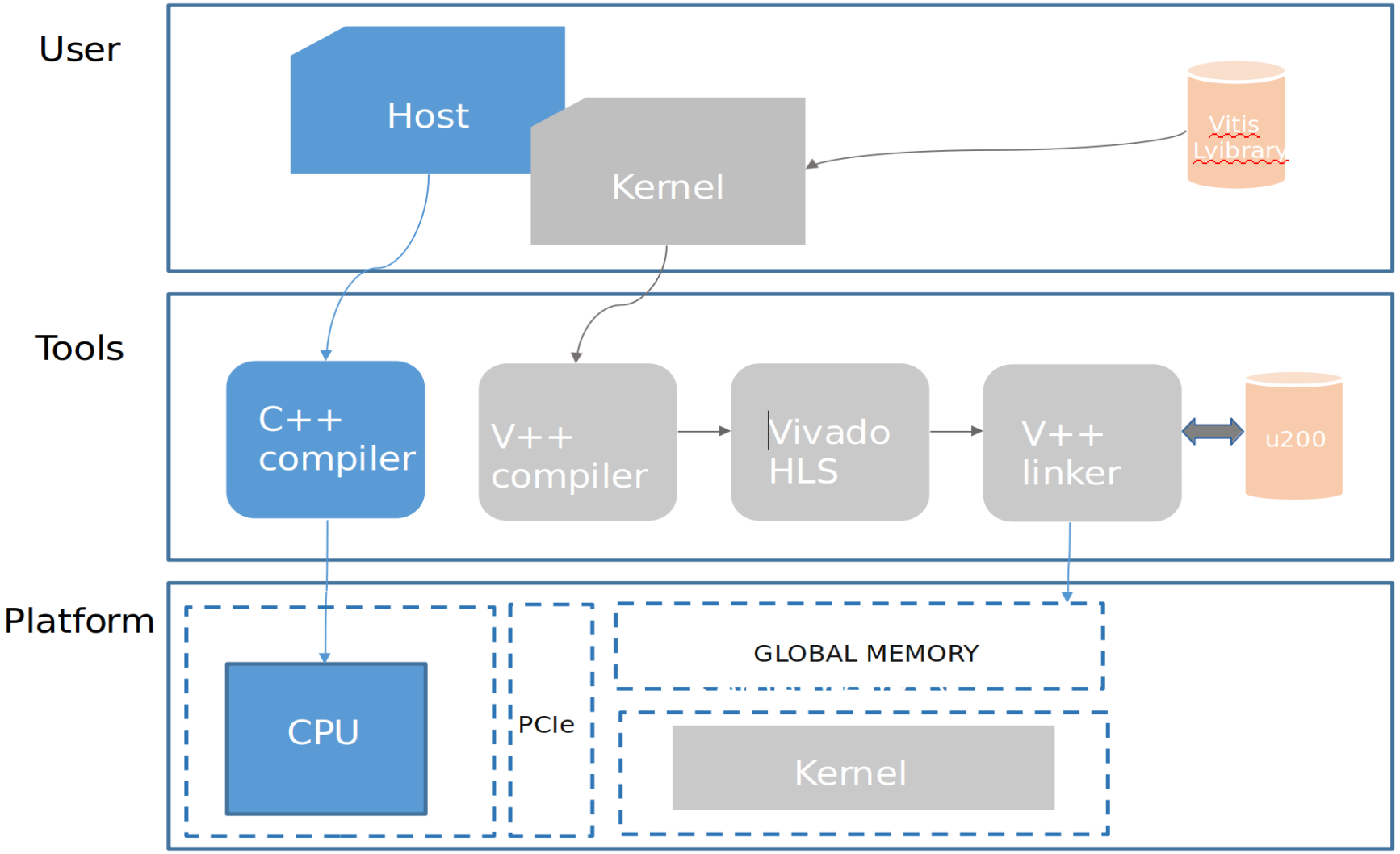
Medium transmission

Dehaze

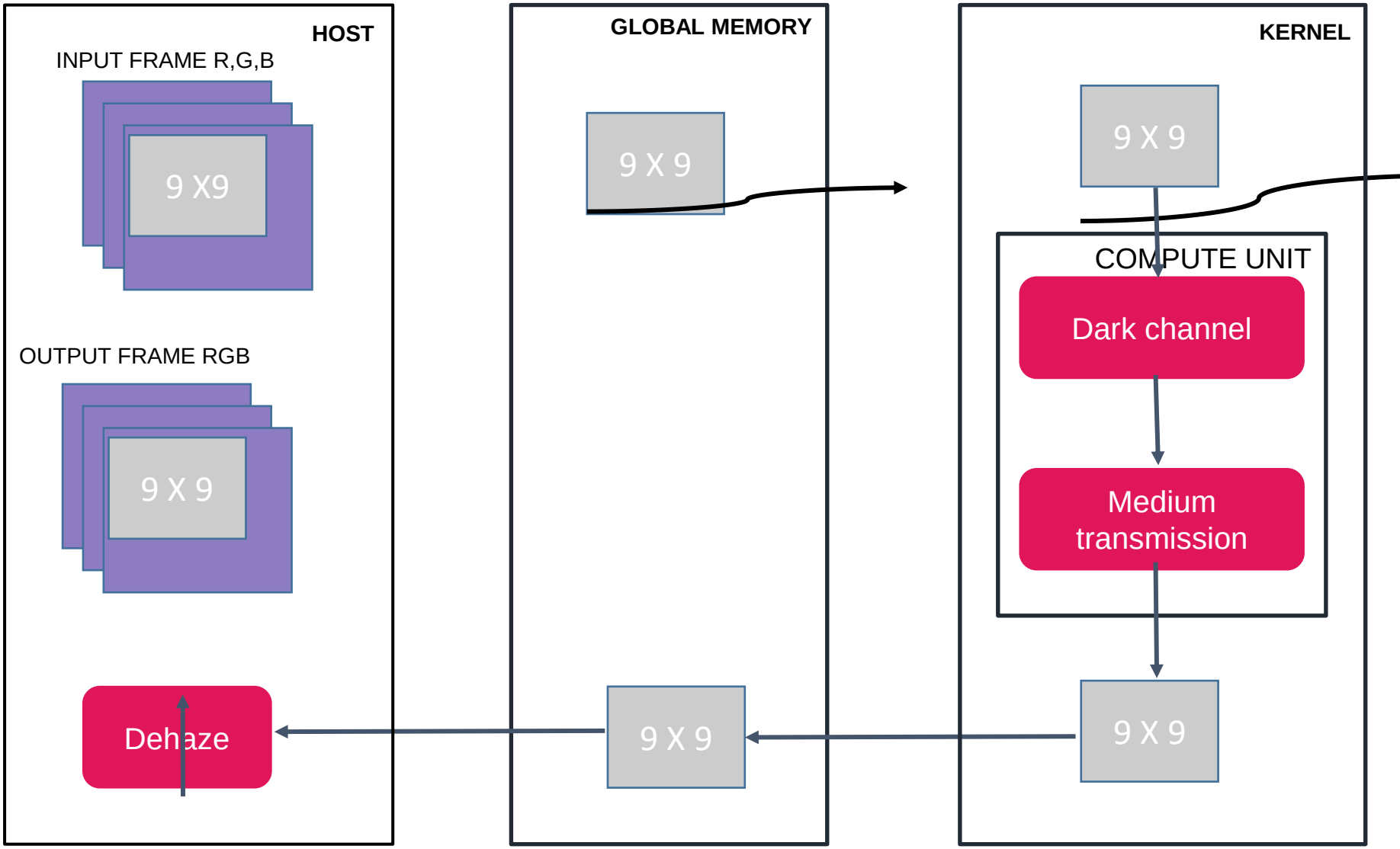
# Global architecture of proposed method



# Design flow of proposed



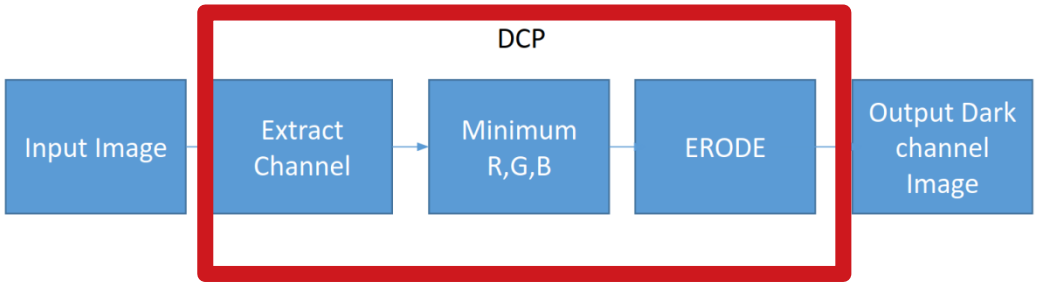
# Host kernel data flow



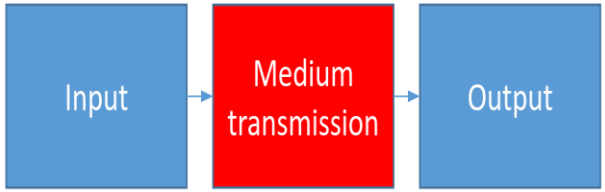
# DCP / Medium transmission on Kernel

DCP and Medium transmission function is accelerated on hardware

- DCP
  (Vitis vision)



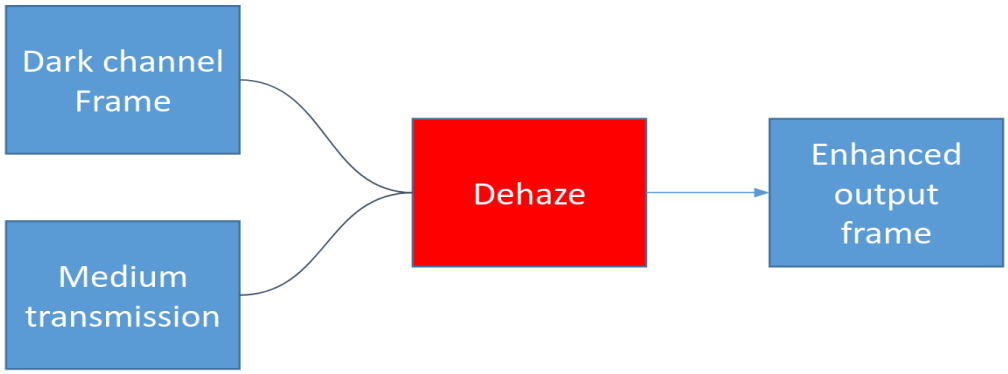

- Medium transmission



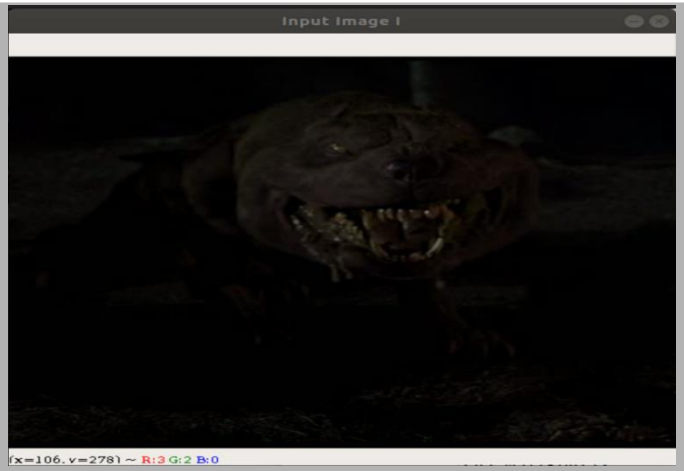
# Dehaze on CPU

Recover frame : executed on software

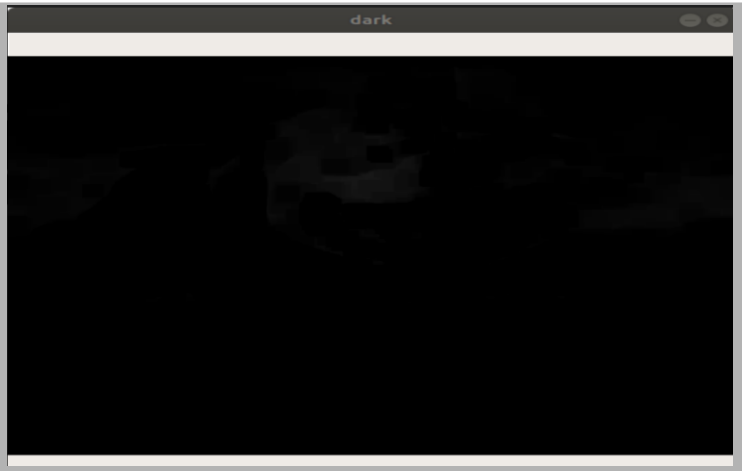
- Recover



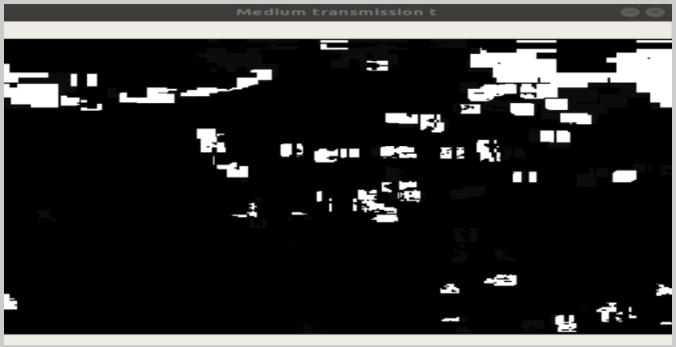
# Output




Input frame




Dark input frame




Medium transmission


Enhanced frame

# Timing summary of ap_clk

| Clock | Target | Estimated | Uncertainty |
|---|---|---|---|
| ap_clk | 3.33 ns | 2.433 ns | 0.90 ns |

# Latency

| Latency (cycles) | | Latency (absolute) | |
|---|---|---|---|
| min | max | min | max |
| 8303056 | 8321548 | 27.674 ms | 27.736 ms |

# DCP / *Medium transmission*/Dehaze Processing time on software

| Frame size | DCP | Medium transmission | Dehaze |
|---|---|---|---|
| 1600 x 900 | 15.58 s | 15.47 s | 0.144s |
| 1600 x 900 | 5.23s | 5.49 s | 0.09s |
| 1200 x 675 | 0.43s | 0.44 s | 0.02s |
| 320 x 180 | 0.05s | 0.045s | 0.06s |

- DCP and Medium transmission: consume much time
- Solution : create kernel for DCP and Medium transmission

# Comparison of software/Hardware Processing time

| Frame size | CPU (second) | Alveo board (msecond) Average running speed |
|---|---|---|
| 1600 x 900 | 31.048 | 4.86 |
| 1200 x 675 | 10.71 | 2.76 |
| 640 x 360 | 0.865 | 0.81 |
| 320 x 180 | 0.087 | 0.23 |

Hardware : 620 faster than software with frame size (1600 x 900 )

Conclusion : Efficient

# Resources used

* Summary:

| Name | BRAM_18K | DSP48E | FF | LUT | URAM |
|---|---|---|---|---|---|
| DSP | - | - | - | - | - |
| Expression | - | - | 0 | 46 | - |
| FIFO | 0 | - | 183 | 1525 | - |
| Instance | 70 | 32 | 15770 | 47107 | 0 |
| Memory | - | - | - | - | - |
| Multiplexer | - | - | - | 72 | - |
| Register | - | - | 18 | - | - |
| Total | 70 | 32 | 15971 | 48750 | 0 |
| Available SLR | 1440 | 2280 | 788160 | 394080 | 320 |
| Utilization SLR (%) | 4 | 1 | 2 | 12 | 0 |
| Available | 4320 | 6840 | 2364480 | 1182240 | 960 |
| Utilization (%) | 1 | 0.4 | 0.6 | 4 | 0 |

The algorithm consume :1 % BRAM, 0.4 % DSP48E , 0.6% FF , 4% LUT

**Conclusion** : Efficient consume less hardware resources.

# Conclusion

- Processing time of DCP on CPU: Not sufficient
- Required an hardware to accelerate DCP Algorithm.
- The DCP algorithm: used less 4 % of resources on hardware.
- Proposed algorithm efficient on hardware.
- Future work : Increased Compute unit on hardware.

# Reference

| Author | Citation | IEEE Paper/ Journal | Year of Publish | Key Learning |
|---|---|---|---|---|
| Kim, M., Park, D., Han, D. K., & Ko, H | Novel framework for extremely low-light video enhancement | IEEE | 2014 | • Enhancement of low light video<br>• Algorithm |
| Bhagya H.K Keshaveni N | Video Enhancement using Histogram Equalization with JND Mode | IJRTE | 2019 | • Enhanced image with histogram equalization |

# Reference

| Author | Citation | IEEE Paper/ Journal | Year of Publish | Key Learning |
|---|---|---|---|---|
| Z. Rahman, D. J. Jobson, and G | **Multi Scale Retinex for Color Image Enhancement,** | IEEE | 1996 | • How to use retinex for enhancement of color image |
| Xuesong Jiang, Hongxun Yao, Shengping Zhang | **Night video enhancement using improved Dark Channel Prior** | *IEEE* | 2013 | • How to implement enhancement of night time using DCP on software |

# Reference

| Author | Citation | IEEE Paper/ Journal | Year of Publish | Key Learning |
|---|---|---|---|---|
| Zhu Q, Mai J, Shao L A | **fast single image haze removal algorithm using color attenuation prior** | IEEE | 2015 | • How to use hze removal algorithm using color attenuation prior |

# Paper publication

## 1.SURVEY ON HARDWARE IMPLEMENTATION OF ARTIFICIAL NEURAL NETWORK USING FIELD PROGRAMMABLE GATE ARRAY

| | |
|---|---|
| Journal | International Journal of Research and Analytical Reviews (IJRAR) |
| Manuscript ID | Paper ID : 212763 |
| Manuscript types | Review Paper |
| Date Submitted by the Author: | December 2019 |
| Complete List of Authors: | Rakotojaona Nambinina, Haresh Suthar |
| Key word: | FPGA, neural network, activation function, look up table,piece wise linear approximation |

# Paper publication

## 1.ENHANCEMENT OF NIGHT TIME VIDEO USING DARK CHANNEL PRIOR IP ACCELERATOR

| | |
|---|---|
| Journal | International Journal of Engineering and Advanced Technology (IJEAT) |
| Manuscript ID | Paper ID: D6707049420 |
| Manuscript types | Research Paper |
| Date Submitted by the Author: | 04 march 2020 |
| Complete List of Authors: | Rakotojaona Nambinina, Haresh Suthar, Yogesh Parmar, Carlos Valderrama |
| Key word: | Image enhancement, DCP, FPGA, Vitis Xilinx software |

# Github Reference

1. https://github.com/nambhine1/Vitis-software-
2. https://github.com/Xilinx/Vitis_Libraries/tree/master/vision
3. https://github.com/nambhine1/DCP_

# Questions?

# Thank You