Embedded Systems Project Proposal - CSEE4840

Project Name: Invisibility-curtain
A Hardware Implementation
An edge image processing tool to mirror scene background against a specific foreground colour

Group Members: Srivatsan Raveendran (sr3859), Abhijeet Nayak (an3075)

[1] Project Abstract

This project aims to use the DE-1 SoC to perform edge image processing to execute the graphic effect of an invisibility cloak/curtain. The Invisibility cloak is such that if it is held before the camera, the regions of the cloak in the image disappear to reveal the original background behind it. This gives an invisibility effect to the person covered in the veil. The project aims to involve design of hardware system, interfaces, user mode programs and kernel mode drivers to obtain an end-to-end system that is capable of performing real-time camera image processing to achieve the task stated.

Fig1. A graphic illustration of the potential working of the invisible cloak in Harry Potter Cinema

[2] Image Processing background and Algorithm

A particular colored cloth is chosen to be fixed for the experiment, say 'Red.' Firstly, the background is captured, and the frame is stored. Secondly, The Red-colored cloth is detected using a color detection pipeline. The extracted BGR video frame is first converted to HSV format (Hue, Saturation, and Value). A mask of a particular hue range is chosen to segment out only the red-colored foreground. An image opening operation is performed to refine the mask depending on the detected red color. This is followed by image dilation using the same filter for opening and bitwise inversion of the mask. To generate the final output, two frames are created. One is a bitwise ANDed version of the background and the prepared mask. The second is the bitwise ANDed version of the live captured video frame and the mask. These two images are combined using weighted addition. Then the final image is dumped into the image buffer displayed on the screen.

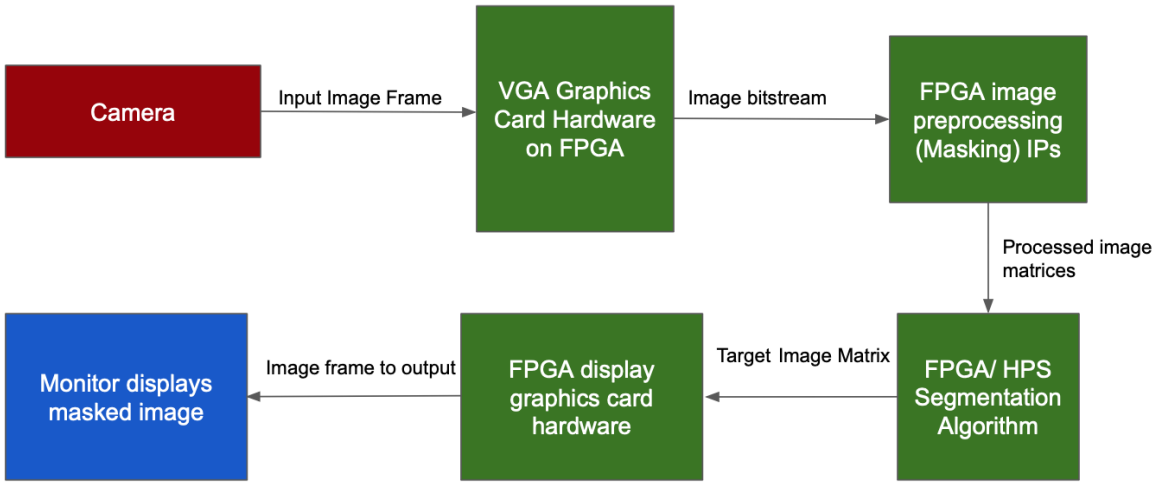

Fig 2. System Diagram

[3] Identification of IPs & Implementation in Hardware and Software:

This project's critical component involves quickly identifying integrable IPs for specific image processing tasks. The following Verilog and C IPs are of interest in this project:-

Verilog side:

a) Sobel Edge detection
b) Red/Green filter
c) VGA buffer/adapter
d) AVALON Bus Master

C side:

e) HPS program
f) OpenCV
g) Ubuntu setup/Camera drivers

Hardware Peripherals

a) Camera (3 channel colour images) - composite or web-cam
b) VGA Monitor with colour display
c) Cables and connectors

[4] Project Milestones:

Tentative Date	Action Item	Comments
03/05	Understanding IP components, VGA adapters, Avalon interconnects, HPS, algorithmic complexity	
03/12	Integrating hardware IPs, adapters, avalon interconnect etc. Interfacing Camera with DE-1.	Design documentation of successful IPs integrated
03/24	Writing Test Benches for bare metal hardware	
03/29	Writing device drivers for hardware integrated (i.e., for the video processing RTL and the)	deeper understanding of video interface and processing real time with FPGA
04/01	Mid Review	
04/15	Testing the integration with linux	
04/25	Writing user mode program to compile the kernel for the FPGA	
05/06	Testing System and debugging backgrounds	Drafting final report with debug results
05/16	Final Presentation	

[5] References:

1. Invisible Cloak using OpenCV | Python Project, https://www.geeksforgeeks.org/invisible-cloak-using-opencv-python-project/
2. DE-1 SoC Manual and Datasheet https://www.intel.com/content/dam/altera-www/global/en_US/portal/dsn/42/doc-us-dsnbk-42-1004282204-de1-soc-user-manual.pdf
3. Video capture using DE1-SoC https://hackaday.io/project/19945-video-capture-using-de1-soc-hps