Examination of Object Detection in Digital Images using FPGAs

Tim Polehna
tpolehna@iastate.edu

Abstract

In this paper, I explain the main algorithms and processes used in performing object detection within a digital image. These algorithms are shown to be basic imagine filtering techniques that can be implemented within an FPGA. Methods for improving edge detection, otherwise known as gradient filtering, and the overall speed of filtering operations are provided. Referenced articles are given that demonstrate different issues with implementing object detection and imagine filtering using FPGAs. As a comparison, additional methods for implementing these algorithms besides FPGAs are provided. Finally, problems seen with the research articles used as the basis for this paper are ezplained.

I. Introduction

Digital image and video cameras appear in more devices every day. Cameras are not simply used to capture images for keep-sakes and entertainment. Laptops, phones, video game systems, and recently even televisions have them. Beyond consumer electronics, cameras are employed by aircraft and military equipment. With all these devices, the camera is commonly used to detect specific items within the image; anything from barcodes and text to faces and retinas. Simple CMOS digital cameras are employed by (Hsiao, Yeh, Huang, & Fu, 2009) to detect vehicle lane markings.

Problems with object detection include the speed of detection, power consumed by the detection device, and the configurability of the device. Speed of detection is the most commonly addressed issue related to digital object detection. As devices are reduced in size and power, the speed at which object detection can occur reduces dramatically. General processors are extremely configurable and offer a decent ability to do image processing, but often require a lot more power than is available where the object detection is being used. Other devices, known as DSP's, that are essentially reduced functionality general purpose processors, offer lower power consumption and decent configurability. These devices are still not ideal because they still have the overhead related to being for more general purposes.

ASIC devices offer the best performance for their speed and power consumption. Unfortunately, ASICs are very expensive to manufacture. FPGAs are similar in speed to ASICs. Since an FPGA is larger and can be purchased off the shelf, it's typically the choice for performing object detection. Research papers often compare the performance of FPGA based solutions to that of DSP and general purpose processors. In particular, (Li, Yao, Tian, & Xu, 2011) program their FPGA algorithm in a DSP and PC and compare the specific timing results from different size images.

The purpose of this paper is to explorer the use of FPGA devices in object detection within digital images. Organization of the paper is as follows. Section II describes the basic techniques for image manipulation. Section III discusses the implementation of the image manipulation algorithms within an FPGA. This is followed by Section IV that explores some of the possible improvements to methods seen in other papers. Finally, the last section contains a conclusion about the how the technology will evolve.

II. Image Filtering and Enhancement

The basic principle behind image filtering or enhancement is that the data within the image is evaluated against itself to determine how it can be manipulated to achieve a certain output. In every research article explored, the digital image is washed of all color data before the filters are applied. This appears to help simplify the evaluation algorithms, which also reduces the amount of processing that has to occur. Each filtering technique is useful in certain situations (Koo, Kim, Dong, & Lee, 2002).

A. Gradient Filter

Gradient filters, also known as edge detection algorithms, are typically the most important part of object detection. A gradient filter uses what is known as a kernel mask, a matrix of defined values, to make lines within an image stand out more. The mask is multiplied by sections of the image being processed. Essentially, the color variance between a pixel and its neighbors is stored back in the image which either increases or decreases the difference between pixels.

There are multiple types of gradient filters. The most common gradient filters have what are known as Roberts, Prewitt, and Sobel operators. Sobel based filters tend to make edges become thicker which helps identify objects but also causes detail to be lost from the image. The Prewitt gradient operator was used in the majority of the research articles due to it being simple to compute and its characteristic of leaving detail in the image. These filters are applied horizontally and vertically to find edges. The result of the two directions is then combined to produce a single image of edges. (Harinarayan, Pannerselvam, Mubarak Ali, & Tripathi, 2011).

B. Median Filter

The median filter is a non-linear filter that replaces one pixel with the median value of its neighbors (Li, Yao, Tian, & Xu, 2011). A median filter is applied to an image to reduce impulse noise, which results in an image that can be described as "blurred". Noise is color variation within an image typically due to digital translation by the hardware. There are multiple types of noise, but impulse noise, also called shot noise or salt-and-pepper noise, is the most common type. Median filtering is typically used for object detection because it removes the noise without disturbing the edges within the image and typically leads to smoother edge during edge detection. Cleaner edges help with object identification later.

The pattern of pixels that the median value is taken from results in different effects that are useful in different circumstances. Six common types of pixel selection are block, horizontal, vertical, cross, scissor, and diamond (Koo, Kim, Dong, & Lee, 2002). Blocks of pixels are often used as a general filter. A block of pixels typically has dimensions of 3x3, 5x5, or 7x7 pixels. As the size of the block increases, the blurring effect increases which can lead to the elimination of small edges (Jain, Bansod, Kushwah, & Mewara, 2010). Selection of the block size depends on the type of object attempting to be detected. Objects that have a higher contrast compared to their background will benefit from a larger median filter block due to the elimination of small lines.

C. Histogram Equalization

Because edge detection relies on the difference in color between background and the object, it is beneficial to enhance the contrast of images before performing edge detection. The method of improving image contrast is known as histogram equalization. A histogram is obtained by counting the number of times each color occurs within an image. The equalization of the image is done by using the histogram to produce an equation based on the minimum and maximum values and then applying it to the pixels within the image.

III. FPGA Usage in Image Manipulation

Image enhancement filters are ideal to be implemented on an FPGA due to the fact that they have simple equations and values to apply. The values used in the filters are also the reason that FPGAs are not used for general purpose object detection. Without the ability to change the filter values, conditions such as different light conditions cause the filters to eliminate important information from the image. This effect can be seen by the change in functionality of the detection system employed in (Hsiao, Yeh, Huang, & Fu, 2009).

A. Genetic Algorithms

The ability to change filtering values is explored in (Koo, Kim, Dong, & Lee, 2002). The system created in the article has filters implemented within an FPGA that can be reconfigured based on the output of multiple filters on the image. After the image passes through the default filters, each of the images is compared with a version of the original image with noise introduced to determine how much the image has been affected. A random new value for each of the filters is set and the process is redone. A linear approximation is computed to find the final values to use in the filters. This process of comparing the filtered images to determine the optimal filters for the image is known as a genetic algorithm. Genetic algorithms are not implemented on FPGAs because of the amount of resources required to store all the values.

B. Image patterns

After filters have been applied to images to find the edges, the task of object identification still exists. Currently, object identification involves comparing a pre-recorded set of data to the values obtained. Storage of the data is why simple, well-defined objects are typically identified using FPGAs and objects such as faces are detected using general purpose processors. However, by reducing images and identifying keys parts of images, FPGAs become more feasible to detection complex objects.

In security systems that employ picture recognition, it is often necessary to reconfigure the algorithm to detect a different image (a new employee for instance). Many FPGA board have a flash interface. The problem with flash interfaces is that they tend to be slower to access from the FPGA. A method of using on-chip shift registers is examined in (Kawai, Yamaguchi, Yasunaga, Glette, & Torresen, 2008). In the end, it's determined that if recognition needs to be changed frequently, using on-chip resources is not ideal because of the amount of time required to reconfigure the device. If the reconfiguration of the device happened infrequently, using shift registers dramatically decreased the amount of time required to recognize the object.

When dealing with very complex images, the amount of data examined against the object database is very large. Thus, the more detail, the longer image recognition is going to take. Conversely, the more detail, the more accurate the detection results. Multiple levels of examination can speed up recognition as discussed in (Yigang, Yanguang, Zhuoyuan, Shengli, & Jialin, 2008). The first level of examination is the lowest resolution image. By examining the lower resolution picture first, the system can more quickly identify areas of the image that are of greater interest. Depending on levels of certainty and what is being detecting, going through all the resolution levels may not even be necessary.

IV. Issues Found with Explored Articles

In almost all the articles that were found, speed was the biggest issue. One of the articles, (Jain, Bansod, Kushwah, & Mewara, 2010), ended up coming to the conclusion that FPGAs were only useful for filtering when the image and filter were small. Looking at their approach, they used the serial UART to transfer the image data back and forth between the computer and the FPGA. The data from those tests were then compared to tests done on a PC without that transfer time. Ideally, a better approach would have been to implement the FPGA so it had access to system RAM. This would have eliminated the need to transmit the data in such a manner.

The article about using shift registers for reconfiguration, (Kawai, Yamaguchi, Yasunaga, Glette, & Torresen, 2008), mentions that the flash has reconfiguration issues due to needing the pattern already configured for storage. The authors never explain why this is the case, and why their method of reconfiguring the shift registers could not be used to reconfigure flash. It seems as though a hybrid of the two methods could be used to obtain speed of detection and speed of reprogramming.

In (Hsiao, Yeh, Huang, & Fu, 2009), the authors focused on false positive detection. Unfortunately, the authors focused on a single source of error and forgot about many others, namely other vehicles and variable driving types. As a result their solution only succeeded to prove that their system worked in basic driving situations.

V. Conclusion

All articles that were examined showed that there is a benefit of using FPGAs for object detection in images. The same filter techniques and detection can also be applied to sound. Several articles are available on the filtering and detection of specific sounds.

Different techniques were used in these articles to increase speed and accuracy of object detection in FGPAs. Speed could be achieved by performing operations in parallel or through simplification (like taking a 2-D filter and making it into multiple 1-D filters that are combined at the end). Accuracy can be improved by applying different types and levels of filters. However, with changing conditions the filters have to be adapted to the situation to avoid the loss of data.

Deployment of FPGAs to consumer devices could increase in the future as the speed, size, and power consumption of the devices improves. Consumers continually want devices that are faster and can last longer running on battery power (i.e. cell phones). Even with the current FPGA technology, real-time detection of objects was achievable.

VI. References

[1] Harinarayan, R., Pannerselvam, R., Mubarak Ali, M., & Tripathi, D. K. (2011). Feature extraction of Digital Aerial Images by FPGA based implementation of edge detection algorithms. *Emerging Trends in Electrical and Computer Technology (ICETECT), 2011 International Conference on*, 631-635 .

[2] Hsiao, P.-Y., Yeh, C.-W., Huang, S.-S., & Fu, L.-C. (2009). A Portable Vision-Based Real-Time Lane Departure Warning System: Day

and Night. *Vehicular Technology, IEEE Transactions on*, 2089-2094.

[3] Jain, T., Bansod, P., Kushwah, C., & Mewara, M. (2010). Reconfigurable Hardware for Median Filtering for Image Processing Applications. *Emerging Trends in Engineering and Technology (ICETET), 2010 3rd International Conference on*, 172-175.

[4] Kawai, H., Yamaguchi, Y., Yasunaga, M., Glette, K., & Torresen, J. (2008). An adaptive pattern recognition hardware with on-chip shift register-based partial reconfiguration. *ICECE Technology, 2008. FPT 2008. International Conference on*, 169-176.

[5] Koo, J., Kim, T., Dong, S., & Lee, C. (2002). Development of FPGA Based Adaptive Image Enhancement Filter System Using Genetic Algorithms. *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on* , 1480-1485.

[6] Li, Y., Yao, Q., Tian, B., & Xu, W. (2011). Fast double-parallel image processing based on FPGA. *Vehicular Electronics and Safety (ICVES), 2011 IEEE International Conference on*, 97-102 .

[7] Yigang, W., Yanguang, L., Zhuoyuan, W., Shengli, F., & Jialin, C. (2008). A Low Complexity and High Performance Real-Time Algorithm of Detecting and Tracking Circular Shape in Hole-Punching Machine. *Intelligent System and Knowledge Engineering, 2008. ISKE 2008. 3rd International Conference on* , 604-608.