Development of a 3D Disparity Estimation Processing Architecture

Lean Karlo S. Tolentino
1* Electronics Engineering Department, College of Engineering, Technological University of the Philippines, Manila 1000, Philippines.
ORCID ID: 0000-0002-8014-8229

Debee Michelle T. Beleno
2 Statistician I/NSTP Department, Technological University of the Philippines, Manila 1000, Philippines.
ORCID ID: 0000-0002-8060-3893

Abstract

The dislocation of Camera focus from the movement of an object in mono-camera or variation of an object in stereo-camera. Using the Digital Image Processing can correct this problem. Projection is one of these Digital Image processes. It is accumulated axes of data pixel and utilizes to process like correlation. In this paper, we propose a hardware implementation technique of divided region projection for Real-time disparity correction. By dividing the whole image into three-sub regions (left, central and right partition). We have verified some advantages such as less computation in high estimation precision, and Central Region's projection value, one of the three-sub regions, use to correct shift value for real-time image stabilization.

Keywords: 3D, correlation, disparity estimation

INTRODUCTION

Digital Image Processing Technology comes a long way from the first photograph ever produced to digital imaging photos that do not take a long time to process. The latest development occurred was now adopted in the field of military applications. Radar, secure communications, and electro-optics are some of the arrays in military field that benefited from the digital image processing. Electronic sensors at the core of these systems continue to evolve, requiring more digital processing functionality and integration.

Image processing is an algorithm that takes an image as input and returns an image as output. It includes image display and printing, image editing, manipulation, morphology and color correction, also image enhancement, and feature detection as well as image compression [1][2][3][4].

The stereo camera should be in a high-speed processing in order to handle the real-time image. The real-time image should be at a large number of operation (like the number of arithmetic) process in hardware because of its difficulty in high-speed processing that requires only in software [5][6].

Projection algorithm is an algorithm for determining the information to the projection of the axis component in the entire image data, and uses time in determining the correlation that mainly detects the subject's movement.

However, the common problem in Projection Algorithm when using for real-time video is throughput. Each axis component in all the operations are being saved. But the needed buffer’s size and computation time are necessary because it cannot be processed in real time [7][8].

In this paper, the whole image data processed can be reduced by at least half by the cumulative borrowing using global processing methods both from the region partitioning scheme.

DISPARITY ESTIMATION PROCESSING

Edge detection using Sobel Mask

Sobel mask filter extracts contour data as well as the contour data of the Left and the Right of video images values that stacked in vertical component. Vertical cumulative was used for the correction of distance difference. Vertical accumulation means that the accumulated treatment data for the vertical contour component in one line buffer. Therefore, the two cameras are on the same horizontal is the extracting contour data from the Sobel mask, and the calculated and the assumed cumulative value of the vertical component extracted by the contour data projection process[9].

Then the left and the right video of the vertical component images will then compare the value to find the distance difference and identify the area of the minimum distance of the accumulated value. The minimum distance value can be found through the Correlation process; that used to correct the image.

Extracting the image data in outline through the Sobel Mask may create an image having the intensity of contour as shown in Figure 1 [10].
The assumption by default that a horizontal axis within the same and to derive the horizontal distance difference as shown in Figure 1 and using the same contour image must go through the process of finding a font that resembles the Projection Correlation process.

**Projection Algorithm**

Projection process means that the image of the new line in each pixel value is the size of the buffer in the vertical as well as in the horizontal of the cumulative value refers to the cumulative value that stored in each vertical.

Each of enhanced value of Projection Data and Target Image Reference Image will obtain the horizontal distance by finding the difference in the degree of resemblance Correlation process.

Equation (1) represents the calculation of the Vertical Projection data for accumulating pixel values of each vertical line.

\[ P_v(i) = \sum_{j=1}^{N_v} Y(i, j) \]  

(1)

Figure 2 shows the Vertical Projection. For better understanding, this figure shows the vertical axis, the image line-by-line data is the accumulated value from the buffer that stored each of the vertical sizes in the horizontal axis [9].

**Correlation Algorithm**

Correlation process can be described as shown in Figure 3. Reference to Target image in viewing the images, the right eye for the right camera as well as the left camera for the left eye. The left eye is more visible on the left side than the right eye, left eye can see more horizontally on the same line. In seeing things in left cameras is biased to the left, while the right side of the camera is concentrated on the right (lean to one side).

Comparing the difference is not actually the difference between the portions of the entire screen. The search area was established in the target image of the certain area. The search area and certain part of the Reference image are compared with each other. In the certain area of Reference, image was moved one by one in repeated process. Then find the minimum distance range extracted difference value of the distance difference correction.

Equation (2) shows the Correlation value of figure 4, the calculation of the calculation process is to obtain the minimal distance difference value.

\[ C_v(w) = \sum_{i=1}^{M_v} P_v^l(i + w - 1) - P_v^r(i + S) \]  

(2)

Figure 5 shows the Original Left Image, Right Stereo Image, while Figure 6 shows the Left Image before correction of Right Image is the project procedures and correlation process through the difference value between the minimum distances of the two video image detected in the correction of the completed image.
PROJECTION REGION SEGMENTATION HARDWARE DESIGN

The small area was shortened due to the calculated sub-areas that were divided into 3 parts. Therefore, it is a good implementing Projection in real time. The sub-area reference frame and target frame divided into three sub-regions of the (left, right, central).

Figure 7 shows the results of horizontal and vertical gray projection value that can verify the effective width of the “S” left and right image.

Figure 8 shows from the test image using MATLAB 7.1 Tool to the detection of the histogram of Projection X. The 2S values can be found by excluding the corrected horizontal distance difference in utilizing the Projection X.

Using MATLAB tool it accumulates Projection X into outline data. The so hist function is expressed in the graph is shown in Figure 8. It described the changes of the image from M 0 to M-1. The instantaneous rate of change can be extracted without constant interval from the rate of change. In Figure 8, by using 70 and 230 S can be determined. Then, the image size M using an axis of 160 M is reduced by 160/300 = 0.58. So, the image size W = MXN can be reduced to 1/2. The section of S does not have the constant rate of change that can be excluded and the remaining is accumulated.

Therefore it can minimize the central portion that can be found. The memory is reduced due to partitioning process. From the Projection as shown in Equation (3) the divided three sub-areas of the central area accumulation process. MATLAB tool shows the graph of vertical axis contour cumulative data. The graph shows the finding of the value of the rapidly changing 2S.

\[ P_{xc}(y) = \sum_{i=0}^{M-2S-1} f(i, y), \quad P_{xc}(x) = \sum_{i=0}^{N-1} f(x, y) \]

\[ x \in [S, M - 2S - 1], \quad y \in [0, N - 1] \quad (3) \]

From the entire video image, the value of the no change 2S projection can be excluded. The Minimized Central Area of segmented Projection can be cumulative.
Figure 9: Divided Region Projection Hardware Block

Figure 9 shows the designed to segmented Projection. Using window dividing scheme, small DPSRAM can be implemented in Correlation Processing on a latter part.

Table 1: Synthesis Results of Original Disparity Correction FPGA

<table>
<thead>
<tr>
<th>Logic Utilizations</th>
<th>Used</th>
<th>Utilizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slices</td>
<td>769</td>
<td>1%</td>
</tr>
<tr>
<td>Number of Slices Flip Flops</td>
<td>691</td>
<td>1%</td>
</tr>
<tr>
<td>Number of 4-input LUTs</td>
<td>1354</td>
<td>2%</td>
</tr>
<tr>
<td>Number of Bonded IOBs</td>
<td>157</td>
<td>24%</td>
</tr>
<tr>
<td>Number of BRAMs</td>
<td>6</td>
<td>5%</td>
</tr>
</tbody>
</table>

Table 1 shows the synthesis results of the original disparity correction algorithm using FPGA. While in Table 2 shows the synthesis results of the proposed disparity correction using FPGA. As seen in the synthesis result, the number of sliced F/F was reduced from 691 to 560 while the 4-input LUT reduced by 42.4% from 1354 to 780. In implementation result, Flip-flop usage is decreased. Also, we can solve the memory problem which is too large on a logic area that is unable download. In addition, implementation block of other correction can be added.

Table 2: Synthesis Results of Proposed Disparity Correction FPGA

<table>
<thead>
<tr>
<th>Logic Utilizations</th>
<th>Used</th>
<th>Utilizations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slices</td>
<td>660</td>
<td>1%</td>
</tr>
<tr>
<td>Number of Slices Flip Flops</td>
<td>560</td>
<td>1%</td>
</tr>
<tr>
<td>Number of 4-input LUTs</td>
<td>780</td>
<td>1%</td>
</tr>
<tr>
<td>Number of Bonded IOBs</td>
<td>101</td>
<td>15%</td>
</tr>
<tr>
<td>Number of BRAMs</td>
<td>6</td>
<td>5%</td>
</tr>
</tbody>
</table>

Figure 10 shows the output images of the Sample Image after utilizes the Disparity compensation using the conventional and proposed method of hardware design.

REFERENCES


Normalized Difference Vegetation Index via Image Processing of In-Field Drone-Captured Image.” Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 9(2-6), pp. 1-5. eISSN: 2289-8131


