

## 거리 측정 센서의 위치와 각도에 따른 깊이 영상 왜곡 보정 방법 및 하드웨어 구현

장경훈 · 조호상 · 김근준 · 강봉순\*

### Depth Image Distortion Correction Method according to the Position and Angle of Depth Sensor and Its Hardware Implementation

Kyounghoon Jang · Hosang Cho · Geun-Jun Kim · Bongsoon Kang\*

Department of Electronic Engineering, Dong-A University, Busan 604-714, Korea

#### 요 약

디지털 영상 처리 분야에서 사람의 동작 인식은 다양하게 연구되고 있으며, 최근에는 깊이 영상을 이용한 방법이 매우 유용하게 사용되고 있다. 하지만 깊이 측정 센서의 위치와 각도에 따라 깊이 영상에서의 객체 크기나 형태가 왜곡되므로 사물 및 사람의 인식 과정에서 인식률이 감소하는 경우가 발생한다. 따라서 뛰어난 성능을 보장하기 위해서는 측정 센서에 의한 왜곡 보정은 반드시 고려되어야 할 사항이다. 본 논문에서는 동작 인식 시스템의 인식률을 향상시키기 위한 전처리 알고리즘을 제안한다. 깊이 측정 센서로부터 입력되는 깊이 정보를 실제 공간 (Real World)으로 변환하여 왜곡 보정을 수행한 후 투영 공간 (Projective World)으로 변환한다. 최종적으로 제안된 시스템을 OpenCV와 Window 프로그램을 사용하여 구현하였으며 Kinect를 사용하여 실시간으로 성능을 테스트하였다. 또한, Verilog-HDL을 사용하여 하드웨어 시스템을 설계하고, Xilinx Zynq-7000 FPGA Board에 탑재하여 검증하였다.

#### ABSTRACT

The motion recognition system has been broadly studied in digital image and video processing fields. Recently, method using th depth image is used very useful. However, recognition accuracy of depth image based method will be loss caused by size and shape of object distorted for angle of the depth sensor. Therefore, distortion correction of depth sensor is positively necessary for distinguished performance of the recognition system. In this paper, we propose a pre-processing algorithm to improve the motion recognition system. Depth data from depth sensor converted to real world, performed the corrected angle, and then inverse converted to projective world. The proposed system make progress using the OpenCV and the window program, and we test a system using the Kinect in real time. In addition, designed using Verilog-HDL and verified through the Zynq-7000 FPGA Board of Xilinx.

**키워드** : 동작 인식, 전처리, 거리 측정 센서, 깊이 영상, 왜곡 보정

**Key word** : Motion Recognition, Pre-processing, Depth Sensor, Depth Image, Distortion Correction

접수일자 : 2014. 02. 12 심사완료일자 : 2014. 03. 05 게재확정일자 : 2014. 03. 16

\* **Corresponding Author** Bongsoon Kang (E-mail:bongsoon@dau.ac.kr, Tel:+82-51-200-7703)

Department of Electronic Engineering, Dong-A University, Busan 604-714, Korea

**Open Access** <http://dx.doi.org/10.6109/jkiice.2014.18.5.1103>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.  
Copyright © The Korea Institute of Information and Communication Engineering.

## I. 서 론

최근 디지털 기기에 고성능 다기능의 부품들이 요구됨에 따라 사용자와 디지털 기기간의 인간과 컴퓨터의 상호작용(Human Computer Interface, HCI)에 관한 연구는 매우 활발하게 진행 되고 있고 사용자 인터페이스(User Interface)의 중요성이 증가되고 있다. 사용자 인터페이스는 사용자와 시스템 사이에서 신속하고 정확하게 정보를 전달하여야 하며, 도구의 사용 없이 디지털 기기들을 제어할 수 있는 것이 가장 이상적인 방법이다. 따라서 사람의 위치나 동작을 인식하여 디지털 기기와 상호작용 할 수 있는 기술은 차세대 핵심 기술로 주목을 받고 있다. 동작인식에 관한 연구는 오래 전부터 진행되어 왔는데, 기존에는 카메라로부터 입력 받은 컬러 영상만을 분석하는 방법을 사용하였다. 하지만, 최근에는 Time-of-flight range (TOF) 카메라나 Kinect와 같은 거리 측정 센서가 대중화 되었고 이를 활용한 동작인식 방법이 활발하게 연구 되고 있다[1].

거리 측정 센서를 이용하여 사람의 동작을 효과적으로 인식 하기 위해서는 배경과 전경을 분리하여 움직이는 객체만을 추출하는 기능이 필수적이다[2]. 또한, 움직이는 객체의 각 신체 부위가 어떠한 행동을 하는지 판단하여야 사람의 동작으로 디지털 기기를 효과적으로 제어가 가능하다. 거리 측정 센서를 이용하여 배경과 전경을 분리하기 위해서는 입력되는 깊이 영상을 깊이 값에 따라 분할하여 깊이 영상 내의 객체를 구분하는 방법이 많이 사용되고 있다. 이에 따라 동작 인식 시스템의 인식을 향상을 위해서는 입력되는 깊이 영상에서 깊이 값에 따라 영상내의 객체 들을 정확하게 분할하여야한다[2]. 하지만 입력 영상에서 깊이 값에 따라 객체를 분할할 경우, 거리 측정 센서의 위치나 각도에 따라 촬영된 영상에서 객체의 크기나 형태, 깊이 값이 다르게 표현된다. 따라서 거리 측정 센서가 촬영할 객체의 정면이 아닌 위쪽이나 아래쪽에 위치하여 다른 각도에서 촬영된다면 하나의 객체가 다른 객체로 분할될 가능성이 존재한다. 이러한 경우 동작 인식 시스템에서 하나의 객체로 인식 하지 못하고 분리된 객체로 인식 하기 때문에 시스템의 인식률이 감소하게 된다. 따라서 동작 인식 시스템의 인식을 향상을 위해서는 전처리 단계로 거리 측정 센서의 위치와 각도에 따른 영상 왜곡 보정 방법이 필수적이다.

기존의 영상 왜곡 보정을 위한 방법으로는 임의의 점들을 이용하여 영상의 기하학적 변환을 하는 직접선형변환(Direct Linear Transform, DLT) 방법과 퍼지 함수를 통한 왜곡 보정과 공간 좌표 상에서 기하학적 변환을 수행하는 어파인 변환(Affine Transform) 등이 있다[2, 3]. 직접선형변환 알고리즘을 이용하여 영상의 왜곡 보정을 수행 할 경우 특이 값 분해(Singular Value Decomposition, SVD) 연산 등의 복잡한 행렬 연산으로 인하여 연산 량이 증가한다. 또한 퍼지함수와 같은 뉴럴 네트워크 방법의 경우 학습 알고리즘이 포함되기 때문에 학습 시간과 많은 연산이 필요하다. 이에 따라 기존의 직접선형변환이나 퍼지함수 등을 이용한 방법은 실시간 하드웨어 구현이 불가능 하고 연산 소모 시간이 증가하게 된다.

따라서 본 논문에서는 거리 측정 센서를 이용한 동작 인식 시스템의 전처리 과정으로써 실시간 처리와 하드웨어 구현이 가능한 깊이 영상 왜곡 보정 시스템을 제안한다. 거리 측정 센서로부터 입력되는 깊이 영상은 투영 공간(Projective world)으로 표현되어 2차원 배열에 각 픽셀에 대한 깊이 값으로 구성되는 데이터이다. 따라서 영상 왜곡 보정 기능을 효과적으로 적용하기 위하여 3차원 점군 데이터로 구성되는 실제 공간(Real world)으로 변환하여, 간단한 행렬 연산을 하는 기하학적 변환 수식인 어파인 변환을 이용한 왜곡 보정 기능을 적용한다. 왜곡 보정이 적용된 3차원 점군 데이터정보는 깊이 영상으로의 출력을 위하여 Projective world로 변환되어 출력된다. 제안한 방법은 Microsoft Visual Studio 2010과 OpenCV를 이용한 윈도우 검증 프로그램을 통하여 성능을 검증하였다. 또한 Verilog-HDL을 이용하여 하드웨어로 구현하고 Field Programmable Gate Array (FPGA)와 Central Processing Unit (CPU)을 연동하여 검증 가능한 Zynq-7000 FPGA Board를 이용하여 실시간 처리를 확인 하였다.

본 논문의 II장에서는 제안하는 거리 측정 센서의 위치와 각도에 따른 영상 왜곡 보정 알고리즘에 대하여 논하고 III장에서는 제안한 알고리즘의 시뮬레이션 결과에 대하여 논하고 성능을 평가한다. IV장에서는 왜곡 보정 시스템의 하드웨어 설계에 관하여 서술하고 Zynq-7000 FPGA Board를 이용한 합성 결과와 성능에 대하여 논한다. V장에서는 결론을 맺고 효용 가치에 대해서 언급한다.

## II. 거리 측정 센서의 위치와 각도에 따른 영상 왜곡 보정 방법

제안한 거리 측정 센서의 위치와 각도에 따른 영상 왜곡 보정 방법은 입력되는 깊이 영상의 효과적인 왜곡 보정을 위하여 3차원 점군 데이터로 구성되는 실제 공간으로 변환하는 좌표 변환과 거리 측정 센서의 위치나 각도에 따라 왜곡을 보정하기 위한 3차원 회전, 투영 공간으로의 변환을 위한 역 좌표 변환으로 구성 된다 [4-6].

### 2.1. 3차원 좌표 변환

거리 측정 센서로부터 입력되는 깊이 정보는 투영 공간에 대한 표현이며, 2차원 픽셀 배열에 깊이 정보들로 구성된다. 효과적인 왜곡 보정을 위하여 좌표 변환 단계에서 2.5차원으로 표현된 입력 영상의 깊이 정보를 실제 공간에 대한 표현인 3차원 점군 데이터로 변환한다. 거리 측정 센서의 측정가능 거리를 벗어난 경우에 3차원 점군 데이터로 재구성하면서 원점으로 모이는 현상을 제거하기 위하여 수식 (1)과 같이 깊이 정보를 제한한다. 입력되는 깊이 정보  $D_i(x,y)$ 가 0일 때, 측정가능 거리를 벗어난 경우이므로 입력되는 깊이 정보의 최대 비트 크기인  $depth\_max\_bit$ 를 이용하여 제한된 깊이 정보  $\widehat{D}_i(x,y)$ 를 생성한다[2].

$$\widehat{D}_i(x,y) = \begin{cases} 2^{depth\_max\_bit} - 1, & \text{if } D_i(x,y) = 0 \\ D_i(x,y), & \text{otherwise} \end{cases} \quad (1)$$

수식 (2)~(4)는 3차원 점군 데이터 변환 수식이다 [4]. 수식 (2)는 3차원 점군 데이터의 x축 좌표  $p_x(x,y)$ , 수식 (3)은 y축 좌표  $p_y(x,y)$ , 수식 (4)는 z축 좌표  $p_z(x,y)$ 이다. 변환 수식 (2)~(3)에서  $w$ 는 입력 영상의 가로 크기를 나타내고  $h$ 는 세로 크기를 나타낸다. 또한  $x, y$ 는 입력 영상의 픽셀 위치이고,  $f_h$ 는 Horizontal FoV(Field of View),  $f_v$ 는 Vertical FoV이다 [4]. 수식 (3)의 axis는 3차원 회전을 적용할 때 회전축의 위치를 결정하는 0에서 1사이의 값을 가지는 파라미터이다. 회전축의 위치에 따라 3차원 회전 적용 결과가 달라지기 때문에 센서의 상하 각도 보정을 위하여 Height 축에서 회전축의 위치를 선택할 수 있도록 하였다.

$$p_x(x,y) = \widehat{D}_i(x,y) \times \left( \frac{x}{w-1} - 0.5 \right) \times \tan \left( \frac{f_h}{2} \right) \quad (2)$$

$$p_y(x,y) = \widehat{D}_i(x,y) \times \left( \frac{y}{h-1} - axis \right) \times \tan \left( \frac{f_v}{2} \right) \quad (3)$$

$$p_z(x,y) = \widehat{D}_i(x,y) \quad (4)$$

그림 1은 3차원 좌표 변환을 적용한 결과이다. 그림 1의 (a)는 제한된 깊이 영상의 2.5차원 정보이며, 그림 1의 (b)는 3차원 점군 데이터 정보이다.

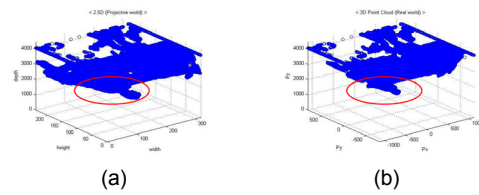


그림 1. 3차원 좌표 변환 적용 결과 (a) 2.5차원 정보, (b) 3차원 점군 데이터 정보  
Fig. 1 Coordinate Conversion Result (a) 2.5D Data, (b) 3D Point Cloud Data

### 2.2. 3차원 회전

3차원 회전은 거리 측정 센서의 위치와 각도에 따른 영상 왜곡을 보정한다. 수식 (5)는 어파인 변환을 사용하여 x축을 기준으로 각도  $\theta$ 만큼 회전하는 수식이다. 측정 센서의 상, 하 각도에 따른 영상 왜곡 현상을 보정하기 위해 x축을 기준으로 3차원 회전을 적용 하여야 한다.  $p_x, p_y, p_z$ 는 입력되는 점군 데이터이고  $r_x, r_y, r_z$ 는 3차원 회전이 적용된 점군 데이터 정보이다 [5, 6].

$$\begin{bmatrix} r_x \\ r_y \\ r_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (5)$$

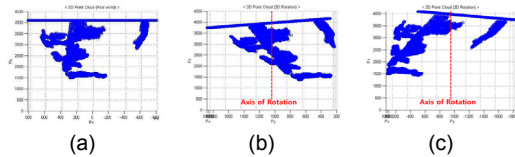


그림 2. 3D Rotation 적용 결과 (a) 원본 3차원 점군 데이터 (b) +15°회전한 3차원 점군 데이터 (c) -15°회전한 3차원 점군 데이터  
Fig. 2 3D Rotation Result (a) Original 3D Point Cloud (b) +15° Rotated 3D Point Cloud (c) -15° Rotated 3D Point Cloud

그림 2는 3차원 회전을 적용한 결과이다. 그림 3의 (a)는 원본 3차원 점군 데이터이며 그림 2의 (b)는 +15°만큼 3차원 회전을 적용한 3차원 점군 데이터, 그림 2의 (c)는 -15°만큼 3차원 회전을 적용한 점군 데이터이다.

### 2.3. 3차원 역 좌표 변환

왜곡 보정이 적용된 데이터는 깊이 영상으로의 출력을 위하여 투영 공간으로 변환을 수행한다. 수식 (6)~(8)은 역 좌표 변환 수식이다. 수식 (6)의  $x'$ 은 깊이 영상의  $x$ 좌표이고, 수식(7)의  $y'$ 는 깊이 영상의  $y$ 좌표, 수식 (8)의  $z'$ 는 깊이 영상의 깊이 값이다[4].

$$x'(x,y) = \left( \frac{p_x(x,y)}{p_z(x,y) \times \tan\left(\frac{f_h}{2}\right)} + 0.5 \right) \times (w-1) \quad (6)$$

$$y'(x,y) = \left( \frac{p_y(x,y)}{p_z(x,y) \times \tan\left(\frac{f_v}{2}\right)} + axis \right) \times (h-1) \quad (7)$$

$$z'(x,y) = p_z(x,y) \quad (8)$$

## III. 시뮬레이션 결과

본 논문에서 제안한 시스템은 Microsoft Kinect 거리 측정 센서로부터 640×480 크기의 깊이 영상을 입력으로 하여 Microsoft Visual Studio 2010과 OpenCV를 이용한 검증 프로그램을 통하여 성능을 검증하였다.

그림 3은 왜곡 보정 알고리즘 적용 결과이다. 그림 3의 (a)에 나타난 입력 깊이 영상을 그림 3의 (b)와 같이 3차원 좌표계에 표현 가능하다. 2.5차원으로 표현된 입력 깊이 영상을 그림 3의 (c)와 같이 3차원 점군 데이터로 변환하여 그림 3의(d)와 같이 3차원 회전을 수행한다. 회전된 3차원 점군 데이터는 그림 3의 (e)와 같이 2.5D 표현으로 변환하여 그림 3의 (f)에 나타난 최종 출력 깊이 영상을 산출한다.

표 1은 깊이 측정 센서와 객체의 각도와 거리에 따른 적용 결과이다. 영상에서의 객체는 컬러 영상에 박스로 표시한 부분이다. #1~#4의 영상은 객체의 정면에서 촬영하여 측정 센서와 객체가 이루는 각도가 0°인 경우와 객체의 위에서 촬영하여 센서와 객체가 이루는 각도가 10°인 경우 제안한 방법의 적용 전과 후의 결과를 나타내었다. #1은 센서와 객체의 거리가 2m이고 #2는 2.5m

#3과 #4는 각각 3m 와 3.5m이다. 측정 센서의 위치가 객체보다 위쪽에 위치하기 때문에 제안한 방법의 적용 결과 영상은 객체의 위치가 적용 전 보다 위쪽에 위치한 것을 확인 가능하다.

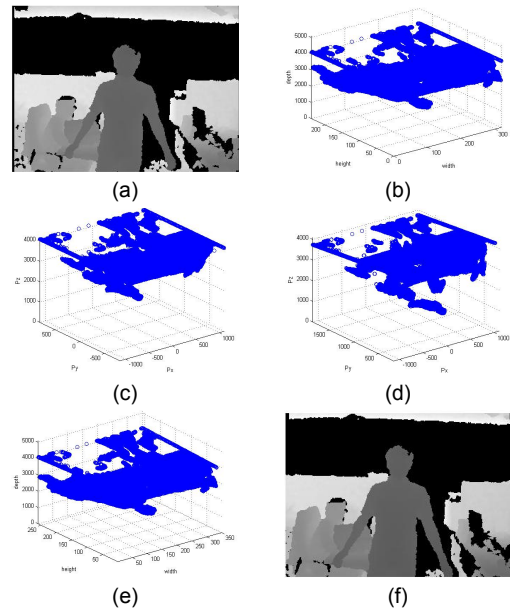


그림 3. 왜곡 보정 알고리즘 적용 결과 (a) 입력 영상 (b) 2.5차원 정보 (c) 3차원 점군 데이터 (d) 회전 된 3차원 점군 데이터 (e) 회전된 2.5차원정보 (f) 출력 영상

Fig. 3 Distortion Correction Algorithm Result (a) Input Image (b) 2.5D Data (c) 3D Point Cloud (d) Rotated 3D Point Cloud (e) Rotated 2.5D Data (f) Output Image

표 1. 깊이 측정 센서와 객체의 각도와 거리에 따른 적용 결과  
Table. 1 Performance of Different Angle and Distance between Depth Sensor and Object

No.	#1	#2	#3	#4
Color Image				
센서 각도 0°				
센서 각도 10° (적용 전)				
센서 각도 10° (적용 후)				

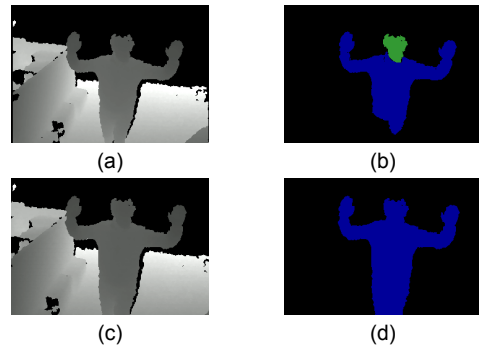
**표 2.** 깊이 측정 센서와 객체의 각도에 따른 적용 결과 비교  
**Table. 2** Performance Comparison of Different Angle between Depth Sensor and Object

No.	객체의 크기 (pixel)			Depth diff.		
	측정 센서 각도 0°	측정 센서 각도 10°		측정 센서 각도 0°	측정 센서 각도 10°	
		제안한 방법 적용 전	제안한 방법 적용 후		제안한 방법 적용 전	제안한 방법 적용 후
#1	122	127	122	12	99	12
#2	81	93	81	9	40	8
#3	52	56	51	9	32	9
#4	32	37	31	6	17	7

표 2에는 표 1에 나타난 영상의 객체의 크기와 객체의 윗부분과 아래 부분의 깊이 값의 차이를 나타내었다. #1의 영상에서 측정 센서의 각도가 0°인 경우 객체의 크기는 영상에서 122 픽셀을 차지한다. 측정 센서가 위쪽에 위치하여 센서의 각도가 10°인 경우 제안한 방법을 적용하기 전에는 127 픽셀이고, 제안한 방법을 적용한 후에는 122 픽셀이다. 깊이 값의 차이는 센서의 각도가 0°인 경우 12이고 센서의 각도가 10°인 경우 제안한 방법을 적용하기 전에는 99, 제안한 방법을 적용한 후에는 12로 정면에서 측정된 값과 동일하다. #2~#4에서도 제안한 방법을 적용 하였을 때 측정 센서가 위쪽에 위치하더라도 정면에서 측정된 크기와 깊이 값으로 보정 가능한 것을 확인 할 수 있다. 따라서 표1과 표2에 나타난 것과 같이 측정 센서의 위치와 각도에 따라 왜곡이 발생하는 깊이 영상에서 제안한 방법을 적용 하였을 때 깊이 영상을 효과적으로 보정 가능하다.

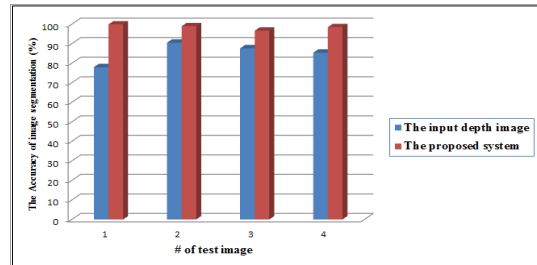
그림 4에는 측정 센서의 위치와 각도에 의하여 왜곡된 영상과 제안한 방법을 적용한 영상의 영상 분할 알고리즘 적용 결과 이다. 그림 4의 (a)는 측정 센서의 위치와 각도에 의하여 왜곡된 입력 영상이고 그림 4의 (b)는 (a)의 깊이 영상 분할 알고리즘 적용 결과이다. 깊이 영상의 영상 분할을 위한 방법은 grassfire기법을 이용하여 영상 내의 깊이 값에 따라 분할하는 방법을 주로 사용한다 [2]. 이는 센서의 위치와 각도에 의하여 촬영하고자 하는 객체의 가장 윗부분과 아래 부분의 깊이 값의 차이가 커지는 경우 다른 객체로 분할하게 된다. 따라서 동작 인식 시스템에 적용되는 경우 인식률이 저하되는 원인이 된다. 그림 4의 (c)는 제안한 방법을 적용한 깊이 영상이고 그림 4의 (d)는 (c)를 깊이 영상 분할 알고리즘에 적용한 결과 영상이다. 왜곡 보정이 효과적으

로 적용 되어 하나의 객체가 동일한 객체로 분할 된 것을 확인 할 수 있다. 그림 5는 영상 분할 알고리즘에 적용 하였을 때 인식률을 나타낸 것이다. 제안된 시스템을 적용 하였을 경우 영상 분할 성공률이 약 11 % 증가 하는 것을 확인 할 수 있다. 그림 4를 통하여 제안한 방법을 이용하여 동작 인식 시스템의 전처리 과정으로 적용 될 경우 인식률 향상이 가능한 것을 알 수 있다.



**그림 4.** 영상 분할 알고리즘 적용 결과 비교 (a) 입력 깊이 영상 (b) (a)의 영상 분할 결과 영상 (c) 제안한 방법 (d) (c)의 영상 분할 결과 영상

**Fig. 4** Performance Comparison of Image Segmentation (a) Input Depth Image (b) Segmentation of (a) (c) The Proposed System (d) Segmentation of (c)



**그림 5.** 영상 분할 알고리즘 인식률 비교

**Fig. 5** Comparison of Image Segmentation Accuracy

#### IV. 하드웨어 구현

그림 6은 제안 한 시스템의 하드웨어 구조이다. 거리 측정 센서의 위치와 각도에 따른 왜곡 보정 시스템은 3 개의 블록으로 나누어진다.

coordinate\_conversion 블록에서는 깊이 정보가 입력으로 인가되면 깊이 영상 크기 정보인 i\_WIDTH와

i\_HEIGHT, FoV의 정보인 i\_H\_FOV와 i\_V\_FOV, 그리고 회전축의 위치를 결정하는 i\_AXIS\_OF\_ROT의 값들을 이용하여 3차원 점군 데이터의 x축, y축, z축 정보들을 계산한다. rotation 블록은 입력되는 i\_DEGREE에 따라 3차원 점군 데이터 정보들에 3D Rotation을 적용하도록 설계하였다. inverse\_coordinate\_conversion 블록은 보정된 3차원 점군 데이터 정보들을 2.5D로 변환하여 출력 깊이 정보를 산출한다. 수식에서 사용되는 Parallel Divider와 Tangent Table, Sine Cosine Table, Inverse Tangent Table을 블록 내부에서 구성하여 연산에 사용하였다.

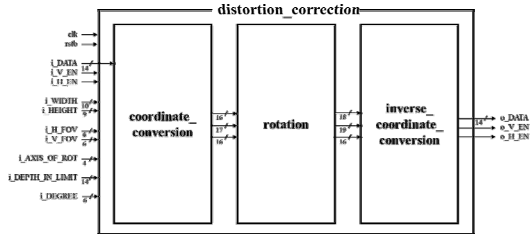


그림 6. 제안한 시스템의 블록 다이어그램  
Fig. 6 Block Diagram of The Proposed System

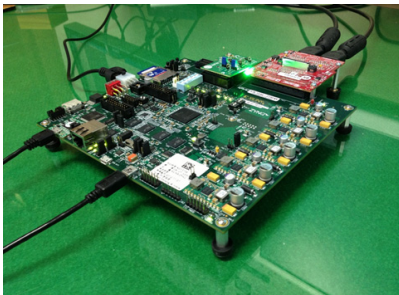


그림 7. 제안한 시스템을 검증하기 위한 FPGA Board  
Fig. 7 FPGA Board for the Proposed System Verification

Verilog-HDL로 구현된 제안한 시스템은 그림 7에 도시된 FPGA와 CPU를 연동하여 확인 가능한 Zynq-7000 FPGA Board를 이용하여 실시간 처리를 검증 하였다. 본 논문에서는 Zynq-7000 All Programmable SoC Series 중 xc7z045ffg900-1 모델을 사용하여 하드웨어 IP를 검증 하였다.

표 3은 Xilinx Design Tool을 합성한 결과이다. Slice Registers는 7,259개를 사용하여 전체 면적의 1%를 차지하였고, Slice LUTs는 6,314개를 사용하여 전체의

2%가 사용되었다. 전체 시스템의 Minimum Period가 6.461ns로 최대 동작주파수가 154.775MHz까지 확보되어 실시간 처리가 가능 한 것을 확인 하였다. 따라서 매우 작은 하드웨어 자원을 사용하면서 빠른 속도의 시스템 구현이 가능한 것을 알 수 있다.

표 3. Xilinx Design Analyzer 합성 결과  
Table. 3 Xilinx Design Analyzer Result

Xilinx Design Analyzer			
Device	Zynq-7000 (xc7z045ffg900-1)		
Slice Logic Utilization	Available	Used	Utilization
Slice Registers (#)	437,200	7,259	1 %
Slice LUTs (#)	218,600	6,314	2 %
RAMB36E1/IFO36E1s	545	0	0 %
RAMB18E1/IFO18E1s	1,090	7	0 %
Minimum period (ns)	6.461		
Maximum Freq. (MHz)	154.775		

\*IDE의 EDA Tool을 제공받아 수행하였음.

## V. 결 론

본 논문에서는 거리 측정 센서를 이용한 동작 인식 시스템의 전처리 과정으로써 실시간 처리와 하드웨어 구현이 가능한 깊이 영상 왜곡 보정 시스템을 제안 하였다. 깊이 영상을 이용한 동작 인식 시스템에서 투영 방식으로 데이터를 입력 받는 거리 측정 센서의 위치와 각도에 따라 촬영된 영상내의 객체의 크기와 깊이 값이 달라짐에 따라 인식률이 낮아지는 문제점이 있다. 본 논문에서는 입력되는 깊이 영상을 3차원 공간으로 표현하여 어파인 변환을 이용한 깊이 영상 왜곡 보정 기능을 적용한다. 제안된 방법을 이용하여 검증한 결과 객체의 크기와 깊이 값을 효과적으로 보정 가능한 것을 확인 하였다. 또한 Verilog-HDL을 사용하여 하드웨어로 설계를 하였고, Zynq-7000 FPGA Board로 합성 시 최대 동작주파수가 154.775MHz로 실시간 처리가 가능한 것을 확인 하였다.

### 감사의 글

본 논문은 동아대학교 학술연구비 지원에 의하여 연구되었음

## REFERENCES

- [1] I. Benbasat, "HCI Research: Future Challenges and Directions," *AIS Transactions on Human-Computer Interaction*, vol. 2, no. 2, pp. 16-21, Jun. 2010.
- [2] Hosang Cho, Kyoungsoon Jang, Jeong-Ho Han, and Bongsoon Kang, "A Background Removal Scheme Based on a Running Average for Motion recognition Using Depth Information," *ITC-CSCC 2013*, pp. 474-476, Jun. 30-Jul. 3, 2013.
- [3] Jihye Jeon, Dahee Kim, Yoongi Yang, Joonki Paik and Changsu Lee, "Image Distortion Compensation for Improved Gait Recognition," *Journal of the Institute of Electronics Engineers of Korea*, vol. 46, no. 4, pp.97-107, July. 2009.
- [4] J. Biswas and M. Veloso, "Depth Camera Based Indoor Mobile Robot Localization and Navigation," *Robotics and Automation (ICRA), IEEE International Conference on*, pp. 1697-1702, May. 2012.
- [5] R. Zhou, J. Wu, Q. He, C. Hu, and Z. Yu, "Approach of Human Face Recognition Based on SIFT Feature Extraction and 3D Rotation Model," *Information and Automation (ICIA), 2011 IEEE International Conference on*, pp. 476-479, Jun. 2011.
- [6] M. A. Garcia and A. Solanas, "3D Simultaneous Localization and Modeling from Stereo Vision," *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, vol. 1, pp. 847-853, May. 2004.



**장경훈(Kyoungsoon Jang)**

2010년 2월 동아대학교 전자공학과 (공학사)  
2010년 3월 ~ 현재 동아대학교 전자공학과 석·박사 통합 과정  
※관심분야 : 영상 신호처리, VLSI architecture design



**조호상(Hosang Cho)**

2010년 2월 동아대학교 전자공학과 (공학사)  
2010년 3월 ~ 현재 동아대학교 전자공학과 석·박사 통합 과정  
※관심분야 : 영상 신호처리, VLSI architecture design



**김근준(Geun-Jun Kim)**

2013년 2월 동아대학교 전자공학과 (공학사)  
2013년 3월 ~ 현재 동아대학교 전자공학과 석·박사 통합 과정  
※관심분야 : 영상 신호처리, VLSI architecture design



**강봉순(Bongsoon Kang)**

1985년 연세대학교 전자공학과 (공학사)  
1987년 미국 University of Pennsylvania 전기공학과 (공학석사)  
1990년 미국 Drexel University 전기 및 컴퓨터 공학과 (공학박사)  
1989년 ~ 1999년 삼성전자 반도체 수석연구원  
1999년 ~ 현재 동아대학교 전자공학과 교수  
2006년 ~ 2011년 멀티미디어 연구센터 소장  
2006년 ~ 2013년 2단계 BK21 사업팀장  
2013년 ~ 현재 BK21 Plus 사업팀장  
※관심분야 : 영상신호처리, SoC설계 및 무선통신