The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 19, No. 6, pp.47-55, Dec. 31, 2019. pISSN 2289-0238, eISSN 2289-0246

https://doi.org/10.7236/JIIBC.2019.19.6.47
JIIBC 2019-6-8

센서 및 카메라 비전을 활용한 OPC UA 기반 협동로봇 가드 시스템의 설계 및 구현

Design and Implementation of OPC UA-based Collaborative Robot Guard System Using Sensor and Camera Vision

김지형[*], 정종필[**]

Jeehyeong Kim[*], Jongpil Jeong[**]

요 약 제조 패러다임 변화에 따라 다양한 협동로봇이 신규시장을 창출하고 있다. 협동로봇은 기존 산업용 로봇 대비 쉬운 운용, 생산성 향상, 단순 작업을 하는 인력을 대체하는 목적으로 모든 산업에서 협동로봇의 수요가 증가하고 있다. 그러나 산업현장에서 협동로봇으로 인한 작업 중 사고가 빈번하게 발생하고 있으며, 작업자의 안전을 위협하고 있다. 인간 중심의 환경에서 로봇을 통한 산업현장이 구성되려면 작업자의 안전을 보장해야 하며 출동 가능성을 업애고 신뢰할 수 있는 통신을 하는 협동로봇 가드 시스템의 개발의 필요성이 있다. 센서 및 컴퓨터 비전을 통해 협동로봇의 작업 반경 내에서 발생하는 사고를 이중으로 방지하고 안전사고 위험을 감소시켜야 한다. 다양한 산업용 장비와 통신을 위한 국제 프로토콜인 OPC UA를 기반으로 시스템을 구축하고 초음파 센서와 CNN(Convolution Neural Network)적용한 영상 분석을 통한 협동로봇 가드 시스템을 제안한다. 제안 된 시스템은 작업자의 불안전한 상황에서 로봇 제어의 가능성을 평가한다.

Abstract The robot is the creation of new markets and various cooperation according to the manufacturing paradigm shift. Cooperative management easy for existing industrial robots, robots work on productivity, manpower to replace the robot in every industry cooperation for the purpose of and demand increases.to exist But the industrial robot at the scene of the cooperation working due to accidents are frequent, threatening the safety of the operator. Of industrial site is configured with a robot in an environment ensuring the safety of the operator to and confidence to communicate that can do the possibility of action.Robot guard system of the need for development cooperation. The robot's cooperation through the sensors and computer vision task within a radius of the double to prevent accidents and accidents should reduce the risk. International protocol for a variety of industrial production equipment and communications opc ua system based on ultrasonic sensors and cnn to (Convolution Neural Network) for video analytics.We suggest the cooperation with the robot guard system. Robots in a proposed system is unsafe situation of workers evaluating the possibility of control.

Key Words : Collaborative Robot Guard System, CNN, OPC UA

[*]정회원, 성균관대학교 스마트팩토리융합학과 석사과정
[**]정회원, 성균관대학교 스마트팩토리융합학과 교수
접수일자: 2019년 7월 2일, 수정완료: 2019년 11월 27일
게재확정일자: 2019년 12월 6일

Received: 2 July, 2019 / Revised: 27 November, 2019 / Accepted: 6 December, 2019
[*]Corresponding Author: ghyeong@skku.edu
Department of Smart Factory Convergence, Sungkyunkwan University, Korea

Ⅰ. 서 론

협동로봇은 산업용 로봇에 안전기능이 강화되어 인간과 같은 공간에서 공동작업이 가능한 제조로봇으로 공정 재배치가 용의하여 생산 유성 증대 효과가 크다. 협동로봇 전문업체들은 제조 패러다임 변화에 따라 다양한 협동 로봇을 출시하면서 신규시장을 창출하고 있으며 글로벌 제조기업들은 다양한 공정에 협동로봇을 적용하여 자동화 영역을 확대하고 있다. 협동로봇은 기존 산업용 로봇 대비 쉬운 운용, 생산성 향상, 단순 작업을 하는 인력을 대체하는 목적으로 자동차, 금속가공, 전기전자, 플라스틱 사출 등 모든 산업에서 협동로봇의 수요가 증가하고 있다. 협동로봇 시장은 유럽에서 태동하여 Industry4.0과 연계하여 시장이 성장하였으며, 이후 북미, 아시아로 시장이 확대되고 2022년에는 전세계적으로 3.6조원, 국내의 경우 1,773억원까지 성장할 것으로 전망되고 있다.

그러나 산업현장에서 협동로봇으로 인한 작업 중 사고가 빈번하게 발생하고 있다. 원인으로는 가드 내 사람을 확인하지 않고 작동, 로봇의 오작동 등으로 다양하며 협동로봇의 안전은 아직 충분히 연구되지 않았기 때문에 연구 및 기술 개발이 진행되어야 한다. 인간 중심의 환경에서 로봇을 통한 산업 현장이 구성되려면 작업자의 안전을 보장해야 하며 충돌 가능성을 없애야 한다. 따라서 센서와 영상을 통한 데이터를 신뢰할 수 있는 통신을 하는 협동로봇 가드시스템을 개발할 필요성이 있다. 센서 및 컴퓨터 비전을 통해 협동로봇의 작업 반경내에서 발생하는 사고를 이중으로 방지하고, 협동로봇의 동작을 실시간으로 감지해 로봇의 오작동으로 인한 안전사고 위험을 감소시켜야 한다[1].

다양한 산업용 장비와의 통신을 위한 국제 표준 프로토콜인 OPC UA를 기반으로 시스템을 구축한다. OPC UA는 다양한 디바이스 및 OS를 통한 유연함과 확장성을 추가, 필드 디바이스에서부터 다양한 상위시스템, 다양한 환경에서 사용이 가능하며 시스템을 확장하기 용이하며 보안기능을 통한 시스템 보호할 수 있다. 초음파 센서를 통해 작업 반경내의 움직임을 확인하고, 카메라 영상을 통해 로봇을 제외한 다른 불완전 요소가 작업 반경에 확인되었을 경우 로봇을 제어한다.

각 센서 및 로봇으로부터 OPC UA 임베디드 서버 모듈 탑재로 TCP/IP를 통해 통신수집서버로 OPC UA를 통해 데이터를 전송하고 모델링한다. Address Space의 장치모델을 편집하여 OPC 서버 프로세스간의 통신을 Serialization한다. 시계열 데이터베이스를 통해 로봇과 센서 데이터[2]를 실시간 저장 및 분석한다. 초음파센서 뿐만 아니라 카메라의 영상 데이터에 CNN(Convolution Neural Network)을 적용하여 영상 분석을 통해 협동로봇 가드 시스템을 제안한다. OPC Server와 로봇, 센서로부터 Read/Write된 시간을 기록한다. Read/Write 사이의 시간 차를 분석해 OPC UA기반의 협동로봇 가드 시스템이 작업자의 불안전한 상황에서 로봇 제어의 가능성을 평가한다.

1절에서는 산업용 로봇 가드 시스템의 필요성과 시스템의 간단한 설명이 이루어진다. 2절에서는 시스템의 관련 기술인 OPC UA, 영상 처리 기술에 대해 알아본다. 3절에서는 시스템 구성에 대해 설명한다. 4절에서는 시스템 구현 및 평가가 진행되며 5절에서 결론은 맺는다.

Ⅱ. 관련연구

1. OPC UA

OPC UA는 데이터 모델을 사용자 정의할 수 있는 정보모델로써 보안 및 플랫폼 독립성, 확장성, 이기종간 호환성, 객체지향 정보모델의 특징을 보유한다. 이러한 특징은 산업용 디바이스의 통신에 유용하며 기타 다른 산업용 프로토콜에는 없는 보안 기능이 많이 부여되어 있어 모든 통신구간을 암호화 통신으로 데이터를 교환한다.

집계 서버는 집계 서버가 소스 서버 그룹의 서비스를 결합하여 그것의 클라이언트에 데이터 서비스를 제공하는 데이터 통합 패턴이다. 일반적으로, 집계 서버는 다양한 종류의 데이터 소스의 데이터를 결합할 수 있다. 그러나 이러한 패턴을 OPC UA에 적용할 때 모든 소스 서버가 OPC UA 서버라고 가정하는 경우가 많지만 현실적이지는 않을 수 있다. OPC UA 서버 응용 프로그램을 집계하는 것은 여러 가지 방법으로 개발될 수 있다. OPC UA의 일반 서버 응용 프로그램의 설계가 제안되었다[3].

OPC UA에서 데이터의 메타 모델은 OPC UA 주소 공간 모델[4]은 객체, 객체 유형, 변수 및 가변 유형과 같은 특정 유형의 노드의 네트워크를 통해 데이터를 나타낸다. 노드는 네트워크의 루트 노드에서 속성이나 경로를 통해 식별할 수 있다. OPC UA 정보 모델은 주소 공간 모델에 따라 특정 애플리케이션 도메인에 대해 개발된 데이터의 표현이다[5]. 이러한 사전 정의된 모델을 집계 서버 애플리케이션에서 사용할 경우, 이를 위해 사전 정의된 집계 작업을 개발할 수 있다.

OPC UA에서 어드레스 공간 변형이 필요할 때 두 가지 다른 상황을 식별할 수 있다. 다른 모델에 따라 표현된 데이터를 OPC UA 주소 공간 모델로 변환해야 할 때 변환의 한 예가 발생한다. 또 다른 경우는 OPC UA 주소 공간이 다른 공간으로 변환되는 것이다. 비록 후 아직 그에 대한 적절한 정의는 보고되지 않았다. 그러나 규칙 기반 변환은 다른 데이터 모델과 함께 적용되었다. XSLT 규칙은 XML로 표현되는 데이터를 변환하는 방법이다. 또한 온톨로지 변환도 제안되었다[6]. 그러나 OPC UA 주소 공간과 XML 또는 OWL의 차이로 인해 OPC UA 주소 공간을 변환하는 규칙은 서로 다를 필요가 있다.

본 논문에서는 OPC UA의 데이터의 전송속도로 성능평가를 다루지만 OPC UA의 성능평가를 다루는 논문은 거의 존재하지 않는다. 예를 들어, OPC UA 규격[7]에 의해 제공되는 보안 메커니즘과 서비스만을 고려하여[8] 성능평가를 실시하며, OPC UA와 이전 규격(COM- 및 XML 기반) 간의 많은 비교를 이용할 수 있다[9], [10]. 이러한 비교는 흥미롭지만 OPC UA의 성과에 대한 유용한 정보는 얻을 수 없다. 또한, OPC UA 통신 스택에 의해 도입된 지연의 실제 측정에 관한 몇 가지 결과가 표시된다[11]. 제시된 결과는 흥미롭지만 실제 시나리오에는 거의 제한되어 있으며 OPC UA 성능에 대한 일반적인 고려사항을 지적할 수 없다.

2. CNN(Convolution Neural Network)

CNN은 이미지에서 객체, 얼굴, 장면을 인식하기 위해 패턴을 찾는데 유용하다. CNN은 특징을 직접 학습하고 수동으로 추출할 필요가 없으며, 가장 높은 수준의 인식 결과를 보인다. 또한, 새로운 인식 작업을 위해 CNN을 재학습하여 사용하는 것이 가능하다. 자율 주행 자동차, 얼굴 인식 애플리케이션[12]과 같이 객체 인식과 컴퓨터 비전이 필요한 분야에서 CNN을 많이 사용한다.

Wu[13]는 무패딩 전에 3D 메쉬를 24×24×24 크기의 이진 점유 복셀로 변환하여 3D ShapeNets를 제안했다. 그 후, 3D 도형의 분포를 배우기 위해 5단계의 경합적 깊은 신념망(입력층 제외)을 구축하였다.

Maturana와 Scherer[14]는 빠르고 정확한 3D 객체 인식을 위해 기본 3D CNN 아키텍처(VoxNet)를 도입했다. 이는 훈련 매개변수가 적은 ModelNet 데이터 세트에서 3D ShapeNet보다 더 나은 인식 성능을 달성한다.

Sedaghat 연구진[15]은 ORION Basic, ORION Fusion 및 ORION F2C를 포함한 세 가지 다른 구조를 통해 VoxNet에 방향 예측 작업을 도입했다. 그 결과 ORION은 ModelNet10(92% ~ 93.8%), Sydney Urban Objects(0.72 ~ 0.778) 데이터셋에서 VoxNet을 능가한다. 방향성이 3D 객체 인식에 중요한 정보임을 입증한다.

Hegde와 Zadeh[16]는 두 개의 볼륨 CNN과 AlexNet 기반의 멀티뷰 CNN을 이용하여 비교적 큰 3D CNN인 FusionNet(약 1억 8천 8백만 개의 매개변수를 가진)을 제안했다. 이 세 개의 서브 네트워크는 성능 향상을 위해 다중 데이터 표현을 결합하기 위해 마지막 출력 계층 이후에 융합되었다.

Qi[17]는 체적 CNN의 성능에 영향을 미칠 수 있는 두 가지 요인, 즉 네트워크 아키텍처와 볼륨 해상도를 논의한 다음 두 개의 체적 CNN을 제안했다. 첫 번째 볼루메트릭 CNN에서는 네트워크(NIN)[18]에서 네트워크의 mlpcon 구조를 기본 빌딩 블록으로 3차원 케이스로 확장했다. 과적 방지를 위해 부량감독의 보조교육도 실시되었다. 두 번째 네트워크의 경우 비등방성 프로빙 커널을 사용하여 3D CNN을 2D 도메인으로 변환한 다음 NIN 아키텍처를 사용하여 객체 분류를 수행했다.

Brock[19]은 개시자 망 모듈, 확률적 깊이 및 대량의 데이터 증가를 사용하여 오버핏하지 않도록 45단 깊이 3D 볼륨 CNN의 앙상블을 제시했다. VRN 앙상블 네트워크는 ModelNet10 및 ModelNet40 데이터셋 모두에서 최신 성능을 실현한다.

Ⅲ. 협동로봇 가드 시스템 구성

1. 시스템 구조

제안된 산업용 로봇 가드 시스템은 그림 1과 같은 구조를 가진다. OPC Edge Server에 Main Server가 존재하며 노드 정보를 바탕으로 로봇, 센서, 카메라를 제어 및 모니터링 한다. OPC Client는 OPC Server와 통신하며, OPC UA Modeler는 OPC Server와 Modeler사이는 NodeSetXML로 인터페이스 된다. Linux Controller는 OPC Client를 바탕으로 동작하고 Historian Database인 MachBase에 로그성 데이터가 실시간으로 저장된다. 카메라의 영상에서 추출된 이미지는 CNN 모델을 통해 분석되고 OPC Server로 업로드 된다.

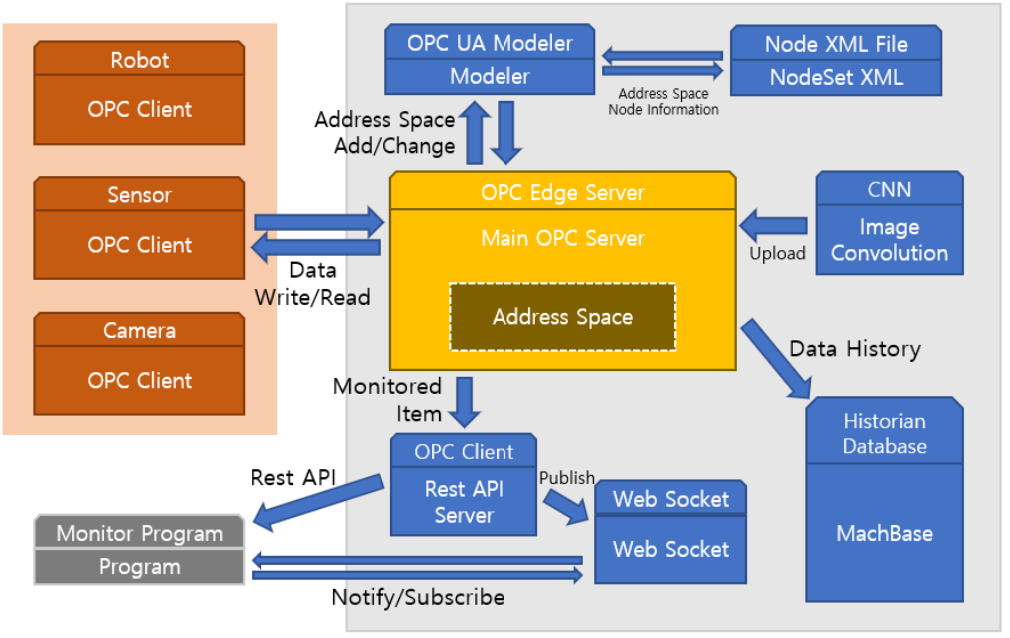

그림 1. 시스템 구조
Fig. 1. System architecture

OPC Server : Main Server에 존재하는 Address Space는 Controller (OPC Client) 중 로봇과 센서, 카메라의 노드 정보를 가지고 있다. 사용자는 OPC Modeler를 통해 로봇과 센서, 카메라의 노드를 모델링 하여 실시간으로 OPC Server에 업로드할 수 있다. 노드 정보를 갖고 있는 OPC Server는 OPC Client의 Variable Node나 OPC Method를 통하여 로봇, 센서, 카메라를 모니터링 한다.

OPC Client : OPC Server의 Address Space의 특정 노드(Monitored Item)를 감시하고 OPC Client는 받아온 데이터를 Web Socket에 Publish하고 개발된 프로그램의 초기값 설정시 Rest API를 통하여 데이터를 전달한다.

Address Space: OPC UA 자체 Node와 구분하기 위해 별도 Namespace를 추가하였으며 Device타입을 구성하였다. 정확한 NodeSetXML을 얻기 위해

UnifiedAutomation UaModeler를 사용하였으며 후속작업으로 일괄 편집된다. OPC Server와 Modeler 사이는 NodeSetXML으로 인터페이스 된다.

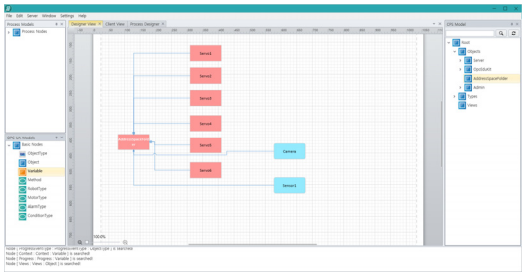

그림 2. Modeler를 통한 AddressSpace 구성
Fig. 2. Modeler addressspace through the configuration

Linux Controller: Linux Controller는 OPC Client를 base로 동작하며 OPC Main Server의 Address Space에 정의되어있는 각 Variable Node의 데이터를 Monitoring 하다 데이터 변경에 의해 Notify가 발생하면 로봇이나 모터를 제어한다.

OPC UA Modeler: OPC Server와 Modeler사이는 NodeSetXML로 인터페이스된다. Address Space의 장치모델을 편집하는 역할을 하는 modeler와 OPC Server 프로세스간의 통신을 위한 Serialization을 위해 XML을 선택하였다.

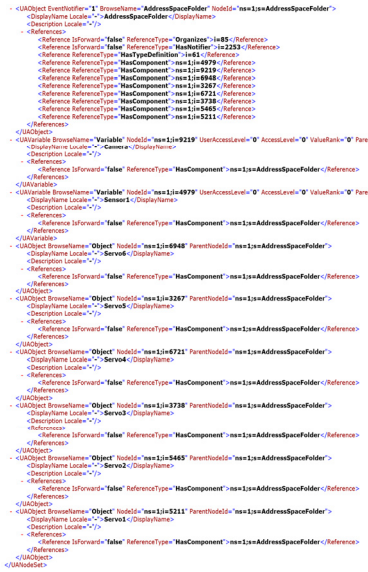

그림 3. NodeSetXML 구성
Fig. 3. NodeSetXML Configuration

Web Socket: Web Socket은 SignalR을 사용하고 OPC UA의 MonitoredItem에서 데이터를 통지 받아 브라우저가 인식할 수 있는 HTTP프로토콜로 변환한다.

Historian Database : 속도를 고려하여 가장 빠르다고 판단되는 MachBase를 사용하였다. Machbase는 로그성 머신 데이터를 실시간으로 저장, 분석이 가능한 실시간 시계열 데이터베이스이다.

2. 영상분석

CNN은 이미지 인식에 좋은 성능을 보이고 가장 널리 사용되는 알고리즘이다. 사람의 시각 피질 메커니즘에 영감을 받아 설계된 네트워크이다. FCNN도 사람 신경망을 착안하여 만들어진 모델이지만 이미지 인식에 있어 모든 픽셀마다 weight가 다려서 분석하지 않고 위치에 관계없이 시각적 특징을 추출하기 위하여 제안되었으며, 지역적 연결(local connection)과 파라미터 공유(shared weight)을 고려하여 학습한다. 따라서 FCNN으론 몇 백개의 층으로 할 수 있는 일을 단 몇 층 만으로 효과적으로 feature extraction한 결과를 뒷단의 FCNN층으로 전달하여 분류작업을 수행한다. 연산속도가 획기적으로 감소하며 local minimum에 빠지거나 vanishing gradient의 위험이 준다. Image 외에도 많은 분야에서 CNN 기반 모델 활용한다.

CNN 모델의 구성요소는 크게 4가지(Convolutional layer(Convolution, Activation, Pooling, FCNN Layer)로 이루어져 있다.

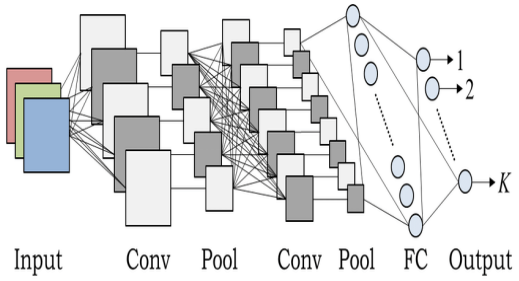

그림 4. CNN 구성
Fig. 4. CNN Configuration

Convolution: convolution층에서는 각 filter가 입력 이미지의 픽셀 전체를 차례로 훑고 지나가며 linear combination을 진행하고 Feature Map을 구성한다.

Activation: activation은 단순히 convolution으로 얻은 feature map을 relu등을 사용해 activate시킨 것이다.

Pooling: pooling층에서는 단순히 featuremap의 dimention을 subsampling 하여 차원을 축소시키는 방법이다. 중요한 데이터는 유지하면서 입력의 크기를 줄임으로서 과적합을 예방하고 계산량을 감소시키는 효과를 가진다. FCNN의 dropdown과 같은 효과을 가지며 subsampling 방법으로는 주로 2x2 max pooling을 사용한다.

FCNN: convolutional layer을 통해 추출한 featuremap을 1d vector로 쭉 나열하여 일반적인 FCNN의 input으로 주어 분류작업을 수행한다.

OpenCV API를 통해 로봇을 Cropping하여 데이터를 확보하였다. 모델 구성은 2개의 convolution layer, 1개의 fully connected layer로 구성하고 후미에 1 fcnn layer로 hidden layer들을 구성하였다. 첫번째 convnet는 8x8 filter 16개, 두번째 convnet은 5x5 filter 36개을 사용하였다. fcnn에서는 128개의 relu node들을 사용하였고 마지막에는 4개의 class을 구분하기위해 softmax로 4개의 output을 가지게끔 하였다. feedforward 하기 전에 한가지 전처리를 하였는데 빠른 converge을 위해 input data들을 255로 나누어 0~1사이의 값으로 normalization하였다. 학습된 데이터는 OPC Server로 업로드 된다.

Ⅳ. 구현 및 평가

1. 시스템 구성

본 시스템의 Server와 Client는 Cubieboard2를 사용하였고, 사양은 ARM A7 Dual Core CPU, ARM Mali400MP2 GPU, 1GB DDR3 Memory이다. 로봇은 6채널 서보 컨트롤러 사용하고 TTL 직렬 통신을 지원한다. 영상 촬영은 Logitec C615 PC 캠을 사용하였고, 사양은 최고 해상도 1920X1080로 촬영이 가능하며 최대 800만 화소 촬영을 지원한다. 거리측정 센서로는 HC-SR04를 사용하였고, 사양은 8.4kHZ 사용, 측정거리는 2cm~4.5m이다. 소프트웨어로 Linux와 C# ASP.NET Core를 사용하였으며 사용된 패키지는 표1과 같다.

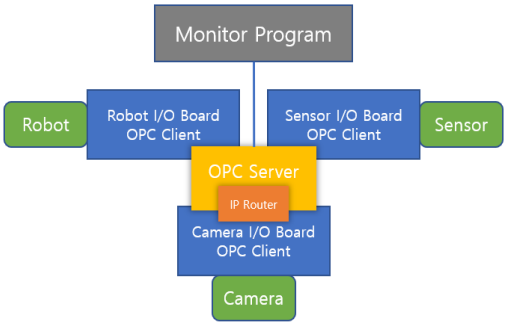

그림 5. 하드웨어 블록다이어그램
Fig. 5. Hardware Blockdiagram

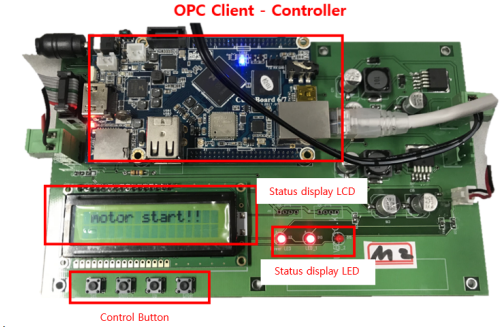

그림 6. OPC Client-Controller
Fig. 6. OPC Client-Controller

표 1. 구현에 사용된 패키지 리스트
Table 1. List of packages used for implementation

Microsoft.AspNetCore.Server.Kestrel
Microsoft.AspNetCore.Server.Kestrel.Https
Newtonsoft.Json
Portable.BouncyCastle
Antlr.3.4.1.9004
FontAwesome.4.7.0
jQuery.1.6.4
jQuery.3.3.1
jQuery.Form.4.2.2
jQuery.UI.1.12.1.1
log4net.2.0.8
Microsoft.AspNet.Cors.5.0.0
Microsoft.AspNet.Mvc.5.2.4
Microsoft.AspNet.Mvc.ko.5.2.4
Microsoft.AspNet.Razor.3.2.4
Microsoft.AspNet.Razor.ko.3.2.4
Microsoft.AspNet.SignalR.2.4.0
Microsoft.AspNet.SignalR.Client.2.4.0
Microsoft.AspNet.SignalR.Core.2.4.0
Microsoft.AspNet.SignalR.JS.2.4.0
Microsoft.AspNet.SignalR.SystemWeb.2.4.0
Microsoft.AspNet.Web.Optimization.1.1.3
Microsoft.AspNet.WebPages.3.2.4
Microsoft.AspNet.WebPages.ko.3.2.4
Microsoft.Owin.4.0.0
Microsoft.Owin.Cors.4.0.0
Microsoft.Owin.Host.HttpListener.4.0.0
Microsoft.Owin.Host.SystemWeb.2.1.0
Microsoft.Owin.Hosting.4.0.0
Microsoft.Owin.Security.2.1.0
Microsoft.Web.Infrastructure.1.0.0.0
MSTest.TestAdapter.1.3.2
MSTest.TestFramework.1.3.2
Newtonsoft.Json.12.0.1
Newtonsoft.Json.6.0.4
NLog.4.5.11
Owin.1.0
WebGrease.1.6.0

OPC UA의 Address Spcae는 Node로 구성되며 Node의 종류로는 Object와 Variable 그리고 Method로 세분화된다. OPC Node는 NodeState로 표현되며, NodeState는 크게 Instance계열과 Type계열로 나뉜다. 서버내부에서는 Node의 List로 PrederfinedNodes 객체로 관리한다.

2. 제어 구조

가드 시스템에서 Robot 및 센서를 제어하는 Control 박스는 총 3개로 구성하였으며 모두 OPC Clinet로 동작한다. Controller는 OPEN62541 SDK OPC Client를 베이스로 동작한다. 최초 구동시 OPC Main Server로부터 모델링 정보인 Address Space정보를 읽어온 후 각 Control마다 가지고 있는 Node를 Monitored Item으

로 등록한다. 등록된 Monitored Item은 Server에서 값이 변경되거나, 머신러닝 모델 쪽 OPC Client에서 데이터 변경을 하면 Controller에 데이터 변경이 통지되고 로봇이 제어된다.

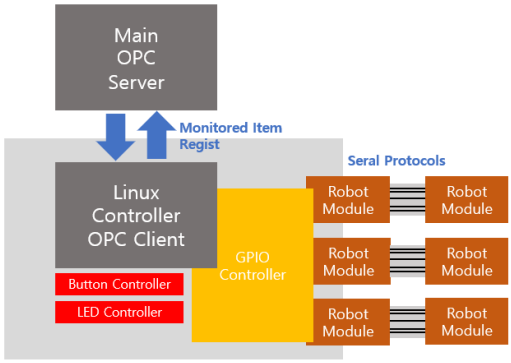

그림 7. 로봇, 센서, 카메라 컨트롤 구조
Fig. 7. Robot, sensor, camera control structure

3. 실험 및 평가

산업 애플리케이션에서 매우 일반적인 다른 종류의 데이터 교환은 클라이언트가 주기적으로 업데이트된 변수의 값에 접근할 때 발생한다. 예를 들어 샘플링 프로세스에 의해 달성된 변수의 값과 Client를 말한다. 샘플링에 따라 이러한 값을 읽을 필요가 있다. 클라이언트 애플리케이션은 기한 내에 생산된 각 값을 받아야 한다. 이 경우, 각 값이 산출된 순간과 Client는 값을 받는 순간 사이의 지연은 성능 평가에 중요하다. 사실, 각 변수에 대한 평균 지연은 데이터 교환의 효율을 측정하기에 적합하다. 기한 내에 주기적으로 생산되는 각 정보를 Clinet에게 전달할 수 있는 통신 시스템의 평가하는 것은 매우 중요하다. 이 시나리오에 관련된 정보는 그림 8을 통해 확인할 수 있다.

시스템의 성능평가는 초음파센서와 카메라로부터 이상이 감지되었을 때 로봇이 반응하는 시간을 평가하였다. 그림 9는 데모로 제작된 시스템을 보여준다. 초음파센서로부터 30cm 이내에 인간이 감지되었을 경우 로봇을 중지하고 카메라는 CNN을 통해 이미지를 분석하고 작업반경내에 로봇을 제외한 다른 물체가 감지되었거나 로봇이 프로세스와 다르게 비정상적으로 움직일 경우 로봇을 중지한다.

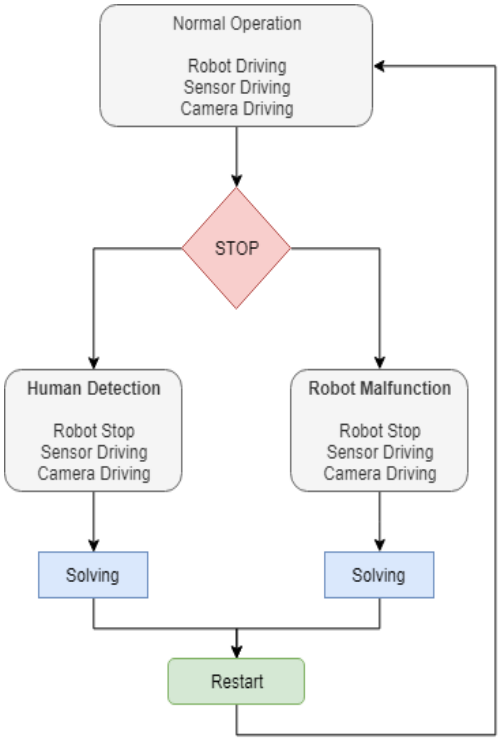

그림 8. 평가 시나리오
Fig. 8. Evaluation Scenario

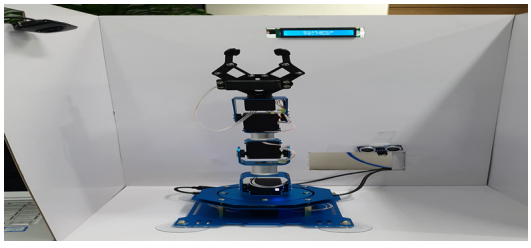

그림 9. 시스템 데모
Fig. 9. System Demo

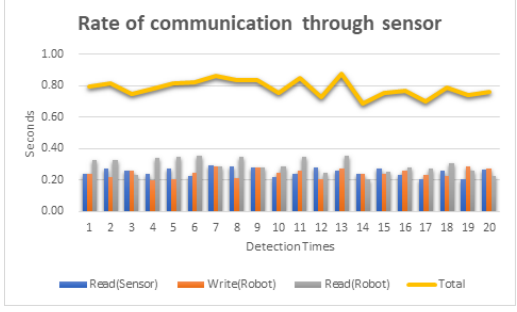

그림 10. 센서를 통한 통신 속도
Fig. 10. Rate of communication through sensor

Read/Write 데이터 교환의 시간의 항목을 지연과 관련하여 수행하고 OPC Server가 생성하고 Client에게 전달되는 각 정보 사이의 지연을 평가한다.

20번의 시나리오로 각각에 대해 10Mbps에서 OPC UA Client와 Server간의 데이터 교환을 진행하였다. OPC UA 보안 및 전송 프로토콜이 Read/Write 서비스를 기반으로하는 Client/Server 데이터 교환에 미치는 영향에 중점을 두었다. Sensor로부터 Server로 데이터가 Read된 시간, Server로 Robot 데이터가 Read된 시간, Server로 Robot으로 데이터가 Write된 시간을 측정하였으며, 측정 결과 1000ms 이내의 성능 보여주었다.

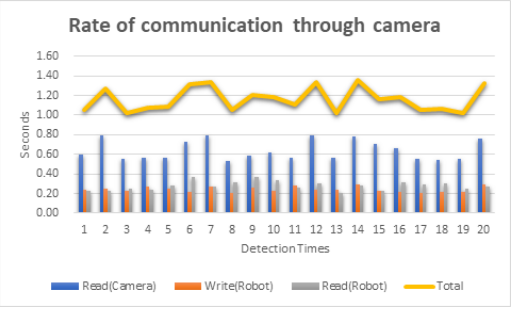

그림 11. 카메라를 통한 통신 속도
Fig. 11. Rate of communication through camera

OPC UA 보안 및 전송 프로토콜이 Read/Write 서비스를 기반으로하는 Client/Server 데이터 교환에 미치는 영향에 중점을 두었다. Sensor로부터 Server로 데이터가 Read된 시간, Server로 Robot 데이터가 Read된 시간, Server로 Robot으로 데이터가 Write된 시간을 측정하였으며, 측정 결과 1500ms 이내의 성능 보여주었다.

V. 결 론

본 논문은 협업로봇 가드 시스템의 문제점을 인식하고 초음파센서와 머신러닝을 활용하여 OPC UA 기반의 새로운 협업로봇 가드 시스템을 제안했다. OPC UA기반의 시스템은 데이터의 수집 및 통합에 적절하였으며 시스템을 확장하게 용이하다. 또한, 초음파 센서와 카메라 비전을 활용하여 거리와 이미지 분석에 따른 데이터를 통해 로봇을 제어할 수 있었다. 제안된 시스템은 센서를 통해 1000ms이내로 로봇을 제어했고 영상 분석을 통해 1500ms이내로 제어했다. 센서를 통한 평가는 실제 산업 현장에 적용할 수 있다는 가능성을 보여줬다.

초음파 센서에 비해 영상 분석을 통한 로봇제어는 성능이 부족함을 보여줬다. 향후 연구에서는 영상 분석을 통한 로봇의 제어 속도를 향상할 수 있는 분석 모델이 필요하며 영상을 통한 로봇의 이상 감지뿐만 아니라 로봇의 고장감지를 위한 분석 모델도 필요하다.

References

[1] Seungmin Oh, Kwansoo Jung, "A Reliable Protocol for Real-time Monitoring in Industrial Wireless Sensor Networks," Journal of Korea Institute of Information, Electronics, and Communication Technology vol. 10, no. 5, pp. 424-434, 2017.
DOI: http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE07254903

[2] Dong-Jin Shin, Ji-Hun Park, Ju-Ho Kim, Kwang-Jin Kwak, Jeong-Min Park, Jeong-Joon Kim," The Journal of The Institute of Internet, Broadcasting and Communication (IIBC),. vol. 19, no. 1, pp. 117-126, 2019.
DOI: https://doi.org/10.7236/JIIBC.2019.19.1.117

[3] B D. Grossmann, M. Bregulla, S. Banerjee, D. Schulz, R. Brown, "Big Data-based Sensor Data Processing and Analysis for IoT Environment," Emerging Technology and Factory Automation (ETFA), 2014.
DOI: https://doi.org/10.1109/etfa.2014.7005354

[4] "OPC UA specification Part 3: Address space model," OPC UA Specification Release 1.02, 2012.

[5] "OPC UA for devices companion specification," OPC UA Companion Specification Release 1.01, 2013.

[6] N. Choi, I-Y. Song, H. Han, "A survey on ontology mapping," ACM SIGMOD Record, vol. 35, no. 3, pp. 34-41, 2006.
DOI: https://doi.org/10.1016/j.phpro.2012.02.273

[7] A. Braune, S. Henning, S. Hegler, "Evaluation of OPC UA secure communication in web browser applications," Proceedings IEEE International Conference on Industrial Informatics (INDIN), pp. 1660-1665, 2008
DOI: https://doi.org/10.1109/indin.2008.4618370

[8] O. Post, J. Deppala, H. Koivisto, "The performance of OPC UA Security Model at field device level," Proceedings ICINCO, pp. 337-341, 2009.
DOI: https://doi.org/10.5220/0002249103370341

[9] W. Mahnke, S. Leitner, M. Damm, "OPC Unified Architecture," Springer, 2009.
DOI: ISBN 978-3-540-68898-3

[10] BJ. Lange, F. Iwanitz, T.J. Burke, "OPC from Data Access to Unified Architecture," OPC Foundation, Softing 2010.

[11] Intel, "Reducing product development effort for OPC unified architecture," Iavailable at.

DOI: http://download.intel.com/platforms/applied/indpc/OPC_Unified_Arch_CS.pdf

[12] Z. Wu, S. Song, A. Khosla, F. Yu, "3D ShapeNets: a deep representation for volumetric shapes," IEEE conference on computer vision and pattern recognition, pp. 1919-1920, 2015.
DOI: http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE07578395

[13] Seung Ho Lee, "A Method for Determining Face Recognition Suitability of Face Image," Journal of Korea Academy Industrial Cooperation Society, vol. 19, no. 11, pp. 295-302, 2018.
DOI: https://doi.org/10.1109/cvpr.2015.7298801

[14] D. Maturana, S. Scherer, "VoxNet: a 3D convolutional neural network for real-time object recognition" IEEE conference on computer vision and pattern recognition (CVPR), pp. 1919-1920, 2015.
DOI: https://doi.org/10.1109/cvpr.2015.7298801

[15] Nima Sedaghat, Mohammadreza Zolfaghari, Ehsan Amiri, Thomas Brox, "Orientation-boosted Voxel Nets for 3D Object Recognition," Procedings of the British Machine Vision Conference, 2017.
DOI: https://doi.org/10.5244/c.31.97

[16] Hegde V., Zadeh R. Fusionnet, "3D object classification using multiple data representations," arXiv preprint, 2016.

[17] C.R. Qi, H. Su, M. Nießner, A. Dai, M. Yan, L. Guibas, "Volumetric and multi-view CNNs for object classification on 3D data," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
DOI: https://doi.org/10.1109/cvpr.2016.609

[18] ZLin M., Chen Q., Yan S, "Network in network," arXiv preprint, 2013.

[19] ZBrock A., Lim T., Ritchie J., Weston N, "Generative and discriminative voxel modeling with convolutional neural networks," arXiv preprint, 2016.

저 자 소 개

김 지 형(정회원)

- 2018년 9월 ~ : 성균관대학교 스마트팩토리융합학과 석사과정 재학중
- 주관심분야 : OPC UA, CPS, 분산컴퓨팅 아키텍처, 스마트팩토리

정 종 필(정회원)

- 2008년 ~ 2009년 : 성균관대학교 컨버전스연구소 연구교수
- 2010년 ~ 현재 : 성균관대학교 정보통신대학 겸 산학협력단 교수
- 2015년 ~ 현재 : 전자부품연구원 IoT 융합연구센터 전문연구위원
- 2016년 ~ 현재 : 성균관대학교 스마트팩토리융합학과 사업총괄책임자
- 주관심분야 : 스마트팩토리, 모바일융합컴퓨팅, 센서 네트워크, 차량 모바일 네트워크, 네트워크 보안, IT융합, 인터렉션 사이언스, 스마트 헬스케어, IoT/M2M, 웨어러블 컴퓨팅 등

※ 이 논문은 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터지원사업의 연구결과로 수행되었음 (IITP-2019-2018-0-01417).