논문 2020-2-14 http://dx.doi.org/10.29056/jsav.2020.12.14

실내 전력관리 시스템을 위한 환경데이터 인터페이스 설계

김도현*, 이규대**†

Monitoring System for Optimized Power Management with Indoor Sensor

Do-Hyeun Kim*, Kyu-Tae Lee**†

요 약

인공지능의 활용성이 다양해지면서 소형 휴대용기기에 알고리즘을 탑재하려는 요구가 증가하고 있다. 또한 임베디드 시스템이 고성능화하면서 운영체제는 물론 고속연산 및 머신러닝의 알고리즘 구현이 가능해 지고 있다. 그러나 반복연산과 방대한 학습데이터를 처리하는 머신러닝알고리즘의 특성으로 네트워크 연결에 의한 클라우드 환경에 의존하고 있다. 임베디드 시스템에서의 독자적인 운영을 위해서는 저 전력화 및 최적화 알고리즘에 의한 빠른 실행이 요구된다. 본 연구에서는 스마트 제어를 목적으로 임베디드 시스템에 에너지 측정용 센서를 연결하고, 실시간 측정 및 모니터링 시스템으로 측정정보를 데이터베이스로 저장하는 장치를 구현하였다. 연속적으로 측정되어 저장된 데이터는 학습 알고리즘에 적용하여, 최적화 전력제어에 활용가능하며, 에너지 측정에 요구되는 다양한 센서의 인터페이스가 가능한 시스템을 구성하였다.

Abstract

As the usages of artificial intelligence is increased, the demand to algorithms for small portable devices increases. Also as the embedded system becomes high-performance, it is possible to implement algorithms for high-speed computation and machine learning as well as operating systems. As the machine learning algorithms process repetitive calculations, it depend on the cloud environment by network connection. For an stand alone system, low power consumption and fast execution by optimized algorithm are required. In this study, for the purpose of smart control, an energy measurement sensor is connected to an embedded system, and a real-time monitoring system is implemented to store measurement information as a database. Continuously measured and stored data is applied to a learning algorithm, which can be utilized for optimal power control, and a system interfacing various sensors required for energy measurement was constructed.

한글키워드 : 임베디드 시스템, 전력제어, 머신러닝, 데이터베이스, 센서인터페이스

keywords : embedded system, power control, machine learning, database, sensor interface

* 국립 제주대학교 컴퓨터공학과
** 국립 공주대학교 정보통신공학부
† 교신저자: 이규대(email: ktlee@kongju.ac.kr)
접수일자: 2020.12.10. 심사완료: 2020.12.14.
게재확정: 2020.12.21.

1. 서 론

임베디드 플랫폼의 저가격화 및 고성능화가 진행되면서, 라즈베리파이와 같은 싱글보드 컴퓨

터는 운영체제가 포함된 독자적인 PC의 성능에 가까워지고 있다. 또한 현재 많은 분야에서 관심을 갖는 인공지능에 의한 판단 알고리즘이 네트워크를 통한 분산 처리의 효과를 보이고 있고, 단순 비교나 분석결과 보다 성능이 우수한 머신러닝의 도입이 진행되고 있다[1]. 이러한 클라우드 환경에서의 인공지능 처리는 네트워크 상황이 불안할 경우, 오동작의 가능성이 있고, 네트워크가 불가능한 환경에서도 동작이 가능한 독립형 머신러닝의 처리가 요구된다[2].

그러나 제한된 메모리와 처리능력을 갖는 임베디드 플랫폼은 네트워크 환경이 없을 경우, 완전한 처리결과를 얻기 힘들다. 본 연구에서는 독립적인 환경에서 인공지능 알고리즘의 운영이 가능한 최적화 알고리즘을 개발을 목표로 하고 있으며, 이를 위한 선행 작업으로 주변 환경의 정보를 수집하는 센서 인터페이스 및 데이터 로깅 시스템 구현에 대해 다루었다. 계속적으로 수집 및 축적된 데이터의 분석을 진행하면서 최적화 알고리즘 개발을 진행할 예정이며, 연구결과 다양한 센서의 인터페이스 구현 및 데이터베이스 구축을 통한 데이터로깅 구현과정을 확인하였다.

2. 환경정보 센싱

인공지능에 의한 정보처리를 위해서는 분석 판단 알고리즘 개발도 중요하지만, 필요로 하는 외부데이터의 입력이 있어야 한다. 임베디드 시스템에서의 머신러닝처리를 위해서는 하드웨어적으로 보드에 각종 센서의 연결 및 신호체계가 인터페이스 환경에 맞게 설치되어야 한다[3][4]. 또한 측정된 데이터들은 시간적으로 데이터베이스에 축적되도록 하여, 인공지능 프로그램이 데이터 분석 및 분리, 예측으로 사용할 수 있게 된다. 알고리즘 개발에는 인터넷에 공지된 데이터베이스를 가져와서 분석을 진행하게 되지만, 독립적인 환경에서의 머신러닝 알고리즘을 구현하기 위해서는 주변 환경의 정보를 실시간을 측정하는 센싱 인터페이스가 구축되어야 한다.

센싱과 데이터의 축적환경이 구축되면, 그림 1과 같이 주변 상황인식과 분석을 통해 제어 출력과 연동하여, 최적 에너지 관리가 가능한 시스템에 적용이 된다.

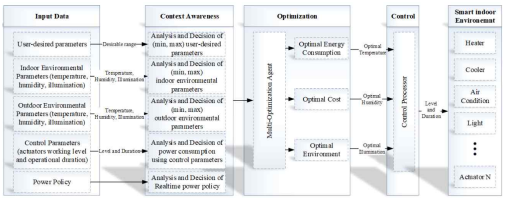


그림 1. 임베디드 상황인식과 최적 제어구성도
Fig. 1. Optimized control Flow in environment

최적화 알고리즘은 주변상황을 인지하고, 에너지의 효율적 사용과 비용감소를 위해 저 전력화 목적으로 작성하게 되며, 인공지능 알고리즘의 개발에 집중연구가 가능할 수 있다.

인공지능 시스템(ANN:Artificial Neural Network)으로 결과를 얻기 위한 방법으로서의 머신러닝(Machine Learning)은 입력데이터의 범위가 제한적인 상황인식 모델 구현에 유용한 방법으로 그림 2와 같은 예측 모델링 프로세스를 갖는다[5][6].

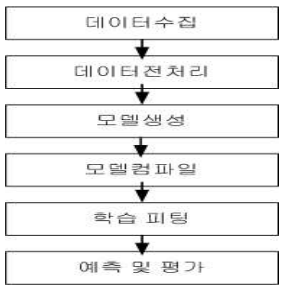


그림 2. 센서데이터 학습 및 추론
Fig. 2. Learning and evaluation

1) 데이터 수집

데이터수집(Data Collection)단계는 예측 시스템에서 필요한 외부정보를 가져오는 단계로서 DNN(Deep Neural Network)의 경우는 빅데이터(Big Data)를 저장하고 있는 데이터베이스를 활용하여, 예측과 평가에 필요한 데이터를 수집한다. 본 연구와 같이 데이터가 제한적인 경우는 시스템에서 각종 센서를 활용한 주변 환경의 신호정보를 수집하는 인터페이스 장치를 통해 수집된다.

2) 데이터 전처리

수집된 데이터는 데이터전처리(Data Pre-processing) 과정으로 예측프로세스에 적합한 형태로 가공된다. 데이터정제(Data Cleaning), 또는 수집된 데이터에서 특징성분을 추출하는 피처링(Featuring) 과정도 포함된다.

3) 모델생성

수집된 데이터의 분석알고리즘을 선택하는 과정으로 모델선택(Model Selection)과정이며, ReLU, sigmode, softmax 등이 활용된다.

4) 모델컴파일

생성된 예측모델이 실행이 가능하도록 계층별 구조를 완성한다.

5) 학습피팅

학습과정에서의 평가예측 정확도를 얻기 위한 반복횟수를 정하는 과정으로, 분석 데이터의 특성에 따라 조정된다.

6) 예측 및 평가

머신러닝모델의 성능을 평가하고, 새로 입력된 데이터에 대한 예측을 하는 과정으로, 정확한 결과를 얻기 위해, 학습용 데이터와 평가용 데이터를 분리하여, 실험에 적용한다.

이상과 같이 예측모델 시스템이 적용되기 위한 기본정보는 주변 환경의 데이터를 수집하는 것으로, 본 연구에서는 임베디드시스템에 센서를 연결(Interface)하고, 입력된 정보를 데이터베이스에 저장하는 데이터 로거를 구축하기 위한 입력 데이터 신호처리 과정에 대해 기술한다.

3. 센서 인터페이스

최적화 전력시스템은 단순한 온오프 기능과 달리, 통계적이고 인간 친화적인 적정 환경을 유지하면서 전력의 효율적 관리를 목적으로 한다. 이를 위해서는 주변 환경의 상태를 모니터링 하는 장치가 필요하며, 모니터링 정보를 수집하는 각종 센서의 설치 및 계속적인 측정 감지 장치가 요구된다. 이러한 센서의 인터페이스는 임베디드 시스템의 GPIO(General Perpose Input Output) 포트와 물리적인 연결 및 신호체계를 구성하여야 한다[7][8]. 또한 적정한 상태를 유지하기 위해서는 냉방기, 난방기, 공기제어기 등의 제어신호출력(actuator)이 연결되어 그림 3과 같이 전체적인 구성도가 작성된다. 실내 환경 상태를 나타내는 주요정보는 온도, 습도, 조도의 측정이 되며, 실내의 위치와 높이에 따라 개별적으로 수집되어야 한다. 시간적으로 연속해서 수집되는 정보는 데이터베이스에 축적되고, 머신러닝 프로그램으로 최적화 상태를 판단하여, 출력제어기에 온오프 명령으로 실내 환경을 변화하도록 한다.

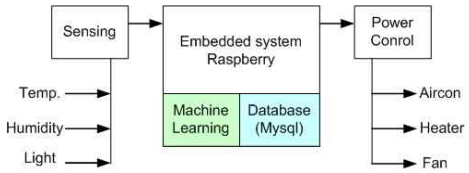


그림 3. 최적화 제어시스템 구성도
Fig. 3. Optimized control system architecture

변화된 실내 환경의 결과는 센서로 입력되는 피드백정보로 연속적인 관리가 가능하도록 구성된다. 주변 환경을 측정 감지하기 위한 센서는 온도, 습도, 조도, 전력소모량 등 실내환경에 쾌적한 상황을 조성하면서, 소요되는 에너지를 측정하는 시스템으로 구성된다. 사용된 임베디드 시스템은 라즈베리파이4 모델이며, Raspbian 운영체제 환경에서 센서의 연결과 통신체계에 필요한 알고리즘은 python, 과 C 언어로 작성되었다.

3.1 온습도측정 센서

온도정보는 실내 환경에서 측정되는 정보를 대상으로 하기 때문에 -50도에서 150도의 범위에서 측정이 가능한 온도센서 DHT11와 LM35, DS18s20이 실험에 사용되었다[9][10]. 온도 값이 아날로그로 출력되는 센서의 활용성을 고려하여, 아날로그 값을 디지털 값으로 변환하여 주는 ADC(Analog to Digital convertor)를 적용하였다. 센서회로와 임베디드 시스템과의 신호 인터페이스 및 측정된 결과는 그림 4와 같이 구성하였다. 시각적 효과를 고려하여, 실시간 측정값이 출력되도록 구성하여, 두 지점의 온도 값과 측정시간이 표시되도록 구현하였다.

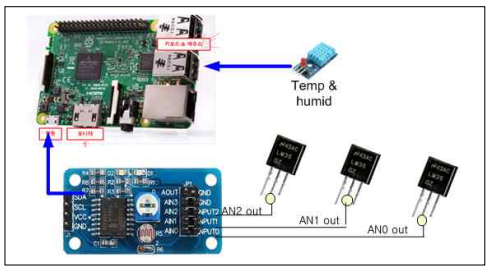


그림 4. 온습도측정 회로
Fig. 4. Temp sensor interface

3.2 전력측정 센서

실내 환경의 적정온도 제어에 소모되는 전력을 측정하기 위해서는 전류 값을 측정하여, 수식으로 환산하는 방법을 적용한다[11]. 그림 5는 전류 센서와 연결된 회로를 구성하고, 측정된 결과를 나타낸다. 전력 값은 전류와 전압의 곱으로 계산되는 것이므로, 센서의 측정값으로 전류와 전압을 얻어 다음 식(1)에 의해 계산되고, 그림 5의 우측과 같이 측정 및 계산 결과가 표시된다.

$$P = I \times V \quad \text{----------(식1)}$$

전류센서는 ACS712를 사용하여, 측정된 아날로그 전류 값을 PCF8591ADC로 디지털화 하여, 시스템으로 전달하는 회로구성을 하였다. 전류센서는 30A까지 측정이 가능하고, ADC는 출력비트 수가 많을수록 정밀한 값을 얻을 수 있다. 시험에 사용된 ADC는 8비트로 동작하고 있으며, 추후 12비트 변환이 가능한 ADS1015 ADC를 사용할 수 있도록 구성하였다.

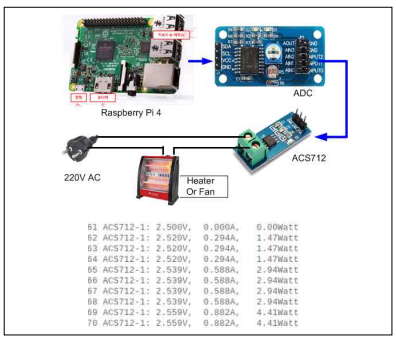


그림 5. 전력측정 회로 및 데이터
Fig. 5. Power sensor and measurement

습도정보는 온도센서와 일체형의 모델을 적용하여, 동시 측정이 가능하도록 하였으며, 예측에 필요한 실외의 날씨 정보와, 계절 정보 등은 예측 알고리즘에서 적용하도록 하였다.

추가적으로 조도센서를 연결하여 조도제어와 등기구 정보를 활용한 전력제어도 시스템에 적용이 가능하다.

4. 데이터 로거 구현

센서 데이터의 입력이 데이터베이스에 연속적으로 저장되도록 동작하는 과정을 데이터로거(Data Logger)라고 한다. 본 연구에서 센서 데이터의 입력은 온도센서와 전류센서로 정하였고, 직렬입력의 온도 센서와 ADC(Analog to Digital Convertor)와 함께 사용되는 LM35를 병행하여 그림 6과 같이 구현하였다.

측정 및 저장된 데이터는 서버에 저장되어 네트워크 구성으로 시간적인 표현이 가능하며, 여기서는 grafana 디스플레이 도구를 사용하여 표현하였다[12]. 출력결과는 그림 6의 화면에서 나타내었다. 시간 순차적으로 측정되는 데이터는 데이터베이스에 축적되도록 하였으며, 측정 시간정보와 측정값이 동시에 저장되도록 하였다. 데이터베이스는 서버로 동작하여, 네트워크를 통한 웹연동이 가능한 오픈소스로 구현되었다.

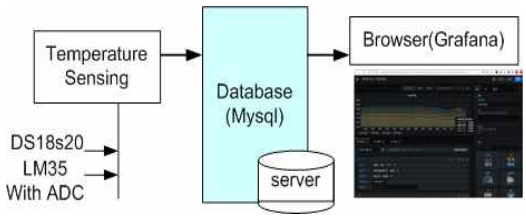


그림 6. 데이터 로거 구성도
Fig. 6. Date Logger architecture

측정된 데이터는 그림 7과 같이 레코드 필드를 구성하고, DHT11 센서 인터페이스로 입력된, 온도와 습도데이터가 mysql dB, Influx dB에 연동되도록 구현하였다. 측정된 온습도 데이터는 측정 시간과 함께 저장되어, 지정된 시간간격으로 연속적인 저장이 되도록 하였다. 그림은 10초단위로 측정된 시간정보와 데이터이며, 데이터필드의 첫 번째 칼럼이 기록시점에서의 시간을 나타내고 있다.

온도측정과 데이터 저장시스템의 구성으로, 축적된 데이터의 시간적, 계절적 변화에 대한 학습 알고리즘을 적용하면, 최적화 알고리즘개발에 효과적으로 활용이 가능할 것으로 판단된다.

```
MariaDB [raspi_db]> select * from collect_data ;
+--------+---------------------+--------+--------+
| sensor | collect_time        | value1 | value2 |
+--------+---------------------+--------+--------+
| DHT11  | 2020-08-12 09:54:04 |     28 |   75.8 |
| DHT11  | 2020-08-12 09:54:14 |     28 |   75.8 |
| DHT11  | 2020-08-12 09:55:59 |     28 |   75.8 |
| DHT11  | 2020-08-12 09:56:09 |     28 |   75.8 |
| DHT11  | 2020-08-12 09:56:19 |     28 |   75.8 |
| DHT11  | 2020-08-12 09:56:29 |     28 |   75.8 |
| DHT11  | 2020-08-12 09:57:17 |     28 |   75.8 |
```

그림 7. 데이터베이스 구조
Fig. 7. database fields

센서 인터페이스로 측정된 각종 센싱 데이터는 최적 값을 분석하기 위해 시간적으로 누적된 값으로 데이터베이스로 저장해야 한다. 그림 8은 mysql 로 저장된 데이터베이스에서 네트워크로 연결된 가상서버와 연결하여, 시간정보와 측정값을 읽어서 센싱된 온도와 습도 값을 나타내는 디스플레이 화면을 보인다. 센싱 결과로 축적된 값은 주변상황을 유지하기 위해 소모된 에너지와 시간별 계절별 변화 상황을 온도, 습도, 조도, 전력 등의 데이터 값으로 표현이 가능하도록 추가될 수 있다.

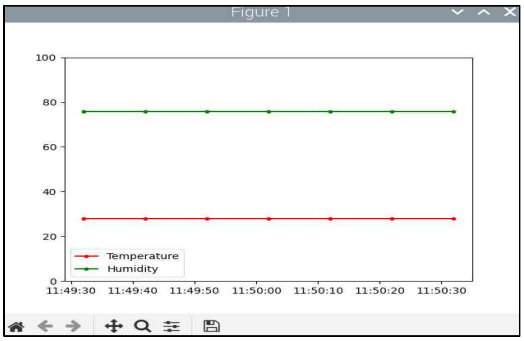


그림 8. 데이터베이스의 측정 데이터
Fig. 8. Database runtime display

5. 결 론

실내 환경은 외부의 날씨 상태와 주야간의 변화에 따라 영향을 받는 공간으로, 적정한 온도와 습도를 유지하기 위해서는 다양한 센서 데이터의 축적 및 히터와 냉방기의 효율적인 관리 및 운영으로 에너지의 최적관리가 가능해질 수 있다. 실내 에너지의 스마트 제어를 위해 임베디드 시스템에 에너지 측정용 센서를 연결하고, 실시간 측정 및 모니터링 시스템으로 측정정보를 데이터베이스로 저장하는 장치를 구현 하였으며, 온도 센서와 습도 센서의 동작상태 및 데이터베이스로 저장하는 기능을 확인하였다. 연속적으로 측정되어 저장된 데이터는 학습 알고리즘에 적용하여, 최적화 전력제어에 활용가능하며, 에너지 측정에 요구되는 다양한 센서의 인터페이스가 가능하도록 시스템을 구성하였다.

임베디드 시스템은 라즈베리보드로 구성하였으나, 인공지능 알고리즘의 실시간 처리를 위해 NVIDIA 개발보드를 병행하여 진행하고 있다. 온도 센서와 습도 센서의 동작상태 및 데이터베이스로 저장하는 기능을 확인하였으며, 조도 센서와 외부 기상정보를 연동하는 방법으로 실시간 제어 알고리즘을 구현할 예정이며, 연구결과는 스마트제어방법의 최적화 구현으로 인공지능 알고리즘을 적용한 에너지 효율화에 적용이 가능할 것으로 기대된다.

감사의 글

이 논문은 2020년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 연구임.(2020R1I1A3070744)

참 고 문 헌

[1] Ali, Safdar, and DoHyeun Kim. “Building power control and comfort management using genetic programming and fuzzy logic”, Journal of Energy in Southern Africa 26.2 (2015): 94-102. http://www.scielo.org.za/scielo.php?script=sci_arttext&pid=S1021-447X2015000200010&lng=en&nrm=iso

[2] Ullah, Israr, Rashid Ahmad, and DoHyeun Kim. “A Prediction Mechanism of Energy Consumption in Residential Buildings Using Hidden Markov Model”, Energies 11.2 (2018): 358. https://doi.org/10.3390/en11020358

[3] 이규대, “임베디드시스템의 이진코드 추출 및 분석”, 한국소프트웨어감정평가학회 논문지, 5권1호, pp27-38, 2009.5 http://www.i3.or.kr/2009spring.pdf

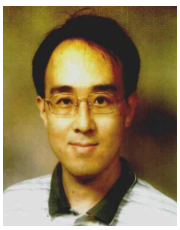
[4] Do-Hyeun Kim, KyuTae Lee, “Management of Reliability and Delivery for Software Object Material”, Journal of Software Assessment and Valuation(JSAV), vol.15, No.2, pp.51-57. Dec. 2019. DOI : https://doi.org/10.29056/jsav.2019.12.07

[5] Kyu-Tae Lee, Hyun-Chang Lee, Jang-Geun Ki. “Establishment of the Subtitle on Materials for Evaluating Intellectual Ownership“. (IJSIP)International Journal of Signal Processing, Image Processing and Pattern Recognition. vol.10, No.9, pp.79-88. Sep., 2017. https://doi.org/10.14257/IJSIP.2017.10.9.08

[6] M. M. Swift, B. N. Bershad, and H. M. Levy. “Improving the Reliability of Commodity Operating Systems“. ACM Trans. on Computer Systems. vol.23, No.1, pp.77-110, Sep., 2003. https://doi.org/10.1145/945445.945466

[7] M. M. Swift, M. Annamalai, B. N. Bershad, and H. M. Levy. “Recovering Device Drivers”, ACM Trans. on Computer Systems.vol.24, No.4, pp.333-360 Nov., 2006. http://u.cs.biu.ac.il/~wiseman/2os/bugs/swift1.pdf
[8] M. Rajagopalan, M. A. Hiltunen, T. Jim, and R. D. Schlichting. “System Call Monitoring Using Authenticated System Calls”, IEEE Trans. on Dependable and Secure Computing, pp.216-229. July 2006. DOI: https://doi.org/10.1109/TDSC.2006.41
[9] T. Naughton, W. Bland, G. Vallee, C. Engelmann, and S. L. Scott. “Fault Injection Framework for system Resilience Evaluation”, Proc. of the Resilience“09. pp.23-28. June, 2009. https://www.christian-engelmann.info/publications/naughton09fault.pdf
[10] scootersoftware company, Oct. 10, 2018. http://www.scootersoftware.com/features.php?zz=features_focused
[11] V.J. Mooney, D.M. Blough. “A hardware-software real-time operating system framework for SoCs”, IEEE Design & Test of Computers. vol.19, No.6, pp.44-51, Nov., 2002. DOI: https://doi.org/10.1109/MDT.2002.1047743
[12] Do-Hyeun Kim, KyuTae Lee, “Management of Reliability and Delivery for Software Object Material”, Journal of Software Assessment and Valuation(JSAV), vol.15, No.2, pp.51-57. Dec. 2019. http://dx.doi.org/10.29056/jsav.2019.12.07

저 자 소 개



김도현(Do-Hyeun Kim)

2000.8 경북대 전자공학과(정보통신전공) 박사
2004.9～현재 국립 제주대학교 공과대학 컴퓨터공학전공 교수.
<주관심분야> 사물인터넷, 예측 및 최적제어, 모바일 컴퓨팅, 임베디드 소프트웨어

이규대(Kyu-Tae Lee)

1991 고려대 전자공학과 박사
1992～국립공주대학교 정보통신공학부 교수
<주관심분야> 신호처리, VLC, 저작권보호, 임베디드 시스템, 상황인식 및 학습