
조기 경보를 위한 화재 판단 알고리즘을 이용한 무선 센서네트워크 기반 화재 감시 응용 시스템 설계 및 구현

Development of WSN(Wireless Sensor Network)-based Fire Monitoring Application System using Fire Detection Algorithm for Early Warning

김아름*, 조경진*, 장재우*, 심춘보**
전북대학교 컴퓨터공학과*, 순천대학교 정보통신공학부**

Ah-Reum Kim(arkim@chonbuk.ac.kr)*, Kyoung-Jin Jo(kj@chonbuk.ac.kr)*,
Jae-Woo Chang(jwchang@chonbuk.ac.kr)*, Chun-Bo Sim(cbsim@sunchon.ac.kr)**

요약

최근 문화재나 시설물 관리를 위한 화재 감시 응용 시스템에 대한 요구가 증대되고 있다. 이러한 화재 감시 시스템은 화재 상황을 대처할 수 있도록 지원함으로써 피해규모를 축소시킬 수 있다. 그러나 기존 시스템은 일정 주기로 화재 감시를 수행함으로써 화재 판단을 지연시키는 단점이 존재한다. 또한 감시 상황을 확인 할 수 있는 사용자 인터페이스를 제공하지 않는다. 따라서 이러한 두 가지 문제점을 해결하기 위하여 첫째, 조기 위험상황 경보를 위해 새로운 화재 판단 알고리즘(Early Fire Detection Algorithm)을 제안한다. 이는 데이터 분포를 기반으로 하여 화재 판단 시작 주기를 동적으로 설정하기 때문에, 화재 판단 시간 측면에서 기존 알고리즘 보다 우수하다. 둘째, 제안하는 화재 판단 알고리즘을 통하여 사용자 인터페이스를 제공하는 화재 감시 응용 시스템을 개발한다. 마지막으로 성능 실험을 통해, 개발된 시스템이 다양한 화재 상황에서 조기 위험상황 경보를 위해 활용될 수 있음을 보인다.

■ 중심어 : | 화재 감시 응용 시스템 | 조기 화재 경보 | 센서 네트워크 |

Abstract

Recently, fire monitoring application systems have been an active research area due to the safety of industries, historical monuments and so on. The fire monitoring application systems can reduce the damage of properties by providing earlier warning for possible fire situation. However, the existing systems have a drawback that they detect fire with delay due to their uniform epoch in fire detection algorithm. Moreover, they do not provide user-friendly graphical user interfaces in their fire monitoring systems. To resolve the problems, First, we propose a new fire detection algorithm (Early Fire Detection Fire Algorithm) which uses the distribution of sensing data for early fire detection. Our fire detection algorithm is better in terms of fire detection time than the existing work because it can set a start time of fire detection epoch dynamically based on data distribution. Second, we develop a fire monitoring application system which provides users with both a user-friendly graphical user interface and a fire alarm message when fire occurs. Finally, we show from our experiment that our developed system is effectively used for early fire warning in a variety of fire situations.

■ keyword : | Fire Monitoring Application System | Early Fire Warning | Wireless Sensor Network |

* 본 연구는 교육과학기술부와 한국산업기술재단의 지역혁신인력양성사업으로 수행된 연구결과임

* 본 연구는 지식경제부의 대학 IT연구센터 지원사업의 연구 결과로 수행되었음 (IITA-2009-(C1090-0902-0047))

접수번호 : #090720-001

심사완료일 : 2009년 10월 05일

접수일자 : 2009년 07월 20일

교신저자 : 장재우, e-mail : jwchang@chonbuk.ac.kr

I. 서론

무선 센서 네트워크(Wireless Sensor Network) 기술은 센서 노드의 배치가 용이하고, 시스템 유지 및 관리 비용이 적게 들기 때문에 다양한 응용 서비스에서 활용될 수 있다[1]. 따라서 무선 센서네트워크 응용 시스템은 시설물 관리, 기상 관측, 환경/농업, 헬스케어 등 서비스 영역이 점차 확산되고 있는 추세이다[2]. 특히, 최근 발생한 대구 지하철 화재사건과 승례문 화재 사건 등을 통하여 문화재나 공공시설, 저장 창고 등의 시설물 관리에 있어서 화재 감시 응용 시스템에 대한 요구가 증대되고 있다[3-5]. 이러한 화재 감시 응용 시스템은 화재 상황을 대처할 수 있도록 지원함으로써 피해 규모를 축소시킬 수 있다. 이를 위해 Hongli Xu et al. 연구[5]와 Yanjun Li et al. 연구[6]가 제안되었다. 그러나 Hongli Xu et al. 연구는 온도 데이터에 대해 설정된 상한선만을 기준으로 화재를 판단하며, Yanjun Li et al. 연구는 일정 주기로 화재 감시를 수행함으로써 화재 판단을 지연시키는 단점이 있다. 즉, 조기 경보에 대한 성능을 일정하게 보장하지 못하고 특히, 사용자 부재 시 위험 상황 경보를 수행 하지 않아서 신속한 화재 상황 대처를 지원하지 못한다.

따라서 이러한 문제점을 해결하기 위해 본 논문에서는 첫째, 조기 위험상황 경보를 위해 데이터 분포를 기반으로 한 새로운 화재 판단 알고리즘을 제안한다. 둘째, 제안하는 화재 판단 알고리즘을 이용하여, 사용자에게 위험 상황 경보 및 편리한 사용자 인터페이스를 제공하는 화재 감시 응용 시스템을 개발한다. 이를 통해, 사용자 부재 시 휴대전화로 화재 상황 발생 메시지(SMS)를 전송하거나 원격지의 사용자를 위해 인터넷을 통한 클라이언트 시스템을 제공하여 신속한 대처를 지원할 수 있다. 본 논문의 구성은 다음과 같다. 먼저 2장에서 기존 화재 감시 응용 시스템에 대한 관련 연구를 분석한다. 3장에서는 기존 응용 시스템의 문제점을 해결하기 위해 조기 위험상황 경보를 위한 화재 판단 알고리즘을 제시한다. 4장에서는 화재 판단 알고리즘을 이용하여 개발하는 화재 감시 응용시스템에 대해 기술한다. 5장에서는 성능 실험을 통해 개발된 시스템이 조

기 화재상황 경보를 위해 유용하게 활용될 수 있음을 입증한다. 마지막으로 6장에서 결론 및 향후 연구를 제시한다.

II. 관련 연구

문화재나 시설물 관리를 위한 무선 센서네트워크 기반의 화재 감시 응용 시스템에 대한 연구가 증가하고 있다. 첫째, 버클리 대학의 Firebug 프로젝트[3]는 무선 센서네트워크 환경의 화재 감시 응용 시스템에 대한 초기 연구를 수행하였다. 이 연구에서는 화재 상황을 인위적으로 발생시켜 데이터를 수집함으로써 시스템 프로토타입을 정의하였지만, 화재 판단을 위한 알고리즘은 제시하지 않았다. 둘째, Hongli Xu et al. 연구[5]는 고대 건축물을 위한 화재 감시 시스템 및 사용자 인터페이스를 제시하였다. 화재 판단을 위해서는, 센서노드로부터 수집한 데이터를 이용하여 임계값(threshold)을 초과한 온도가 수집되었을 때, 이를 화재 발생 상황으로 판단하는 간단한 규칙을 사용하였다. 따라서 임계값이 높게 설정된 경우 화재 발생 시점으로부터 다소 시간이 경과한 후 임계값에 도달하기 때문에 화재 판단 시간이 지연된다. 반면에 임계값이 낮게 설정된 경우 화재가 발생하지 않은 상태를 화재 상황으로 오판하는 문제점이 존재한다. 셋째, Yanjun Li et al. 연구[6]는 이러한 문제점을 해결하기 위해, 일정 주기마다 화재 판단을 수행하는 알고리즘을 제안했다. 이 연구에서는 수집된 온도가 45℃ 이상일 때와 짧은 주기 동안 온도가 5℃ 이상 상승할 때를 화재 발생 상황으로 판단하였다. 마지막으로, C.-L. Fok et al. 연구[7]는 질의를 이용한 화재 감시 시스템을 제시하였다. 예를 들어, “60초마다 온도가 60℃ 이상이거나 온도가 10℃ 이상 증가할 때의 센서노드 식별번호와 온도를 구하라”의 질의가 주어지면, 일정 주기마다 온도가 충분히 높거나 갑자기 증가할 때를 화재 상황으로 판단한다. 한편, Yanjun Li et al. 연구와 Mueller et al. 연구는 데이터 분포를 고려하지 않고 시스템에서 정의한 일정 주기마다 온도 변화량을 기준으로 화재 판단을 수행하기 때문에 화재 판단 시간

이 지연되는 단점이 있다.

III. 조기 화재 판단 알고리즘

화재 감시 응용 시스템은 화재 발생 초기에 위험 상황을 조기 탐지하고 사용자에게 알려 신속히 위험에 대처할 수 있도록 지원해야 한다.

Hongli Xu et al. 연구는 온도의 상한선만을 기준으로 화재 여부를 판단하기 때문에[5] 화재 판단 시간이 지연되거나, 화재가 발생하지 않은 경우를 화재 상황으로 오판하는 문제점이 있다. 즉, 온도 상한선이 높게 설정된 경우 화재 발생 시점으로부터 다소 시간이 경과한 후 상한선에 도달하기 때문에 화재 판단 시간이 지연되거나, 온도 상한선이 낮게 설정된 경우 화재가 발생하지 않은 경우를 화재 상황으로 오판하는 문제점이 있다.

이를 해결하기 위한 Yanjun Li et al. 연구(이하 Li 연구라 칭함)는 시스템에서 정의된 일정 주기마다 화재 판단을 수행한다[6]. 그러나 화재 판단 주기가 온도 상승 곡선에서 어느 위치에 나타나는지에 따라 화재판단 시간이 달라지기 때문에 항상 최적의 화재 판단 시간을 제공하지 못하는 문제점이 존재한다. 예를 들어 [그림 1]과 같이 화재 감시 주기가 시작되는 시점에 따라서 화재 판단 시간이 달라진다. 화재 판단 주기 p초 동안 온도변화량 c를 측정한다고 가정할 때, 화재 감시 주기가 시작되는 시점에 따라 화재 판단 시점의 온도가 A 또는 B가 될 수 있다. A지점은 화재 발생 직후 시점이고, B지점은 화재가 발생한 후 시간이 경과하여 A온도인 시점에 비해 큰 피해가 발생한 시점이다. [그림 1]에서 A지점의 시간은 190초이고, B지점의 시간은 250초이다. B지점은 A지점보다 화재 판단 시간이 60초 지연됨을 나타낸다. 소방방재청 자료에 따르면, 화재가 발생한 후 5분 이상 경과하면 화재의 연소 확산 속도와 피해 면적이 급격히 증가하기 때문에[8] 신속한 대처를 지원하기 위한 연구가 필요하다.

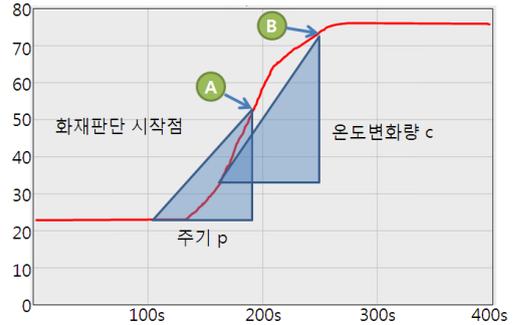


그림 1. Li 연구의 문제점

따라서 이를 해결하기 위하여 조기 화재판단 알고리즘(Early Fire Detection Algorithm, EFDA)을 제안한다. 이는 데이터 분포를 기반으로 하여 화재 판단 시작 주기를 동적으로 설정하기 때문에 기존 알고리즘에 비하여 최적의 화재 판단 시간을 보장한다. 즉, 실시간으로 온도 변화를 확인하여 화재 가능성이 있는 변화일 경우 온도가 변화하기 시작한 시점을 화재 판단 주기의 시작지점([그림 2]의 S)으로 설정하고 화재 판단 주기 p동안의 온도변화를 측정하여 기준 값을 초과했을 경우를 화재 상황으로 판단한다([그림 2]의 E). 습도 데이터의 경우도 동일한 알고리즘에 의해 화재 판단을 수행한다. 한편, 화재가 발생하여 온도가 상승한 상태로 일정하게 유지되는 부분은 온도변화량으로 화재 여부를 확인 할 수 없으므로, 임계값 이상인 경우에도 화재 상황으로 판단한다[그림 2].



그림 2. 조기 화재판단 알고리즘(EFDA)

```

1. Function EarlyFireDetection() {
2.   warning ← false // 화재 판단 주기의 시작 지점의 설정 여부
3.   fireTemp1    // warning 플래그를 위한 온도 변화량
4.   fireHumid1   // warning 플래그를 위한 습도 변화량
5.   fireTemp2    // 화재 판단을 위한 온도 변화량
6.   fireHumid2   // 화재 판단을 위한 습도 변화량
7.   absTemp     // 항상 화재로 판단하는 온도
8.   absHumid    // 항상 화재로 판단하는 습도

9.   while new tuple
10.    if current temperature >= absTemp and current humidity <= absHumid
11.      alarm fire
12.    if warning is false
13.      A ← (current temperature) - (last tuple's temperature)
14.      B ← (last tuple's humidity) - (current humidity)
15.      if A > fireTemp1 and B > fireHumid1
16.        warning ← true
17.        lastTemp ← last tuple's temperature
18.        lastHumid ← last tuple's humidity
19.        break
20.      else warning ← false

21.    else //warning is true
22.      A ← (current temperature) - (last tuple's temperature)
23.      B ← (last tuple's humidity) - (current humidity)
24.
25.      if A > fireTemp2 and B > fireHumid2
26.        alarm fire
27.      else warning ← false
28.  }
    
```

그림 3. 조기 경보를 위한 화재 판단 알고리즘(EFDA)

조기 위험상황 경보를 위한 화재 판단 알고리즘은 [그림 3]과 같다. 첫째, 새로운 데이터가 수집 될 때마다 화재 판단 알고리즘을 수행한다(9줄). 둘째, 임계값을 초과하는 데이터가 수집되면 사용자에게 화재 경보를 알린다(10~11줄). 셋째, 임계값을 초과하지 않을 경우, 화재 가능성이 있는 변화인지 검사하여 유효한 경우 화재 감시 주기의 시작지점으로 설정하고 위험구간으로 플래그를 설정한다(12~19줄). 마지막으로, 화재 감시 주기의 시작지점이 설정되어 있을 때 온도와 습도 변화량을 기준으로 화재 발생 여부를 판단한다(21~27줄).

IV. 화재 감시 응용 시스템

본 절에서는 3장에서 제안한 화재 판단 알고리즘을

이용한 서버-클라이언트 구조의 무선 센서네트워크 기반 화재 감시 응용 시스템을 설계한다.

1. 화재 감시 응용 시스템 구조

개발하는 화재 감시 응용 시스템은 질의처리 모듈, 데이터 관리 모듈, 상황인식 모듈 그리고 GUI 인터페이스 모듈로 구성된다[그림 4]. 첫째, 질의처리 모듈은 질의를 기계어로 번역하여 게이트웨이(gateway) 역할을 하는 센서노드(싱크노드)로 전송한다. 둘째, 데이터 모듈은 데이터를 유효한 형태로 가공하여 저장 및 관리한다. 셋째, 상황인식 모듈은 수집된 데이터의 화재 여부를 판단하기 위해, 조기 경보를 위한 화재 판단 알고리즘을 이용한다. 넷째, GUI 인터페이스는 데이터 관리 모듈과 상황 인식 모듈로부터 서비스를 제공받아 사용자에게 제공한다.

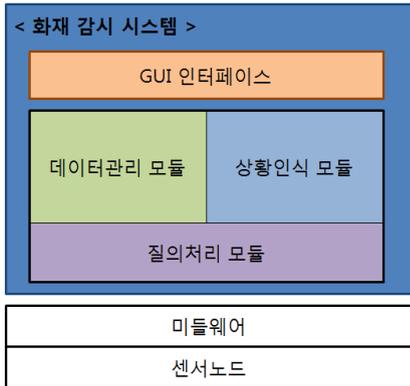


그림 4. 화재 감시 응용 시스템 구조

화재 감시 응용 시스템의 각 모듈에 대하여, 구체적인 질의를 통한 수행 절차를 설명한다. 첫째, 서버에서 "각 센서에서 5초마다 노드의 식별자(node ID), 온도(temperature), 습도(humidity)를 수집하라."와 같은 질의가 바이트코드 형태로 주어진다. 질의처리 모듈은 자바 어셈블러 파서를 이용하여 각 명령어를 기계어 형태로 변환하여 게이트웨이(싱크노드)로 전송한다. 둘째, 게이트웨이는 네트워크상에 구성된 라우팅 경로를 바탕으로 모든 센서노드에게 질의를 전송한다. 그에 따라 센서노드는 수집한 노드 아이디와 온도, 습도 데이터를 5초마다 게이트웨이를 통해 서버로 전송한다. 이때, 센서노드로부터 수집된 데이터는 가공되지 않은 값이므로, 데이터모듈을 통해 변환식을 이용하여 온도, 습도 값을 얻는다. 이렇게 변환된 데이터를 GUI 인터페이스 모듈에 제공한다. 셋째, 데이터 모듈에서는 사용자 요청에 따라 로그파일이나 데이터베이스 형태로 저장하여 관리한다. 마지막으로, 상황 인식 모듈은 조기 경보를 위해 제안하는 화재 판단 알고리즘을 바탕으로 화재 감시를 수행한다. 실시간으로 온도나 습도 변화를 검사하다가 화재 가능성이 있는 변화량이 감지되면 위험 구간으로 판단하여 화재 판단 주기의 시작지점으로 선정하고 화재 조건을 검사한다. 온도와 습도 변화량이 화재 조건을 만족할 경우, GUI 인터페이스를 통해 사용자에게 경고 메시지를 전달하고 문자 메시지(SMS)를 미리 등록된 사용자 휴대 전화로 전송하여 신속한 대처가 가능하도록 지원한다.

2. GUI 인터페이스

개발하는 화재 감시 응용 시스템의 인터페이스는 [그림 5]와 같다. 크게 화재 감시 시스템 화면, 감시데이터 화면, Graph 화면으로 구성되고, 위험 상황 경보를 위한 인터페이스가 존재한다. 아울러, 전체 센서노드의 감시 상황을 확인할 수 있는 기능을 제공한다. 이는 현재 센서노드가 배치된 건물이나 방의 조감도([그림 5]의 'Viewer'틀)를 바탕으로 하여 사용자가 쉽게 이해할 수 있는 직관적인 인터페이스를 제공한다.

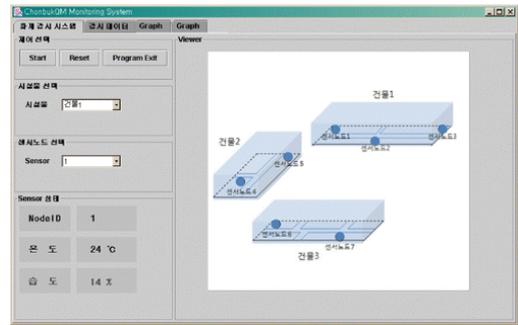


그림 5. 화재 감시 시스템 인터페이스

'Viewer'틀을 통해 센서노드의 이상 유무나 화재 여부에 대해 표시함으로써, 관리·보수를 용이하게 하도록 제공한다. 센서 노드의 이상 유무 표시를 통해 센서노드가 배터리 방전 등의 문제로 인해 작동하지 않을 경우, 사용자에게 미 동작 센서를 알림으로써 외부적인 문제로 감시 서비스가 중단되지 않도록 지원한다. 또한 화재 발생 여부 표시를 통해 화재가 발생 시, 해당하는 센서노드를 붉은색으로 표시하여 화재 발생 지점의 위치 및 위험상황에 대한 정보를 알린다.

둘째, 감시 데이터 화면에서는 센서 네트워크를 통해 주기마다 수집되는 모든 센싱 데이터를 실시간으로 나타낸다. 사용자 질의에 따라 수집되는 센싱 데이터의 종류에 따라 감시 데이터 화면에 표 형식으로 나타난다. [그림 6]은 수집된 노드 식별자, 온도, 습도, 전압 값을 나타낸다. 감시 데이터 화면은 기능은 화재 감시 시스템에서 온도나 습도의 전체적인 데이터 범위 등을 관측하고 싶을 때 유용하게 사용될 수 있다.

timestamp	epoch	nodeid	temp	light	volt
61047	18	2	24	242	2872
61052	18	1	17	432	2795
78197	17	2	24	241	2872
78198	17	1	17	432	2795
77282	16	2	24	239	2872
77285	16	1	17	432	2795
68407	15	2	24	239	2878
68389	15	1	17	432	2795
61825	14	2	24	241	2872
61844	14	1	17	432	2795
58704	13	2	24	241	2872
58688	13	1	17	432	2795
51891	12	2	24	238	2872
51875	12	1	17	432	2795
48032	11	2	21	245	2872
48000	11	1	18	432	2795
44157	10	2	24	244	2878
44155	10	1	17	432	2795
38078	9	2	21	287	2872
38053	9	1	14	432	2795
34718	8	2	24	368	2872
34188	8	1	14	432	2795
29341	7	1	16	431	2881
29344	7	2	24	381	2872
24489	6	1	18	432	2795

그림 6. 감시 데이터

셋째, Graph 화면은 실시간으로 수집되는 각 노드의 센싱 데이터에 대한 그래프를 제공한다. [그림 7]과 [그림 8]은 온도와 습도에 대한 변화 그래프를 나타낸다. 온도 그래프 화면은 현재 배치된 각 센서노드가 수집한 온도 정보를 바탕으로 나타낸 실시간 그래프이다. 또한 습도 그래프 화면은 각 센서노드에서 수집한 습도 정보를 그래프로 나타낸다. 이러한 그래프 기능을 통해 사용자는 데이터가 증가하거나 감소하는 변화량을 쉽게 파악할 수 있고, 과거 데이터에 대한 상태 변화 추이도 쉽게 확인할 수 있다

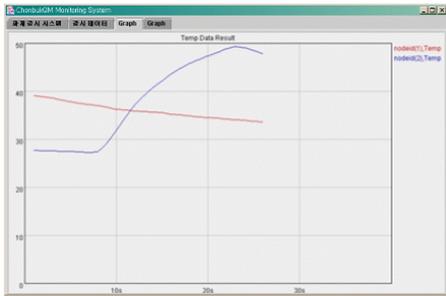


그림 7. 온도 그래프

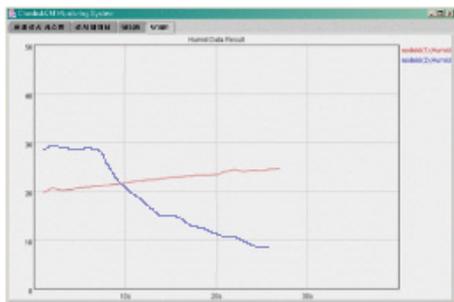


그림 8. 습도 그래프

마지막으로 화재 발생시, 화재 감시 및 위험 상황 정보 인터페이스를 통해 사용자에게 경고 메시지를 전달한다. [그림 9]와 같이 경고창을 띄워 해당 센서 노드의 식별번호를 나타내고, 동시에 뷰어 창에 화재가 발생한 센서노드의 위치를 붉은색 원으로 표시한다. 아울러, 문자 메시지(SMS)를 미리 등록된 사용자 휴대 전화로 전송하여 신속한 대처가 가능하도록 지원한다.

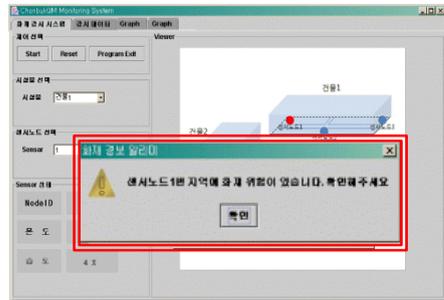


그림 9. 화재 상황 경고창 및 뷰어

V. 성능 평가

조기 경보를 위한 화재 판단 알고리즘을 이용한 화재 감시 응용 시스템을 AWT(The Abstract Window Tool-kit)를 이용하여 자바로 개발하였다. 따라서 Applet을 이용한 클라이언트 프로그램은 원격지의 사용자가 웹 브라우저를 통해 쉽게 접속할 수 있다. 개발된 화재 감시 응용 시스템은 [그림 10]과 같다. 무선 센서네트워크 구성을 위한 센서노드와 게이트웨이(싱크노드), 수집 데이터 처리 및 화재 감시를 수행하기 위한 서버, 원격지의 사용자를 지원하기 위한 클라이언트 시스템으로 구성된다.

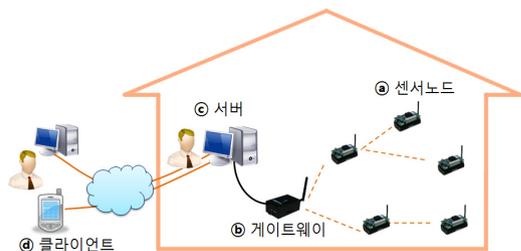


그림 10. 화재 감시 응용 시스템 구성

센서노드([그림 10]의 ㉓)는 센서보드 MTS400[9]과 MicaZ 모트(mote)[10]로 구성된다. 센서보드는 온도, 습도 등의 여러 센서를 내장하고 있어 수집한 상태 정보를 커넥터로 연결된 모트로 전송한다. 이때 모트는 센서보드를 통해 수집한 데이터를 무선 신호로 변환하여 센서 네트워크상의 다른 센서노드나 게이트웨이(Gateway)로 전송한다. 게이트웨이([그림 10]의 ㉔)는 MIB520[11]을 사용하고, 센서노드로부터 수신한 센싱 데이터를 서버로 전송한다. 이때 게이트웨이는 센싱하지 않고 데이터를 수신하여 서버로 전송하는 역할만 수행한다. 또한 서버에서 발생한 사용자 질의를 센서 네트워크상으로 전송한다. 서버([그림 10]의 ㉕)는 게이트웨이로부터 수신한 데이터를 바탕으로 화재 감시를 수행하고, 인터넷망을 통해 클라이언트에게 서비스를 제공한다. 아울러, 화재 발생 시 클라이언트 PC를 통해 원격지의 사용자에게 경고창을 통해 화재 상황 발생 메시지를 전달한다. 특히 사용자 휴대폰으로 화재 경보 SMS를 발송함으로써 신속한 대처를 지원한다. 클라이언트 시스템([그림 10]의 ㉖)은 원격지의 사용자가 장소에 구애받지 않고 수집 데이터 현황 및 화재 감시 현황을 인터넷 망을 통해 확인할 수 있도록 지원한다.

1. 화재 상황 시나리오

화재 상황을 재현하는 실험을 다음과 같은 시나리오를 바탕으로 MicaZ 센서노드를 이용하여 수행한다.

- ① 센서노드 및 싱크노드에 미들웨어를 탑재한다.
- ② 싱크노드를 서버PC에 연결하고 센서노드를 배치한다.
- ③ 서버 프로그램의 시작 버튼을 클릭하여 화재 감시 시스템을 시작한다.
- ④ 화재 발생 상황을 재현하기 위해 [그림 11]과 같이 센서노드를 전기난로 가까이 가져간다.



그림 11. 화재상황 재현

시간이 경과하면 난로 가까이 위치한 센서노드의 온도가 증가하고 습도가 하강하게 된다. 이에 따라 서버에서 화재 판단 알고리즘에 의해 조기에 화재 발생을 탐지하면, [그림 12]와 같이 사용자에게 화재 발생 메시지를 전송한다. 이러한 SMS발송을 통해 사용자 부재시에도 화재 상황을 신속하게 대처할 수 있도록 지원하는 위험 상황 경보를 수행한다.

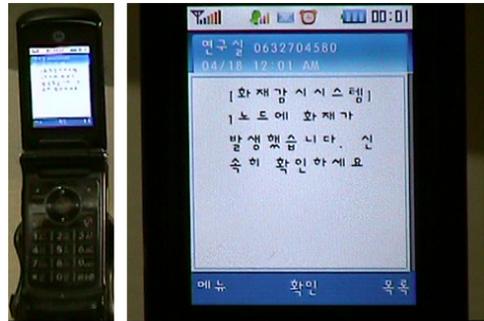


그림 12. 화재 경보 SMS 서비스

2. 화재 발생 상황에 따른 성능평가

화재 발생 시 인화물질의 종류, 발화점과 센서노드와의 거리 등에 따라 온도나 습도의 변화는 다양하게 나타날 수 있기 때문에, 다양한 화재 발생 상황을 위한 데이터 모델을 수립한다. 실대형 실물화재 실험을 수행한 연구[12]를 통해 화재 발생 상황을 크게 3가지로 나누어 데이터 모델을 수립한다. 상황 1은 특수한 경우를 실험하기 위한 모델이다. 발화점과 배치된 센서노드의 거리가 가까울수록 높은 온도 증가율을 나타내므로, 이때의 발화점과 센서노드의 거리의 측정값은 70cm 이내이

다. 상황 2는 일반적인 화재 발생 상황을 의미하며, 발화점과 센서노드와의 거리의 측정값은 반경 3m 내외이다. 상황 3은 겨울철 난방중인 상황을 가정한 데이터 모델이고, 난방기기와 센서노드와의 거리는 반경 5m 내외이다. [그림 13]은 위와 같은 화재 발생 상황의 온도 변화 데이터를 나타낸 것이다.

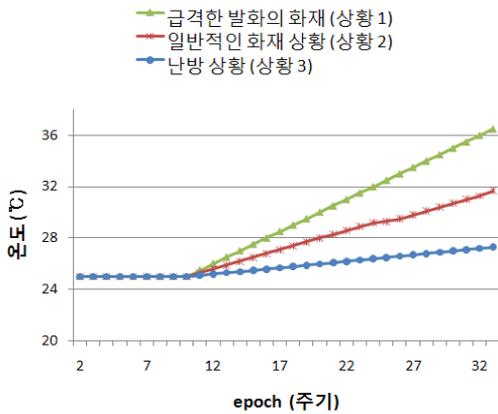


그림 13. 화재 발생 상황 모델

이러한 세 가지 데이터 모델을 바탕으로 화재 발생 상황에 따른 성능평가를 위해, 화재 감시 응용 시스템에 메시지 튜플(tuple) 형태로 데이터를 삽입하였다. 이를 통해 일정 주기마다 화재 판단을 수행하는 Yanjun Li의 알고리즘[6](이하 Li 알고리즘이라 칭함)과 성능을 비교한다. 이 연구는 임계값 이상의 온도가 탐지되거나 일정 온도 이상 상승하는지를 검사하여 화재 판단을 수행한다. 제안하는 화재 감시 시스템과의 성능 비교를 위하여 화재 판단을 위한 주기, 임계값, 온·습도 변화량을 동일하게 설정하여 실험하였다.

일반적인 화재 발생 상황을 위한 최적의 화재 감시 판단 조건은, 화재 판단 주기(10epoch)당 온도 3°C 이상 증가하고, 습도가 3% 이상 하강하는 지점이다. 이러한 화재 판단 조건을 바탕으로 조기 경보를 위한 화재 판단 알고리즘을 통해 화재 발생 여부를 판단한다. 상황 3은 난방중인 상황이므로, 온도 증가량이 미미하여, 제안하는 시스템과 Li 알고리즘에 대한 실험에서 모두 화재 상황으로 판단되지 않는다. 32epoch 이후에도 온도

가 꾸준히 증가한다면 절대 온도에 의한 화재 판단 알고리즘에 의해 화재 판단을 수행하게 된다. 일반적으로 화재 감시 시스템에서 화재 판단 주기는 일정하게 반복되고, 또한 화재가 어느 시점에 발생 될지 예상할 수 없으므로 화재 발생 시점을 변화시키면서 실험하여 화재 탐지 시간을 측정한다. 이를 위해 화재 상황에 대한 실험 데이터는 각 모델에 따른 온도 값으로서 $X_{temp} = \{x_1, x_2, x_3, \dots, x_n\}$ 라고 정의할 때, 온도가 급격히 증가하는 화재 발생 시점은 x_K 라고 가정한다. 화재 발생 시점을 변화시키기 위하여 각 X_{temp} 의 시작 주기를 1epoch(초)부터 50epoch까지 변화시켜가며 각 화재 상황에 따라 50회씩 실험을 수행하였다. 이때 화재 탐지 시간은 화재 발생 시점 x_K 의 주기로부터 화재 판단을 위해 경과한 시간을 의미한다. [그림 14]는 상황 1에 따른 화재 판단 시간을 나타낸다. 제안하는 화재 감시 시스템이 Li 알고리즘에 비해 평균 화재 판단 시간이 약간 늦지만, 최고 화재 판단 epoch는 제안하는 시스템이 더 낮은 것을 볼 수 있다. 이는 Li 알고리즘이 평균적으로 더 빨리 화재를 판단하는 경우가 있지만, 최악의 경우에는 화재 판단이 지연된다는 것을 의미한다. 즉, Li 알고리즘은 최적의 화재 판단 시간을 항상 보장하지 못한다는 것을 나타낸다. [그림 15]는 상황 2에 따른 화재 판단 시간을 나타낸다. 가장 일반적으로 발생하는 화재 상황인 상황 2에서 제안하는 시스템은 화재가 발생한 시점으로부터 평균 14epoch에 화재 발생을 탐지하여 Li 알고리즘보다 좋은 성능을 나타내고, 최고 화재 판단 시간도 Li 알고리즘보다 12epoch 빠른 성능을 나타낸다. [그림 14]와 [그림 15]를 통해 살펴보면 화재가 급격하게 진행되는 상황 1의 경우 화재 판단이 용이하기 때문에 Li 알고리즘도 일정 성능을 유지함을 알 수 있다. 하지만 일반적으로 발생하는 화재 상황인 상황 2의 경우 화재가 완만하게 진행됨에 따라 화재 판단의 지연이 생겨 제안하는 알고리즘이 성능이 더욱 좋은 성능을 나타낸다.

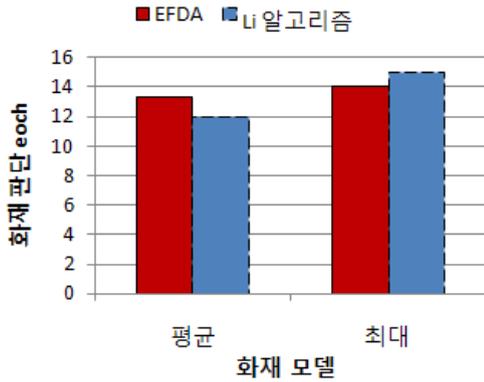


그림 14. 상황1에 대한 화재 판단 epoch

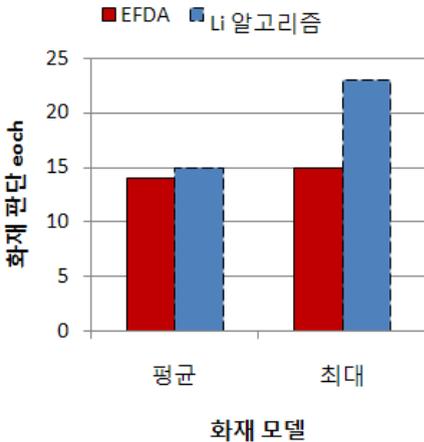


그림 15. 상황2에 대한 화재 판단 epoch

[그림 16]은 상황 2에 대한 화재 판단 epoch의 분포도를 나타낸다. Li 알고리즘은 가장 좋은 성능을 나타내는 경우, 화재 발생 시점으로부터 9 epoch만에 화재 판단을 수행하지만, 느린 경우 17~23epoch까지 화재가 진행된 후에야 탐지하는 경우가 발생한다. Li 알고리즘은 데이터 분포를 고려하지 않고 설정된 주기에 의해 반복적으로 화재 판단을 수행하기 때문에 화재 탐지가 지연된다. 그러나 제안하는 시스템에서는 이러한 최악의 경우를 방지하기 위해 데이터의 분포를 고려한다. 데이터가 증가하는 시점을 찾아서 화재 판단 주기의 시작점으로 하여 알고리즘을 수행하기 때문에, 빠른 시간

내에 화재를 탐지하면서 일정한 성능을 보장한다. 성능 평가를 통해 본 시스템은 화재가 발생하기 시작한 시점으로부터 12~14 epoch 이내로 모든 경우에 대해 화재 탐지 및 경고를 수행한다.

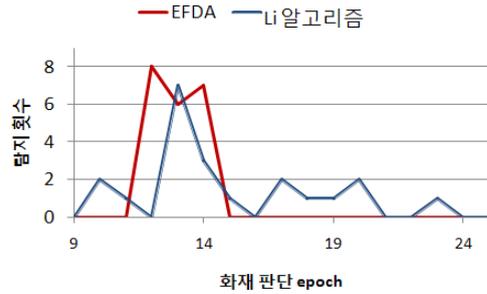


그림 16 화재 판단 epoch 분포도 (일반적인 화재 상황 -상황 2)

3. 기존 연구와의 특징 비교

조기 경보를 위한 화재 판단 알고리즘을 이용한 개발된 화재 감시 응용 시스템 및 기존 화재 감시 응용 시스템의 특징은 [표 1]과 같다. 첫째, 화재 감시 응용 시스템의 초기 연구인 Firebug 프로젝트는 화재 상황에서의 실시간 데이터 수집을 통해 웹기반 클라이언트를 통해 나타내었다. 둘째, 고대건축물을 위한 화재 감시 시스템은 주기적인 화재 감시를 수행하며, 측정된 온도 값이 상한선을 초과하는지를 확인하여 화재 발생 여부를 판단하였다. 아울러 사용자를 위해 데이터 수집 및 감시를 위한 GUI 인터페이스를 제공하였다. 셋째, Li의 시스템에서는 사용자 편의성을 위하여 웹 기반 클라이언트를 구현하였다. 마지막으로 본 시스템은 기존 화재 감시 시스템에서 화재 판단이 지연되는 문제점을 해결하기 위해, 조기 경보를 위한 화재 판단 알고리즘을 제시하였고, 이를 이용하여 효율적인 화재 감시 응용 시스템을 개발하였다. 또한 화재 감시 응용시스템의 관리 및 유지 보수를 효율적으로 지원하는 GUI 인터페이스를 서버-클라이언트 구조로 개발하여 원격지에서도 사용 가능하도록 하였다. 마지막으로, 화재 발생 시 경고 장과 SMS를 통해 위험 상황 경보를 전송함으로써 사용자 부재 시에도 신속한 대처를 지원할 수 있다.

표 1. 화재 감시 응용 시스템 특징

	Firebug [3]	고대건축물 화재 감시 [5]	Yanjun Li 알고리즘[6]	EFDA
화재 감시 수행	×	△	○	○
GUI 인터페이스	×	○	×	○
화재 위험 경고	×	○	○	○
웹기반 클라이언트	△	×	○	○

○지원 △부분 지원 ×미지원

VI. 결론 및 향후 연구

본 논문에서는 문화재 및 시설물 관리를 위한 효율적인 화재 감시 응용 시스템을 개발하였다. 기존 시스템의 문제점을 해결하기 위하여 조기 경보를 위한 화재 판단 알고리즘을 제안하였고, 이를 통해 사용자에게 위험 상황 경보를 전송하는 화재 감시 응용 시스템 인터페이스를 개발하였다. 개발된 화재 감시 응용 시스템은 화재 발생 시 조기 경보를 통해 신속한 위험 상황의 대처를 지원하고, GUI 인터페이스를 통해 효율적인 시스템 관리 및 유지 보수를 지원한다. 성능평가를 통해, 개발한 시스템의 유용성을 검증하고, 다양한 화재 발생 상황에서 조기 경보를 위해 활용 될 수 있음을 입증했다.

향후 연구로는 개발된 화재 감시 응용 시스템을 확장하여, 침입 감지 등 다양한 응용 서비스를 지원하고, 하드웨어 및 소프트웨어 결합으로 인한 시스템 문제를 최소화하기 위해 fault-tolerant 기능을 갖춘 무선 네트워크 기반의 실내 안전 관리 시스템에 대한 연구를 수행하는 것이다.

Fire Rescue Applications: Requirements and Challenges," Electro/ information Technology, IEEE International Conference, 2006(5).

[5] Hongli Xu, "Wireless fire monitoring system for ancient buildings," Vol.304, No.2, Infoscalle, ACM 2007.

[6] L. Yanjun., "Wireless sensor network design for wildfire monitoring," Vol.1, pp.109-113, GLOBECOM, IEEE 2006.

[7] C.-L. Fok, "Mobile Agent Middleware for Sensor Networks: An Application Case Study," IPSN, ACM 2005(4).

[8] 변정호, Safe Korea 칼럼, "소방차 길 터주기' 귀중한 생명 살린다", 2007(9).

[9] http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MTS400-420_Datasheet.pdf

[10] http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MICAZ_Datasheet.pdf

[11] http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/MIB520_Datasheet.pdf

[12] 이재승, "병원용 모듈러 블록의 내화성능 평가", RIST 2009.

참고 문헌

[1] 김지은, 김세한, 정운철, 김내수, "USN 센서노드 기술 동향", 전자통신동향분석 제22권, 제3호, 2007(6).

[2] 박용재, 임명환, 김관중, "USN 서비스 및 시장동향", 전자통신동향분석 제24권, 제2호, 2009(4).

[3] <http://firebug.sourceforge.net/>

[4] S. Kewei, "Using Wireless Sensor Networks for

