

# Improvement of SWoT-Based Real Time Monitoring System

Myung-han Yu<sup>†</sup> · Sangkyung Kim<sup>††</sup>

## ABSTRACT

USN-based real-time monitoring systems, which receive raw data from sensor nodes and store the processed information in traditional servers, recently get to be replaced by IoT(Internet of Things)/WoT(Web of Things)-based ones. Especially, Social Web of Things(SWoT) paradigm can make use of cloud storage over Social Network Service(SNS) and enable the possibility of integrated access, management and sharing. This paper proposes an improved SWoT-based real-time monitoring system which makes up for weak points of existing systems, and implements monitoring service integrating a legacy sensor network and commercial SNS without requiring additional servers. Especially, the proposed system can reduce emergency propagation time by employing PUSH messages.

**Keywords :** IoT, WoT, SWoT, SNS, Monitoring

## SWoT 기반 실시간 모니터링 시스템 개선

유 명 한<sup>†</sup> · 김 상 경<sup>††</sup>

## 요 약

기존의 USN 실시간 모니터링 시스템은 데이터를 수신받아 별도의 서버 시스템에 데이터베이스를 구성하여 정보를 처리하였으나, 최근의 IoT 및 WoT 기술의 대두로 인해 기존의 USN 환경에 새로운 Social Web of Things(SWoT) 개념을 도입하면 소셜 네트워크 서비스(SNS) 상에서 비용 효율적이고 통합적인 관리, 공유, 접근을 가능케 할 수 있다. 본 연구에서는 응급상황 전파에 PUSH 메시지 서비스를 접목하여 전달 시간을 단축시키고, 별도의 서버 구축 없이 SNS가 제공하는 부가 기능만으로 동작하는 모니터링 시스템을 구성 및 구현함으로써 기존 SWoT 기반의 실시간 모니터링 시스템의 단점을 극복할 수 있는 개선형 모델을 제안하고 검증한다.

**키워드 :** IoT, WoT, SWoT, SNS, 모니터링

### 1. 서 론

현재 다양한 실시간 자동 관측 분야에서 USN(Ubiquitous Sensor Network)이 사용되고 있다. USN을 바탕으로 한 실시간 모니터링 시스템은 관측 장소에 구성된 센서 네트워크로부터 데이터를 수신받아 별도의 서버 시스템에 데이터베이스를 구성하고, TCP 기반의 비표준적인 별도 프로토콜을 이용하는 모니터링 프로그램이 데이터베이스의 자료를 수신하여 표출하는 형태를 갖춘다[1].

최근 IoT 기술의 대두로 인해 기존의 USN 환경에 새로운 Social Web of Things(SWoT) 개념을 도입하여 모니터링을 원하는 사용자에게 소셜 네트워크 서비스(SNS) 상에

서 스마트 오브젝트의 통합적인 관리, 공유, 접근을 제공할 수 있다[2]. 이는 구축 및 유지보수 과정 전체에서 비용 절감 효과를 가져올 수 있으며, 기존의 USN 데이터 로거뿐만 아니라 다양한 이기종 IoT 디바이스를 지원할 수 있다.

SWoT 기반 모니터링 시스템은 실시간으로 모니터링에 필요한 각종 데이터를 수집하여 이를 게이트웨이로 전송하고, 별도의 서버가 아닌 SNS 서비스에 데이터를 업로드하는 형태로 구성된다[2-4].

하지만 이러한 SWoT 기반의 모니터링 시스템은 센서 네트워크에서 데이터를 SNS에 포스팅하고 모니터링 시스템에서 이를 수신받는 과정에서 지연되고, 긴급한 데이터 처리를 위해 특화된 전송 수단이 없다는 문제점이 있다. 또한 통계와 제어명령 전송 등의 SNS에서 지원하지 않는 기능을 구현하기 위해 기존의 USN 기반 시스템과 동일한 별도의 센서 네트워크 제어용 패킷과 함께 독립된 서버 및 데이터베이스 스토리지를 요구하게 되는 단점이 있다[4, 5].

본 연구에서는 기존 SWoT 기반의 모니터링 시스템에서

<sup>†</sup> 준 회원 : 강릉원주대학교 컴퓨터공학과 박사수료

<sup>††</sup> 종신회원 : 강릉원주대학교 컴퓨터공학과 교수

Manuscript Received : February 12, 2015

First Revision : April 27, 2015

Accepted : May 21, 2015

\* Corresponding Author : Myung-han Yu(skkm98@gwnu.ac.kr)

는 기대하기 힘들었던 응급 상황의 신속한 전파를 위해 상용 PUSH 메시지 서비스를 접목하여 전달과정에서 성능적인 향상을 이루어낸다. 또한 기간별 통계 데이터 산출 및 USN 장치의 제어, 이미지 센서 데이터의 저장과 표시를 별도의 서버 구성 없이 SNS가 제공하는 부가 기능만으로 구현함으로써 관리 효율 면에서도 개선된 SWoT 기반 실시간 모니터링 시스템을 제안한다. 또한 이 개선형 모델을 바탕으로 상용 SNS 상에서 모니터링 시스템을 실제로 구현하고 시험한다.

본 논문의 구성은 다음과 같다. 2절에서는 기존 USN 및 SWoT 기반 모니터링 시스템 관련 연구내용을 살펴보고, 3절에서는 개선된 시스템의 기능별 상세 구성을, 4절에서는 시스템의 구현과정과 사용자 인터페이스를 기술하고 성능을 평가하였으며, 5절에서는 결론을 맺는다.

## 2. 관련 연구

### 2.1 기존의 USN 실시간 모니터링 시스템 모델

기존의 USN 실시간 모니터링 시스템은 데이터 저장을 위해 데이터 수집 서버를 필수적으로 요구하며[1, 6] 이러한 시스템의 일반적인 연결 구성 및 시스템 특성을 Fig. 1과 Table 1에 나타내었다.

기존의 시스템은 비표준 프로토콜을 이용하여 데이터 서버와 모니터링 시스템을 연결하게 되므로 확장성 면에서 문제시된다. 예를 들어, 센서종류 추가 등 기능 개선 시 프로토콜과 데이터베이스 구조의 변경 작업이 필요하며, 이에 대한 통합적인 관리가 불가능하여 별도의 모니터링 시스템과 지정된 단말장치로만 센서 네트워크로의 접근 및 제어가 가능하다. 또한, 응급 상황 시 발생하는 상황전파 역시 휴대전화 문자 메시지 및 상황실 경고장치 등의 제한된 수단으로만 수신 가능하다는 단점이 있다.

이러한 문제점을 극복하기 위해, 최근에는 SNS에 기반한 SWoT 개념을 통해 USN 모니터링 시스템을 대체하는 시도가 이루어졌다[2-4].

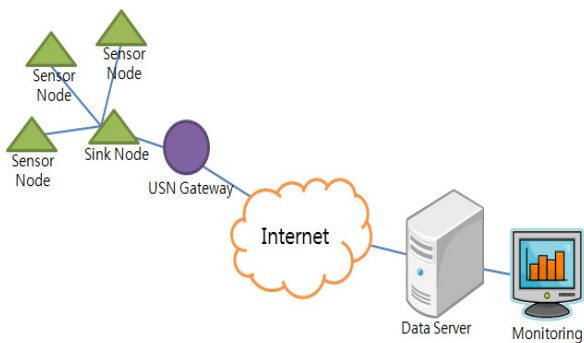


Fig. 1. Schematic Diagram of a Legacy USN Monitoring System

Table 1. Property of Legacy USN Monitoring System

구성요소	특성	연결방식
센서노드	센서데이터 수집 및 변환, 타 노드들과 통신 및 데이터 상호교환	2.4GHz IEEE 802.15.4 (또는) 802.11 무선 통신, RS232 유선 통신
USN 게이트웨이	싱크노드로부터 데이터 수신, 데이터 서버로 전송	수신 측 : RS232 송신 측 : 유선 이더넷/ 3G/4G CDMA등의 무선 통신수단 이용
데이터 서버	게이트웨이로부터 수신한 데이터를 DBMS에 저장	TCP/IP, 유선 이더넷 비표준 프로토콜 이용
모니터링 시스템	서버로부터 모든 데이터를 수집하여 표시	TCP/IP, 유선 이더넷 비표준 프로토콜 이용

### 2.2 SWoT 기반의 실시간 모니터링 시스템 모델

SWoT란 인터넷과 소셜 미디어를 이용하여 다양한 IoT 장치들을 연결하고 데이터를 공유하는 개념으로[2], USN 모니터링 시스템에 SWoT 개념을 도입하면 서비스를 제공하는 측에서 SNS를 일종의 단순 데이터 스토리지로 활용함과 동시에 사람 간의 소셜 네트워크와 유사한 장치 간의 관계를 형성함으로써 다양한 이득을 취할 수 있다. 이메일, 채팅, 인스턴트 메시징(IM) 등의 수단으로 SNS를 통해 센서 네트워크에 쉽게 접근할 수 있게 되며[2], 이러한 서비스를 대부분의 경우 무료로 제공받을 수 있고, 표준화된 API를 사용하므로 전체적인 서비스 개발 시간 역시 단축시킬 수 있어 경제적이다[4]. 결함 감내(fault tolerance)식으로 구성된 상용 SNS의 특성상 서비스가 24시간 안정적으로 동작할 수 있으며, 다수의 모니터링 시스템 사용자와 장치를 별다른 제약 없이 사실상 무제한으로 수용할 수 있다는 장점이 있다. 이러한 SWoT 기반의 실시간 모니터링 시스템의 구성을 Fig. 2에 나타내었다.

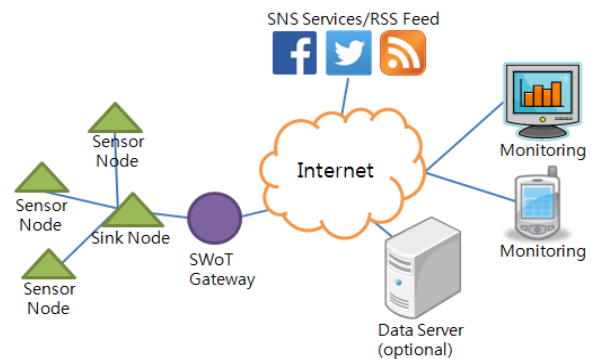


Fig. 2. Schematic Diagram of SWoT Monitoring System

Dominique Guinard et al.의 연구에서는 Web of Things(WoT)의 개념을 SNS로 확장, 데이터에 대한 인증과 공유, 시스템 구현에 대한 모델과 가능성을 제시하였다[3]. Yacine et al.,과 Chung et al.의 연구는 이러한 Social Web

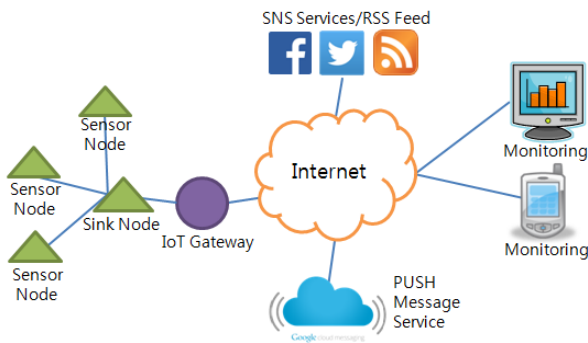


Fig. 3. Schematic Diagram of the Proposed System

of Things의 개념을 기존의 센서 네트워크 시스템에 접목할 수 있도록 게이트웨이 등 각 구성요소를 직접 구현하기 위한 모델과 방법을 제시한다[7, 8]. M. Baqer의 연구에 따르면 트위터 서비스의 메시지 길이 140자 제한은 센서 네트워크에 사용하기에 적합한 수준으로, 실제로 이를 이용하여 USN 데이터를 트위터 서비스로 전송하여 저장 및 제공하는 기본적인 서비스를 구현하였다[9].

그러나 이러한 기존 SWoT 기반의 실시간 모니터링 시스템에는 긴급한 이벤트가 발생할 경우 SNS로부터 데이터를 수신하는 과정에서 전파가 늦는다는 문제점이 있으며, 게이트웨이와 센서노드를 대상으로 한 명령어 전송수단의 부재, SNS를 스토리지로 활용 시 통계 및 보고서 등의 부가기능 구현에 제약이 따른다는 문제가 있다[1, 9, 10]. 또한 실시간 센서 데이터 및 통계를 위해 별도로 저장하는 텍스트 형태 데이터는 사람이 직접 읽을 수 없다는 문제가 있다[4].

### 3. SWoT 기반의 실시간 모니터링 시스템 개선

#### 3.1 시스템 개선 개요

본 논문에서 제안하는 시스템은 센서노드로부터 수집되는 데이터를 SNS에서 규정하는 API를 이용하여 포스팅하며, 모니터링 시스템이 이를 SNS를 통해 수신하는 SWoT 구조를 바탕으로 하여 추가적으로 개선된 부가기능을 별도의 서버 시스템 구축 없이 제공할 수 있다.

개선된 부가기능에는 긴급한 이벤트가 발생할 경우 게이트웨이와 모니터링 시스템 간의 PUSH 메시지 전달 기능, 모니터링 시스템에서 센서 네트워크로의 DM(Direct Message) 발송을 통한 장치 제어 기능, 기간별 통계 및 보고서 산출을 가능토록 하는 게이트웨이 측의 주기별 통계 저장 및 모니터링 측의 SNS 통계 및 보고서 산출 기능, 이미지 데이터 처리를 위한 사진 포스팅 기능이 있다.

통계와 장치 제어를 포함한 모든 SNS 상의 메시지는 사람이 직접 읽을 수 있는(human-readable) 포맷을 갖추어 별도의 클라이언트 없이 SNS 타임라인을 통해서도 확인이 가능토록 한다. PUSH 메시지 전달기능 또한 구글이나 애플 등에서 제공하는 클라우드 메시징 서비스 API를 이용, 별도

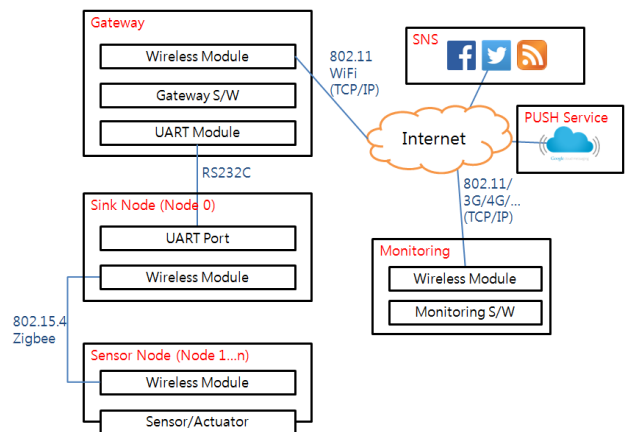


Fig. 4. Detailed View of the Proposed System

의 서버 구축 없이 구현토록 하며[11], SNS와 PUSH 메시지 서버 접속을 위한 인터넷(IPv4 또는 IPv6) 연결은 상용 통신망을 활용한다.

다수의 상용 SNS가 자사 서비스의 외부접속을 위해 JSON(JavaScript Object Notation) 또는 XML(Extensible Markup Language) 기반의 REST(REpresentational State Transfer) 웹 서비스 API를 지원하고 있으므로, 이를 적극적으로 활용한다[10, 11].

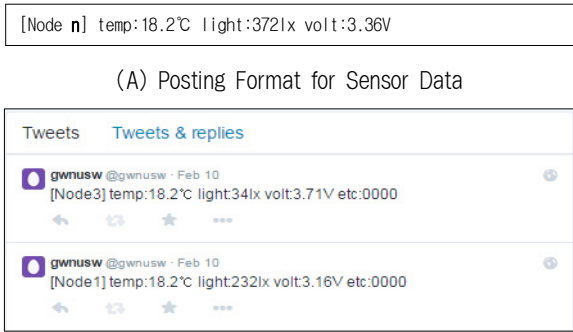
본 연구에서 제안하는 모니터링 시스템의 구성은 Fig. 3과 같다. USN 노드는 측정된 센서 데이터를 싱크노드와 게이트웨이로 송신하는 전통적인 USN 형태를 취하고, SWoT 게이트웨이는 각 USN 노드로부터 수집된 데이터를 가공하여 SNS에 포스팅한다. 게이트웨이는 또한 친구 관계인 특정 사용자의 SNS 타임라인 또는 DM을 읽어들이어 제어 명령을 수행하며, DM을 통해 응급 상황 발생 시 PUSH 메시지를 수신할 모니터링 시스템의 경로에 대한 가입(Subscription)을 수행한다.

모니터링 시스템은 SNS에 포스팅된 게이트웨이의 타임라인을 이용하여 모니터링 화면의 데이터를 구성하고, DM을 통해 게이트웨이로 제어 명령을 전송할 수 있다. 또한 응급 상황 시 PUSH 메시지를 수신하기 위한 가입 요청 전송과 PUSH 메시지 수신 역할을 수행한다.

제안하는 시스템의 상세한 구성을 Fig. 4에 나타내었다. 본 시스템에서는 기존의 USN 센서노드가 텍스트 형태의 데이터를 싱크노드(Node 0)로 전송하며, 이 데이터는 RS232를 이용해 게이트웨이로 전달된다. 게이트웨이는 얻은 데이터를 SNS 및 PUSH 서비스에서 사용할 수 있는 형식으로 가공하여 인터넷을 통해 트위터 및 Google Cloud Messaging(GCM) 등의 상용 SNS/PUSH 서비스로 전달한다. 모니터링 시스템은 SNS 및 PUSH 서비스에 접속하여 모니터링 데이터를 수신 및 표시한다.

#### 3.2 센서 데이터의 전달

SWoT 게이트웨이는 센서 네트워크로부터 수집한 센서



(B) Twitter Timeline Controlled by SWoT Gateway  
Fig. 5. Human-readable SNS Posting

데이터 정보를 인터넷을 통해 SNS에 포스팅 형태로 전달한다. 개선된 시스템에서는 이 포스팅 형식을 사람이 읽을 수 있는 포맷으로 등록함으로써 SNS에서 제공하는 자체 타임라인에서도 간단하게 최근 데이터를 확인할 수 있다. 이러한 사람이 직접 읽을 수 있는 형태의 센서 데이터 및 해당 데이터로 형성된 SNS 타임라인을 Fig. 5에 나타내었다.

각 데이터 필드는 띄어쓰기로 구분하며, 앞에서부터 순서대로 포스팅된 정보의 종류와 센서노드 번호(n), 센서 또는 모니터링 항목 이름, 데이터값 정보로 구성된다. 각 포스팅의 길이는 트위터 서비스의 140자 최대 제한을 따르는 한도 내에서 자유 길이를 가진다.

SNS에 저장된 데이터의 표현을 위해, 휴대용 모니터링 시스템은 데이터가 필요한 시점에 각 SNS에 대해 Open API에 규정된 REST 기법을 이용하여 필요한 데이터를 요청한다[10]. 트위터 SNS API를 기준으로 모니터링 프로그램에 전달되는 JSON 데이터 포맷은 Fig. 6과 같다. created\_at 및 geo 필드는 각기 센서노드가 위치한 좌표와 게이트웨이로부터의 데이터 전송시간을 나타내며, text 필드는 포스팅된 텍스트 데이터를 나타낸다. entities/media에는 첨부된 이미지센서 데이터의 정보가 실려있으며, id값을 통해 포스팅된 데이터의 중복 여부를 검출해낼 수 있다.

```

(root)
{
  "created_at": "",      ... Tweet register date/time
  "id": "",            ... Message ID (Unique)
  "text": "",          ... Message body (Posting content)
  "geo": "",           ... Location (Geo) info
  "entities": "",     ... Additional/extended info
  "media": "",        ... Contained media info (Photo)
  { "media_url": "" } ... URL of media file
}
    
```

Fig. 6. JSON Data Format of Twitter

### 3.3 응급 상황 데이터의 전달(PUSH)

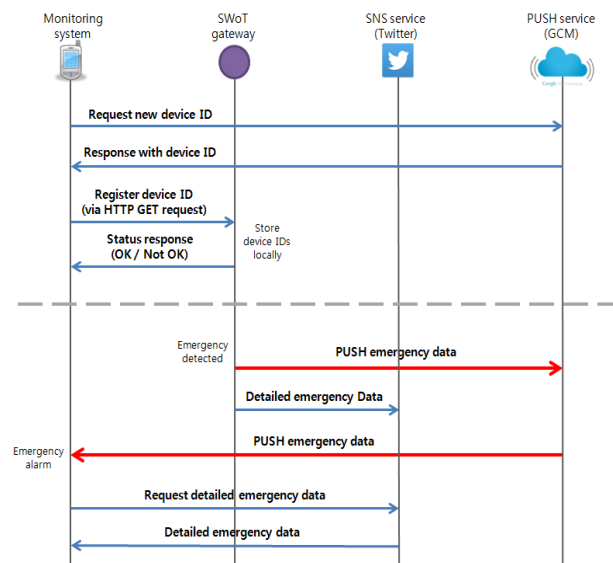
SNS에서 제공하는 API를 통해 주기적으로 또는 필요한 경우에만 데이터를 수신하는 경우, 각종 센서의 수치 초과 및 미달 등의 이상 발생을 나타내는 응급 상황 데이터의 빠른 수신이 불가능하다는 단점이 있다. 이를 해결하기 위해 개선된 시스템에서는 SNS의 REST 이외에 PUSH 메시지 프로토콜을 이용한 별도의 긴급 데이터 전달 기능을 정의하였다. 각 게이트웨이에서 응급 상황 데이터를 신속히 수신하기 원하는 모니터링 시스템은 게이트웨이에 미리 지정된 HTTP GET Request용 주소로 가입을 수행하여, 응급 상황 발생 시 데이터가 PUSH 메시지를 통해 즉시 전송되도록 요구한다.

PUSH 가입 과정에서 SNS의 DM 등 자체 API를 사용하지 않고 별도의 URL을 통한 처리 구조를 가지는 이유는 트위터의 경우 DM과 일반 메시지 모두에서 가능한 길이를 140자로 제한하는 반면[10], GCM 장치 등록 ID값의 길이는 140자를 초과하여 최대 4KB까지의 길이를 가지기 때문이다 [11].

PUSH 가입 과정에서 이용되는 URL 형식과 메시지 흐름도를 Fig. 7에 나타내었다. 모니터링 시스템에서는 PUSH 가입을 위해 SWoT 게이트웨이에 내장된 웹 서버로 자신의 GCM 장치 ID값을 전송하고, SWoT 게이트웨이는 이를 수신하여 주기억장치에 배열 변수로 저장한다. 이에 대한 의사코드를 Fig. 8에 나타내었다.

http://gateway:8004/push\_subscription/Register.php?key={DEVICE ID}

(A) PUSH Subscription URL Example



(B) Message Flow for Emergency Data

Fig. 7. Handling of Emergency Data

```

/* Monitoring System */
my_pushkey = get_push_key("GCM"); // from GCM server
subscribed = check_subscription_stat(my_pushkey); // from Gateway

if( !subscribed ) {
    do_subscript(my_pushkey); // to Gateway
}

/* Gateway */
got_push_key = received_push_key_buf; // from Monitoring
add_subscribed_key_list(got_push_key); // to local array variable
    
```

Fig. 8. Pseudo Code of PUSH Subscription Process

모니터링 시스템은 우선 GCM 서버로부터 장치의 고유 ID(my\_pushkey)를 발급받게 되며, 이를 SWoT 게이트웨이에 전송하여 등록여부(subscribed)를 판별한다. 만약 장치가 등록되어있지 않다는 응답을 받으면, 새로운 장치 등록 요청(do\_subscript)을 보낸다. 게이트웨이는 모니터링 시스템으로부터 이를 수신하여 저장(add\_subscribed\_key\_list)한다.

응급 상황이 발생하면 SWoT 게이트웨이는 주기억장치에 저장된 모니터링 시스템들의 ID 배열값을 응급 상황 데이터와 함께 GCM 서버에 전송하고, GCM은 해당 ID들에 대해 PUSH 메시지를 발송한다. 모니터링 시스템은 이를 수신하여 진동 또는 알람음으로 사용자에게 응급 상황이 발생하였음을 알리고, 사용자가 이를 확인하면 SNS 서비스에 접속하여 해당 경고에 대한 상세한 데이터를 가져온다. 이에 대한 의사코드를 Fig. 9에 나타내었다.

센서노드로부터 수신된 센서 데이터(sensor\_data)가 지정된 값(over\_limit, under\_limit) 범위를 벗어나면, 주기억장치에 저장된 모니터링 시스템 목록(subscribed\_push\_key)을 이용하여 GCM에 PUSH 요청(send\_push)을 보낸다.

모니터링 시스템에서 PUSH 메시지가 전달되면 이에 포함된 정보를 이용하여 사용자에게 경고 메시지를 표시하고(alert) 트위터 타임라인으로부터 최신 데이터를 가져와 화면에 표시하게 된다(update\_twitter\_timeline).

```

/* Gateway (Send) */
array subscribed_push_key[] = get_subscribed_key_list();

if( sensor_data > over_limit || sensor_data < under_limit ) {
    message = "DATA " + sensor_data + " EXCEEDS SPECIFIED RANGE";
    send_push(subscribed_push_key, message);
}

/* Monitoring System (Receive) */
function on_receive_push_message() {
    message = get_push_message_content();
    alert( message );
    update_twitter_timeline();
}
    
```

Fig. 9. Pseudo Code of PUSH Send/Receive Process

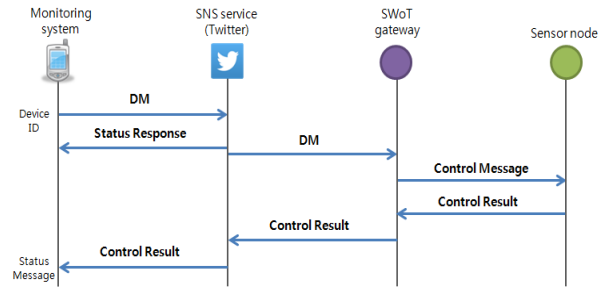


Fig. 10. Message Flow for Device Control Via Twitter DM

사용자는 화면을 통해 이를 확인하고, DM을 통한 장치 제어 명령을 통해 센서노드 및 게이트웨이에 부착된 각종 액추에이터의 동작, 현장 사진 촬영 등의 기능을 수행할 수 있다.

### 3.4 친구기능 및 DM을 통한 장치 제어

트위터에서 제공하는 DM 기능은 사용자 간에 1:1로 메시지를 주고받기 위해 개발되었다[10]. 이를 이용하여 사전에 Follow 등을 통해 친구관계를 형성한 트위터 계정을 대상으로 장치제어 등의 비공개 기능을 수행할 수 있다.

제안하는 시스템에서는 모니터링 시스템이 DM을 통해 제어 명령을 발송하면 메시지가 트위터 SNS를 거쳐 게이트웨이로 전달되는 구조를 취하며, DM을 통해 모니터링 시스템으로부터 센서노드까지 명령어가 전송되는 과정은 Fig. 10과 같다.

장치 제어 메시지는 게이트웨이에서 수신한 뒤 센서 네트워크에 맞는 USN 제어 프로토콜로 변환, 무선을 통해 각 센서노드 또는 네트워크 전체에 전파된다. 모니터링 시스템으로부터 DM을 통해 게이트웨이로 전달되는 제어 명령의 메시지 포맷 및 게이트웨이에서 변환되어 센서 네트워크로 전파되는 제어 명령의 메시지 포맷을 Fig. 11에 나타내었다.

```

// for a single sensor node
[Control n] sensing_interval:4000ms

// for the whole network (using ALL_NODES keyword)
[Control ALL_NODES] sensing_interval:8000ms
    
```

(A) Examples of DM

```

// for a single sensor node
n:sensing_interval:3000ms [CR][LF]

// for the whole network
-1:sensing_interval:3000ms [CR][LF]
    
```

(B) Control Message Translated by a Gateway

Fig. 11. Handling of Device Control Message

```
// Generate at every 00:00:00
[DailyAverage n] between:2015-01-01~2015-01-01 temp:17.0°C

// Generate at every Sunday
[WeeklyAverage n] between:2015-01-01~2015-01-06 temp:16.5°C

// Generate at every Last day of the month
[MonthlyAverage n] between:2015-01-01~2015-01-31 temp:16.5°C

// Generate at every Last day of the year (12/31)
[YearlyAverage n] between:2014-01-01~2014-12-31 temp:16.5°C
```

Fig. 12. Examples of Posting Data Format for Statistics

DM 제어 메시지는 센서 데이터 메시지와 동일하게 각 필드를 띄어쓰기로 구분하며, 앞에서부터 순서대로 제어 메시지 임을 알리는 헤더(Control)와 센서노드 번호, 제어 대상(sensing\_interval), 제어할 값(4000ms) 정보로 구성된다. 각 DM의 길이는 다른 메시지와 동일하게 트위터 서비스의 140자 최대 제한을 따르는 한도 내에서 자유로운 길이를 가진다.

3.5 검색을 통한 통계 기능 구현

타임라인에 통계 데이터를 포스팅하고, 주기적으로 과거 통계 데이터 포스팅을 검색하고 갱신하여 다시 포스팅하는 방법으로 통계 데이터를 최근 값으로 유지할 수 있다. 제안하는 시스템에서는 통계 데이터를 유지하는 역할을 게이트웨이에서 수행하며, 게이트웨이는 센서 데이터와 유사한 형식의 통계 데이터를 생성하여 SNS에 포스팅한다. 통계 데이터의 포스팅 포맷은 Fig. 12와 같다.

모니터링 시스템에서는 게이트웨이가 통계 데이터를 저장할 때 사용한 “between” 및 “Average” 키워드에 대한 트위터 검색 요청만으로 별도의 서버 구축 없이 통계작업을 수행할 수 있다.

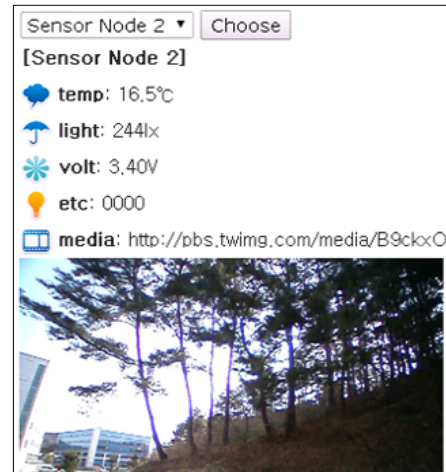
3.6 사진 포스팅 기능을 통한 이미지 센서 데이터 공유

트위터에서 제공하는 사진 포스팅 기능을 활용하면 별도의 서버 없이도 이미지 데이터를 사실상 무제한으로 보관할 수 있는 이점이 있다. 모니터링 시스템은 트위터 서버에 포스팅된 메시지의 첨부파일 이미지를 HTTP를 이용하여 수신하여 모니터링 화면에 표시한다. 이미지 데이터는 다른 센서 데이터와 함께 게이트웨이에 의해 주기적으로 포스팅되거나, 모니터링 시스템으로부터 발송되는 DM 장치 제어 메시지에 의해 비주기적으로 촬영하여 포스팅될 수 있다.

이미지 데이터의 촬영 및 포스팅을 요청하는 DM의 포맷과, 모니터링 시스템에서의 DM 발송 이후 처리결과 표시화면을 Fig. 13에 나타내었다.

```
[Control n] take_photo:true timer:0
```

(A) DM Format for Photo Send Request



(B) Monitoring Interface after Sending DM

Fig. 13. Photo Posting Example

모니터링 시스템에서 DM을 통해 지정된 형태의 이미지 포스팅 요청을 보내면 게이트웨이는 이를 수신하여 센서노드 또는 게이트웨이에 장착된 이미지 센서를 통해 사진을 촬영하고, SNS에 사진이 첨부된 포스팅을 업로드한다. 촬영된 이미지 데이터는 모니터링 시스템에 즉시 나타난다.

4. 구현 및 시험

4.1 구현

실제 적용가능성을 확인하기 위하여 본 연구에서 제안한 개선된 SWoT 기반 모니터링 시스템을 직접 구현 및 시험하였다. 사용된 하드웨어의 제원은 Table 2와 같다.

Table 2. Hardware Specification of Testbed

항목	하드웨어 제원	수량
센서노드	AVR Atmega128L+ CC2420 802.15.4 전압, 온도, 조도 센서	3개 (싱크노드 1 포함)
SWoT 게이트웨이	삼성 S5PV210(CortexA8) CPU 512MB RAM 주기억장치 2GB SD card 보조기억장치 15MP CMOS 카메라 802.11 무선 LAN	1개
모니터링 시스템	팬택 IM-A910K 스마트폰 (안드로이드 버전 4.4.2)	1개

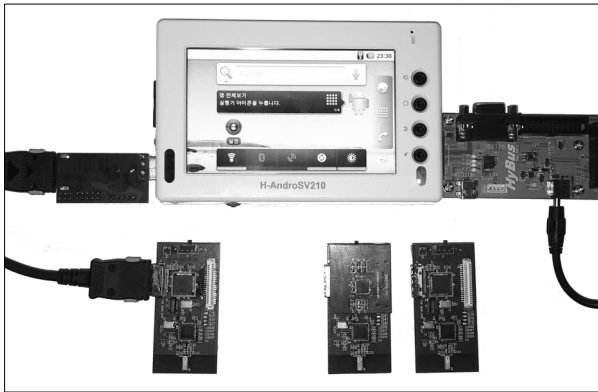


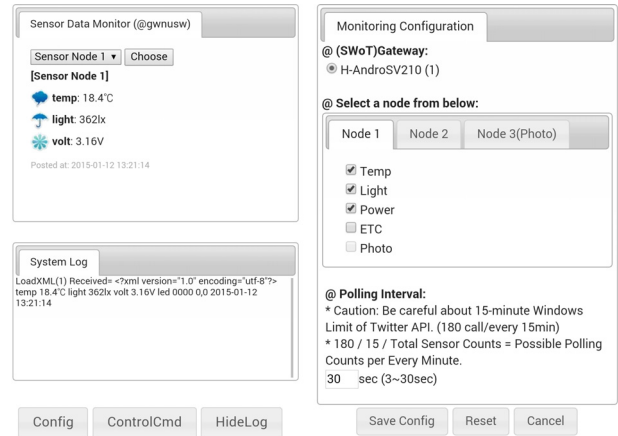
Fig. 14. Hardwares Used by Testbed

센서노드와 싱크노드 간의 통신은 802.15.4 Zigbee 프로토콜을 기반으로 한 무선으로 이루어지며, 싱크노드와 게이트웨이는 RS232 프로토콜을 통해 유선으로 연결된다. 이는 기존 USN 시스템의 구성과 동일하다. 게이트웨이는 802.11 무선 네트워크를 통해 인터넷에 연결되며, 모니터링 시스템인 스마트폰은 SWoT 게이트웨이와 서로 다른 서버넷 IP주소를 가진 802.11 또는 4G LTE 무선망을 통해 인터넷에 연결된다.

구현에 사용된 하드웨어의 외관 및 구성은 Fig. 14와 같다. 센서노드는 배터리로 구동되는 두 개의 센서노드와 게이트웨이에서 전원을 공급받는 한 개의 싱크노드로 구성되며, 임베디드 리눅스 OS를 실행하는 게이트웨이에는 하드웨어적 디버깅과 RS232 통신을 위한 전용 베이스 보드가 부착되었다. 모니터링 시스템은 안드로이드 기반 스마트폰을 사용했다.

상용 SNS 플랫폼으로는 트위터 서비스를 사용했으며, SWoT 게이트웨이의 ID는 @gwnusw로 생성하였다. 긴급 데이터의 전달을 위한 PUSH 메시지 서비스는 Google에서 제공하는 GCM 서비스를 사용하였다[11]. 모니터링 시스템은 JAVA와 PHP 언어를 이용하여 안드로이드 스마트폰용 웹-앱으로 구현하였으며, PHP 인터프리터의 동작을 위해 모니터링 시스템상에서 실행되는 임베디드 웹 서버 소프트웨어가 사용되었다.

모니터링 시스템의 인터페이스는 Fig. 15과 같이 구현되었다. 인터페이스는 크게 주 화면인 센서 데이터 모니터 화면과 모니터링 환경설정 화면, 장치 제어 윈도우로 나뉘어진다. 센서 데이터 모니터 화면에는 선택한 센서노드에 대한 실시간 센서 데이터 수치와 시스템 동작 로그가 나타나며, 하단부에는 부가기능의 실행 또는 타 인터페이스 화면으로의 이동을 위한 버튼들이 있다. 환경설정 화면에는 게이트웨이가 여러 개로 구성된 USN 시스템에 대비한 게이트웨이 선택 기능, SNS 타임라인 데이터에서 어떤 센서를 선택적으로 표시할 것인지를 결정하는 센서 종류 필터, SNS로부터 모니터링 프로그램이 수신할 JSON 데이터 수신주기(Polling 주기)가 있다.



(A) Main Screen (B) Configuration Screen

Fig. 15. Monitoring Mobile Application Interface

DM 발송을 위한 장치 제어 윈도우는 Fig. 16과 같이 구현되었다. 장치 제어 윈도우에서 센서노드를 선택하고 센싱 주기 설정과 사진자료 요청 명령을 DM을 통해 발송할 수 있으며, 사진(Photo) 자료는 센서노드 하드웨어의 한계로 인해 게이트웨이 후면 카메라에서 촬영 및 수신한다.

4.2 시험

개선된 모니터링 시스템을 바탕으로 SNS상의 데이터 수신주기를 30초 간격으로 설정하여 12시간 동안 관찰한 결과, 모니터링 기능이 중단 없이 정상적으로 동작함을 확인하였다. 임의의 긴급 데이터를 설정하여 SWoT 게이트웨이에서 모니터링 시스템으로 PUSH 메시지를 전달한 결과 정상적인 경고 동작을 확인하였다.

개선된 모니터링 시스템에서의 긴급 데이터 전달속도를 기존 시스템과 비교하여 측정한 결과를 Table 3에 수록하였다. 시험은 각 모니터링 시스템에 동일한 긴급 데이터를 수신받아 비교 측정하는 방법으로 진행되었으며, 긴급 데이터는 임의의 시각에 10회 발생시켰다. 게이트웨이와 모니터링 시스템의 시계는 NTP로 동기화하였으며, 긴급 데이터에 밀리초 단위의 송신시각을 포함하여 전송하였다.

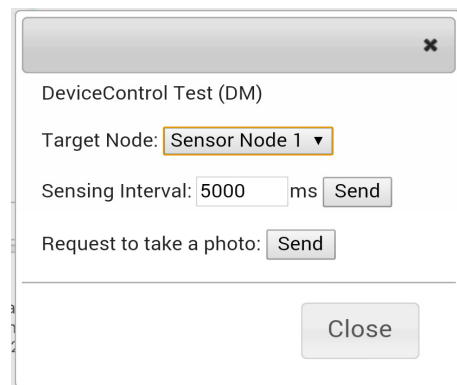


Fig. 16. Device Control Window(Send a DM)

Table 3. Average Delay in Receiving Emergency Data

기존 시스템 (Web 기반)	제안한 시스템 (PUSH 기반)	차이
3,744ms	2,584ms	+116ms

기존 시스템의 경우 긴급 데이터 전달 속도는 데이터 수신 주기에 따라 차이가 많이 날 수 있으며, 일반적인 설정은 30초이나 본 실험에서는 기존 모니터링 시스템에서 가능한 최소한의 측정주기인 5초로 설정하였다.

기존 모니터링 시스템과 제안한 시스템을 동일한 환경에서 성능측정한 결과, 제안한 시스템이 긴급 데이터 전달속도 면에서 평균적으로 더 빠름을 확인하였다.

또한, 부가기능의 활용예로 제시한 DM을 통한 제어명령 전송과 수신, 사진 촬영 및 포스팅 기능 역시 올바르게 동작 가능함을 확인하였다.

### 5. 결 론

본 연구에서는 SNS를 기반 플랫폼으로 활용하여 모니터링 시스템과 센서 네트워크가 서로 통신하는 SWoT 개념의 USN 실시간 모니터링 시스템을 개선, PUSH 메시지를 통한 응급 상황 처리기능과 기간별 통계 산출, 센서노드 장치 제어, 이미지 데이터의 저장 및 표출을 추가적인 전용 프로토콜이나 서버 구현 없이 제공하는 방법을 제안하였다.

이러한 개선사항으로 기존 모니터링 시스템보다 비용 효율적이고 신속하며 다양한 부가기능을 제공할 수 있었으며, 앞으로 추가적인 연구를 통해 SNS 친구관계를 바탕으로 한 논리적인 센서 네트워크 구현, SNS 서비스의 정보 전파능력 및 친구관계 형성을 활용한 여러 센서 네트워크 및 모니터링 시스템 간의 다양한 데이터 전파 및 공유 기능 등을 구현할 예정이다.

본 연구를 바탕으로 추후 USN 모니터링 분야에서 개선된 SWoT 기반 실시간 모니터링의 적용과 확산이 기대되는 바이다.

### References

[1] Chang-Hee Lee, Seok-Chan Jeong, Young-Seok Ock, and Min-Soo Kim, "Development of Water Quality Monitoring System using USN," *The Journal of Korean Institute of Information Technology*, Vol.10, No.8, pp.153-163, 2012.

[2] Chung et al., "Social Web of Things: A Survey," *International Conference of Parallel and Distributed Systems*, 2013.

[3] Dominique et al., "Sharing Using Social Networks in a Composable Web of Things," *Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 8th IEEE International Conference, pp. 702-707, 2010.

[4] Dae-Hyun Ryu, "Development of Urban Farm Management System using Commercial SNS as IoT Platform," *The Journal of the Institute of Internet, Broadcasting and Communication*, Vol.13, No.5, 2013.

[5] M. C. Domingo, "A context-aware service architecture for the integration of body sensor networks and social networks through the IP multimedia subsystem," *IEEE Communications Magazine*, Vol.49, No.1, pp.102-108, 2011.

[6] Woojin Joe, Meilan Jiang, and Karpjoo Jeong, "An M2M/IoT based Smart Data Logger for Environmental Sensor Networks," *Journal of KIISE: Computing Practices and Letters*, Vol.20, No.1, pp.1-5, 2014.

[7] Yacine et al., "A Social Web of Things Approach to a Smart Campus Model," *Green Computing and Communications (GreenCom), IEEE and Internet of Things (iThings/CPSCoM), IEEE International Conference on and IEEE Cyber, Physical and Social Computing*, pp.349-354, 2013.

[8] Chung et al., "Design and implementation of light-weight smart home gateway for Social Web of Things," *Ubiquitous and Future Networks (ICUFN), Sixth International Conference*, pp.425-430, 2014.

[9] M. Baqer, "Enabling collaboration and coordination of wireless sensor networks via social networks," *6th IEEE International Conference on Distributed Computing in Sensor Systems Workshops (DCOSSW)*, pp.1-2, 2010.

[10] Twitter inc., REST APIs Documentation [Internet], <https://dev.twitter.com/rest/public>.

[11] Google inc., Google Cloud Messaging (GCM) for Android [Internet], <https://developer.android.com/google/gcm/index.html>.



#### 유 명 한

e-mail : mhyoo@gwnu.ac.kr

2010년 강릉원주대학교 컴퓨터공학과 (학사)

2011년 강원대학교 전자공학과(석사)

2012년 강릉원주대학교 컴퓨터공학과 박사수료

관심분야: IoT, USN, Wireless Networking



#### 김 상 경

e-mail : skkim98@gwnu.ac.kr

1985년 고려대학교 전자공학과(학사)

1987년 고려대학교 전자공학과(석사)

2002년 고려대학교 전자공학과(박사)

1987년~1989년 삼성전자 주임연구원

1989년~2004년 KT 선임연구원(부장)

2004년~현 재 강릉원주대학교 컴퓨터 공학과 교수

관심분야: Wireless Network Protocol, D2D Communications, IoT, Advanced Network Architecture