

센서태그 통합 데이터 필터링에 관한 연구

종신회원 류 승 완*, 준회원 오 슬 기*, 정회원 박 세 권*, 준회원 오 동 옥*

Cooperative Data Stream Filtering for Sensor Tag

Seung-Wan Ryu* *Lifelong Member*, Seul-ki Oh* *Associate Member*,
Sei-kwon Park* *Regular Member*, Dong-ok Oh* *Associate Member*

요 약

센서 태그의 데이터는 태그 정보와 센싱 정보를 동시에 가지며 미들웨어 또는 상위 레벨에서의 필터링 및 가공이 필요하다는 특징을 가지고 있다. 기존의 필터링 알고리즘에서는 태그데이터와 센서 데이터를 각각 필터링하는 알고리즘이 주로 제안되었다. 그러나 센서 태그의 사용 요구는 점차 증가하고 있으며, 사용요구에 적합한 필터링을 위해서는 센싱 데이터와 RFID 데이터를 통합 처리할 수 있는 새로운 필터링 알고리즘이 필요하다. 본 논문에서 제안하는 필터링 알고리즘에서는 각 태그의 시간 축에 대한 필터링만을 고려하는 것이 아니라 공간적으로 근접한 태그의 데이터도 함께 고려하여 필터링하여 오류 및 이벤트 검출의 정확성을 향상시키고 데이터의 대표값 저장으로 데이터 저장에 필요한 비용을 감소시킬 수 있다.

Key Words : RFID/USN, Data Filtering, Sensor Tag, EPCGlobal

ABSTRACT

The conventional sensor tag data filtering algorithm uses time window based data filtering for each tag data. However, this approach shows many performance problems such as low error and event detection rate and larger storage size requirement. In this paper, we propose a collaborative sensor tag data filtering algorithm to improve sensor data processing performance. simulation study shows that the proposed sensor tag filtering algorithm outperforms the conventional filtering algorithm in terms of the processing time, the size of required data storage memory and accuracy of error and event detection rate.

I. 서 론

RFID(Radio Frequency Identification) / USN(Ubiquitous Sensor Network)은 유비쿼터스 컴퓨팅의 핵심 기술로서 EPCGlobal에서는 RFID 태그를 태그의 기능 및 구성에 따라 Class-1부터 Class-4로 구분된다.^[10] 다양한 태그 중에서 센서가 포함된 Class-3 센서태그는 태그 데이터와 센서 데이터를 동시에 수집하여 미들웨어로 전달하기 때문에 제품 인식 및 환경 정보 수집이 동시에 가능하기 때문에 다양한 분야

에서 관심을 갖고 있다.

하지만 센서태그의 경우 미들웨어에서 두 유형의 데이터를 동시에 처리해야 하지만 기존 RFID 및 USN 미들웨어에서는 이러한 센서 태그 정보를 동시에 처리하지 못하는 한계를 나타내고 있다. 특히 데이터 필터링 부분에서는 RFID 태그데이터 뿐 아니라 내장된 센서에서 수집된 데이터까지 함께 저장 및 전달되기 때문에 기존 RFID 데이터 필터링 혹은 USN 데이터 필터링을 위한 알고리즘을 이용하여 센서태그의 데이터를 필터링하기에는 문제가 있다.

* 본 연구는 2011년도 중앙대학교 연구장학기금 지원에 의한 것임.

* 중앙대학교 정보시스템학과(ryu@cau.ac.kr, osk_1004@naver.com psk3193@cau.ac.kr, jirox7@hanmail.net)

논문번호 : KICS2011-05-234, 접수일자 : 2011년 5월 31일, 최종논문접수일자 : 2011년 7월 14일

따라서 본 논문에서는 기존의 데이터 필터링 알고리즘이 가지고 있는 문제점을 해결하고 센서태그 정보의 효율적인 사용 및 센서태그 정보의 통합처리를 위한 통합 필터링 알고리즘을 제안한다. 제안하는 알고리즘은 센서태그에서 수집되는 센서데이터를 시간축을 기준으로 비교하여 각 센서태그의 대표 값을 결정하고, 동일한 위치에 있는 센서태그들을 하나의 그룹으로 구성하여 동일 그룹에 속한 태그들의 대표 값을 비교하여 센서태그의 오류 및 이벤트 발생에 대한 검출률을 향상시켜 센서데이터의 효율성을 증대시킬 수 있다. 또한 그룹에 대한 대표값을 데이터베이스에 저장함으로써 데이터 저장공간을 절약할 수 있게 된다.

본 논문은 다음과 같이 구성된다. 1절에서는 센서태그의 개요와 기존 RFID 데이터 필터링, USN 데이터 필터링, RFID/USN 데이터 필터링에 대해 각각 살펴보고 기존의 필터링 알고리즘이 가진 한계에 대해 살펴본다. 2절에서는 RFID/USN 통합 데이터를 위한 새로운 필터링 알고리즘을 제안하고, 3절에서는 알고리즘의 결과를 분석하기 위한 시뮬레이션 환경 및 결과를 살펴본다. 마지막 본 논문의 연구에 대한 결론을 제시한다.

II. 기존 연구 및 한계점

2.1 센서태그의 개요

센서태그는 기존의 EPCGlobal의 Class-3 태그에 해당하며, 기존의 ID 정보 획득만 가능한 수동형 태그에 외부 환경정보를 습득할 수 있는 센서와 자체 전원 공급을 위한 박형 전지(Film Battery)로 구성되어있다. 센서태그는 기존의 수동형 태그의 단점인 짧은 인식 거리, 낮은 인식률과 신뢰성을 보완할 수 있고, 온도/습도 등의 다양한 외부 환경정보를 수집하여 제공하는 것이 가능하다. 따라서, 센서태그는 물류과정에서 혈액 및 식의약품과 같은 환경에 민감한 물품의 상태를 모니터링하거나 환경 관리 분야 등에 다양하게 응용될 수 있다.

센서태그는 주변 상황 정보를 수집하는 센싱 기능을 가지고 있으며, 센싱된 값을 태그 내부메모리에 저장하고, 센싱된 데이터가 축적되면 주기적으로 시스템에 보고하게 된다. 또한, 센서태그는 RFID 태그가 지니는 이동성의 특징을 가지므로 태그가 리더의 인식 범위에서 벗어나는 경우에도 센싱된 데이터를 RFID 태그의 사용자 메모리에 저장하게 된다.

센서태그의 사용이 증가함에 따라서 태그 데이터와 센서 데이터의 통합처리를 위한 미들웨어의 필요성이

대두되었다. 이를 위하여 RFID 미들웨어에 센서데이터 처리를 위한 모듈을 추가함으로써 센서 데이터의 처리를 지원하는 미들웨어가 설계되고 있으며, 사용자가 입력한 범위를 벗어나는 센서 데이터를 감지하는 수준의 필터링과 센싱값의 이상유무를 보고하는 모듈이 추가된 미들웨어가 제안되었다^{2,6)}.

이러한 미들웨어에서 사용하는 필터링 알고리즘은 RFID 미들웨어에서 센서데이터를 처리하는데 중점을 두었을 뿐 센서태그 데이터의 특징을 고려하지 않았기 때문에 센서태그의 정보를 효과적으로 처리할 수 없다는 한계를 가지고 있다.

본 논문에서는 기존 RFID 데이터 필터링과 센서데이터 필터링 기법의 특징과 장단점을 살펴본 후 센서태그 정보의 효율적 처리, 저장 및 응용환경에 따른 다양한 데이터 처리과정을 지원할 수 있는 센서태그 데이터 필터링 기법을 제안한다.

2.2 RFID 데이터 필터링

RFID 데이터 필터링은 태그 데이터의 신뢰도 향상을 위한 필터링이나 저장 공간 확보를 위한 필터링, 태그 발생에 대한 이벤트 필터링을 목적으로 하고 있다. 태그 데이터의 신뢰도 향상을 위한 필터링에는 일정 시간 동안 태그로부터 데이터를 수집하고 시간축을 중심으로 데이터를 처리하는 슬라이딩 윈도우를 이용하여 유효한 데이터를 검출하는 방법이 있다¹⁴⁾.

저장공간 확보를 위한 필터링은 태그데이터의 중복을 최소화하기 위한 데이터 정제작업이 선행되며, 태그데이터의 시간 유형에 따라 데이터 저장공간을 달리하여 쿼리에서 요구하는 시간에 따라 검색하는 데이터베이스를 달리하거나, 태그데이터를 그룹핑하여 데이터의 중복저장을 최소화하는 방식 등이 있다^{7,9,11)}. 이벤트 필터링에서는 태그의 인식 시간 및 인식 순서를 분석하여 사용자가 사전에 작성한 이벤트 폴에서 적합한 이벤트를 인지하는 방식이 있다^{3,13)}.

RFID 필터링의 경우 태그의 신뢰도 등 태그 데이터 처리에 초점이 맞춰져 있고, 슬라이딩 윈도우는 시간축을 중심으로 구성되었기 때문에 센서 데이터 분석을 위해 이용하기에는 한계를 가지고 있다.

2.3 USN 데이터 필터링

USN 데이터 필터링의 목적은 노드간의 데이터 통신량을 감소시켜 노드의 전력소모를 감소시키기 위한 필터링과 어플리케이션 레벨에서 필요한 데이터를 추출하기위한 필터링으로 구분할 수 있다.

노드간의 데이터 통신량을 감소시켜 소모 전력을

최소화 시키는 것을 주 목적으로 하고 있는 필터링의 경우 센서 네트워크에서는 2-tier 이상으로 구성된 경우가 많기 때문에 상위 레벨로 데이터를 전송하기 전에 대표 노드에서 주변 노드의 데이터를 수집하여 전송할 데이터를 압축시키는 방식이 주로 사용되고 있으며, 센서 노드에서 데이터 필터링이 수행되기 때문에 복잡한 필터링 알고리즘은 센서노드의 전력 소모량을 증가시키므로 최소한의 전력을 소모하는 단순한 형태의 필터링 알고리즘이 사용된다⁴¹⁾. 그러나 노드에서 단순한 알고리즘을 이용하여 데이터를 압축시키는 경우 데이터의 정확도가 낮아지기 때문에 데이터 정확도 향상을 위해 센서데이터에 따라 허용 오류 한계를 변경하는 연구가 진행되고 있다⁴²⁾.

어플리케이션에서 수행되는 센서 데이터 필터링은 사용자가 입력한 범위를 벗어나는 센서 데이터를 감지하거나, 데이터베이스에서 필요한 데이터를 수집하기 위한 데이터 마이닝 기법이 주로 사용되고 있다⁸⁾.

USN 데이터 필터링의 경우에는 대부분의 필터링이 센서노드에서 수행되기 때문에 알고리즘이 단순해지고, 어플리케이션에서 수행되는 데이터 필터링 역시 기초적인 데이터 필터링이 주를 이루기 때문에 효율적으로 센서 데이터를 분석하기 어렵다. 센서태그 데이터 처리를 위해서는 미들웨어 레벨에서 깊이 있는 센서 데이터에 대한 분석이 수행될 필요가 있다.

2.4 센서태그 데이터 필터링

RFID 데이터 필터링과 센서데이터 필터링은 목적과 적용환경이 상이하므로 센서태그를 위한 데이터 필터링을 위해 두 가지 필터링을 동시에 적용할 수는 없다. RFID 데이터 필터링에서는 태그데이터의 진의 여부에 대한 필터링과 데이터 통합을 통한 저장공간 확보, 태그 발생에 대한 이벤트 필터링을 주 목적으로 하고, 센서 필터링에서는 주로 센서노드에서 수행하며 데이터정제를 위한 필터링이 아닌 통신 전력 절감 목적으로 필터링을 수행한다.

또한 센서태그 데이터 필터링은 RFID 미들웨어에 센서 데이터 처리모듈을 추가하여 데이터를 처리하지만 사용자가 사전에 입력한 범위 내에서만 데이터가 처리되기 때문에 사용자가 입력하지 못한 상황에 대한 대처가 부족하며, 센서데이터 저장 공간에 대한 고려가 부족하다.

표 1에서는 기존 필터링방법인 RFID 데이터 필터링 기법과 센서데이터 필터링 기법, 센서태그 데이터 필터링 기법들의 목적, 특징과 한계를 보여주고 있다. 표 1에서 제시하는 바와 같이 기존의 RFID 혹은 센서

표 1. 데이터 필터링의 목적 및 한계

	목적	한계
RFID 데이터 필터링	- 태그 데이터의 신뢰성 향상 - 저장공간 확보 - 쿼리 처리 성능 향상 - 이벤트 검출	- 데이터 인식 여부에 대한 필터링
USN 데이터 필터링	- 노드간 데이터 통신량 감소 - 노드 생존 시간 증가 - 어플리케이션 레벨에서의 데이터 추출	- 단순한 센서 데이터 필터링 - 태그 데이터와 센서 데이터의 연계 부족
RFID/USN 데이터 필터링	- RFID 미들웨어를 이용한 센서 데이터 처리 - 사용자 요구사항 기준의 필터링	

데이터 필터링 기법은 센서태그 데이터의 효율적인 처리를 위한 직접 적용이 불가능하므로 센서태그의 데이터를 효과적으로 처리하고 저장할 수 있는 새로운 데이터 필터링 기법이 요구되고 있다.

이를 위해 본 논문에서는 센서 데이터의 처리를 개선하여 센서데이터와 태그데이터의 연계를 강화하고, 강화된 센서데이터의 분석을 통해 문제 발생에 대한 신속한 감지가 가능한 데이터 필터링 알고리즘을 제안하고자 한다.

III. 통합 데이터 필터링 알고리즘

본 논문에서는 센서태그에서 수집된 센싱 데이터를 통합 처리하여 필요한 저장 공간의 크기를 줄이고, 유사한 기능과 적용환경을 갖는 태그들을 단위 그룹으로 설정하여 각 그룹의 센서 데이터를 시간적 측면의 데이터 변화와 인접 태그 그룹의 공간적 측면의 두 가지 측면에서 비교하는 복합 필터링 방식을 제안하고자 한다.

복합 필터링 방식은 크게 3단계로 구성된다. 1단계는 RFID 데이터 필터링 단계로 RFID 태그 데이터의 중복되는 데이터를 제거하고 각 태그의 위치정보를 분석하게 된다. 2단계는 시간축 센서 데이터 필터링 단계로 개별 센서태그에 축적된 데이터들의 시간적 측면의 변화추이를 분석하여 각 태그별 대표 센싱값을 설정하는 단계이다. 3단계는 그룹별 센서 데이터 필터링 단계로 동일 그룹에 속해있는 태그들의 사용 환경의 특징을 반영하여 각 그룹내의 센싱 데이터 분석을 통해 그룹별 대표값을 결정하고 저장한다.

3.1 RFID 데이터 필터링

1단계인 RFID 데이터 필터링에서는 위치와 사용

환경이 같은 이웃한 태그들을 하나의 그룹으로 분류하고 Group_ID를 부여한다. 각 그룹이 특정장소에 들어오는 시간과 나가는 시간을 같은 row에 함께 저장함으로써 데이터의 중복저장을 피하고 저장공간을 절약한다. 각각의 태그가 최종적으로 속한 그룹의 ID를 별개의 테이블에 저장하여 태그와 그룹 ID의 매핑이 용이하도록 한다.

그룹 ID는 최초 인식된 시점에서의 위치를 기본으로 순차적으로 부여하게 되며, 그룹 내의 태그가 나누어 이동하게 될 경우에는 (기존의 Group_ID + 새로운 Group_ID)의 형태로 Group_ID를 결정한다.

태그데이터는 Tag Table과 Location Table에 나누

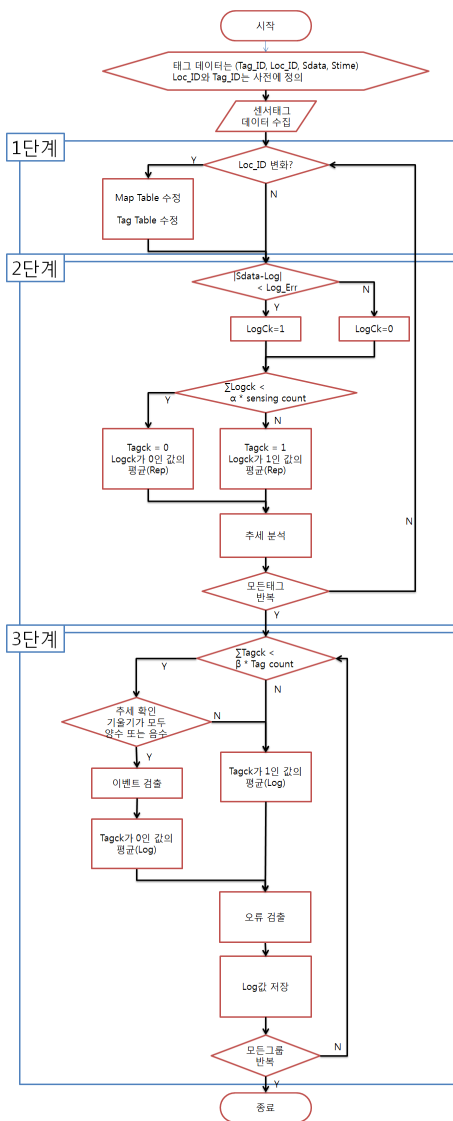


그림 1. 통합 데이터 필터링 알고리즘 순서도

어 저장되게 된다. Location Table에는 Group_ID, Loc_ID, Time_In, Time_Out이 저장되며, Tag Table에는 Tag_ID와 태그가 속한 Group_ID가 저장된다. Group_ID는 과거의 Group_ID도 포함되어 있기 때문에 그룹 ID에 대해서 살펴보면 Tag_ID의 이동 경로에 대해 쉽게 확인할 수 있다.

3.2 시간축 센서 데이터 필터링

2번째 단계는 각 태그에 저장된 센서 데이터를 시간축을 기반으로 필터링하여 각 태그의 센싱 대표값을 결정한다. 이 단계에서는 현재의 센서데이터 값을 $S_i, i \geq 0$ 라고 하면 이 값을 과거 n 회의 센서 데이터 평균값인 $\bar{S}_n = \sum_{x=i-n}^{i-1} S_x/n, n \geq 1$ 와 비교하여 현재 센싱된 데이터의 유효성을 검사한다. 이 경우 S_i 와 \bar{S}_n 의 차이가 미리 정한 오차범위인 ϵ_i 이 아닌 경우 유효성 지표인 Tag_ck 값을 1로 설정하고 그렇지 않은 경우 0으로 설정한다. 즉,

$$Tag_ck = \begin{cases} 1, & \text{if } |S_i - \bar{S}_n| < \epsilon_i \\ 0, & \text{otherwise} \end{cases}$$

또한 하나의 센서태그에는 내부메모리에 k 개의 센싱값이 누적되어 있으므로 만약 유효성있는 센서 데이터의 수가 전체 센서 데이터에 대해 $\alpha(0 \leq \alpha \leq 1)$ 이상의 비율을 나타내면 센서 데이터는 유효한 것으로 판단한다. 이 경우 센서태그에 있는 k 개의 센서 데이터의 대표값은 Tag_ck 가 1로 설정된 센싱데이터들의 평균값이 그 태그의 대표 센싱값으로 설정된다.

만약 유효성 있는 센서 데이터의 비율이 αK 개 이하이면 전체 k 개 데이터의 유효성은 0이 되고 대표 센서 데이터의 값은 유효성이 없는 데이터들의 평균값이 된다.

k 개의 센싱데이터의 대표값은 $S_{(K)}$ 이고, K 개의 값 중 유효성 있는 센서 데이터의 수는 $Tag_ck_{(K)} = \sum_{k=1}^K Tag_ck_{(k)}$ 이고, K 개 센서 데이터의 대표 유효성을 $Tag_{(K)}$ 라 정의한다.

만약 $Tag_ck_{(K)} \geq \alpha K$ 이면 $Tag_{(K)} = 1, S_{(K)} = \sum_{k=1}^K s_k \delta(Tag_ck_{(k)}, 1) / Tag_ck_{(K)}$ 이고, $Tag_ck_{(K)} < \alpha K$ 이면 $Tag_{(K)} = 0$ 라하면

$$S_{(K)} = \sum_{k=1}^K s_k \delta(Tag_ck_{(k)}, 0) / Tag_ck_{(K)}$$

이 된다. 여기

서 $\delta(i, j)$ 는 $i = j$ 이면 1이고 그렇지 않으면 0이 되는 지표함수이다.

각 태그의 대표값을 결정한 뒤 각 태그의 데이터 변동 폭을 살펴보게 된다. 데이터 추세 분석은 다양한 예측 이론 중 추세분석법(Trend Analysis)을 이용하도록 한다. 추세분석법이란 시계열을 잘 관통하는 추세선을 구하여 미래 수요를 예측하는 방법으로 본 논문에서는 추세선을 모두 구하지 않고, 추세선의 기울기만을 이용하여 데이터 변동의 방향성을 감지할 수 있도록 한다. 추세선의 기울기를 구하는 공식은 다음과 같다.^[15]

$$trend = \frac{n \sum_{t=1}^n t Y_t - \left(\sum_{t=1}^n Y_t \right) \left(\sum_{t=1}^n t \right)}{n \sum_{t=1}^n t^2 - \left(\sum_{t=1}^n t \right)^2}$$

여기서 $trend$ 는 시계열 추세선의 기울기, n 은 데이터의 수, t 는 시간, Y_t 는 시간 t 의 실제 데이터를 뜻한다. 위 과정은 모든 태그들의 대표값과 $trend$ 가 결정될 때까지 반복되며, α 값은 0.1~1.0 사이의 값으로 시뮬레이션을 통해 최적의 값을 구하도록 하며 본 논문에서는 0.9로 설정하였다.

3.3 그룹 별 센서 데이터 필터링

3 단계에서는 동일 그룹에 속한 태그들의 대표 센싱값을 비교하여 특정 태그의 이상 유무와 이벤트 발생을 감지한다. 만약 동일그룹 내에서 이상이 있는 태그수의 비율이 전체 태그 수에 대해 β ($0 \leq \beta \leq 1$) 이상이고, 이러한 태그들의 센싱데이터 값의 추세가 동일한 형태를 나타내면 이벤트가 발생한 것으로 판단된다. 즉 그룹 G 에 I 개의 센서태그가 소속되어 있을 경우, 이상이 있는 태그의 집합은 I_t^G 이고, 이러한 태그의 수는 $E_{(G)} = \sum_{i=1}^I \delta(Tag_{(K)}, 0)$ 가 된다. 만약 $E_{(G)} \geq \beta I$ 이고, 이상이 있는 태그들의 트렌드 T_i^G 가 모두 같다면 이벤트가 발생한 것으로 판단한다. 이벤트 발생 시 $S_{(K)_i}$ 의 평균값인 $S_G = \sum_{i=1}^I S_{(K)_i} \delta(Tag_{(K)_i}, 0) / E_{(G)}$ 가 그룹 G 의 대표 센싱 값이 된다. $E_{(G)} = \beta I$ 인 경우에는 $S_G = \sum_{i=1}^I S_{(K)_i} \delta(Tag_{(K)_i}, 1) / E_{(G)}$ 가 대표 센싱 값이 된다. β 값은 0.1~1.0 사이의 값으로 시뮬레이션을 통해 최적의 값을 구하도록 하며 본 논문에서는 최적 성능

표 2. 시뮬레이션 환경

플랫폼	Windows7
사용 언어	JAVA
DBMS	MySQL
통신횟수	144회
K(센싱횟수)	10개
I(태그갯수)	10~100개
이벤트 유형	4종
오류 유형	1종

을 제공하는 0.6으로 설정하였다.

IV. 성능 평가

4.1 시뮬레이션 환경

본 논문에서 제안하는 센서데이터 필터링 기법의 성능 평가를 수행하기 위해 다음과 같은 실험환경을 가정하였다. 각 센서태그에서 수집되어 미들웨어로 전송하는 모든 데이터는 충돌 및 오류 없이 미들웨어로 전달된다고 가정한다. 실제 운용환경에서는 다수의 태그가 리더기와 통신할 경우 충돌이 발생하며, 이를 해결하는 충돌 방지 및 해결 기법들이 많이 제안되었다.^[5,12] 그러나, 본 논문에서는 이러한 충돌방지 및 해결은 이미 리더기와의 통신단계에서 해결된 것으로 가정하였으므로 전달된 데이터는 오류 없이 전달된 것으로 가정한다.

모든 센서태그는 최대 K 개의 센서 데이터를 내장 메모리에 저장하고 내장메모리의 내용을 주기적으로 리더기에 송신한다. 본 논문에서는 센서태그는 1분단위로 환경정보를 센싱하고 최대 10개의 ($K=10$) 데이터를 저장할 수 있다고 가정한다. 따라서 센서태그는 10분단위로 리더기와 통신한다.

본 논문에서 센서데이터는 환경정보 중 가장 빈번하게 사용되는 온도데이터를 가정하였으며, 각 센서에서는 이벤트 4종 및 오류 1종이 발생하는 것으로 가정하였다. 오류는 개별 태그에 독립적으로 발생하는 태그 고장으로 가정하고 이는 센서 데이터의 오류를 야기하게 된다. 이벤트는 동일 그룹 내의 모든 태그 혹은 많은 태그의 데이터 센싱에 영향을 미치게 된다. 이러한 이벤트로는 특정 환경에서의 급격하거나 완만한 온도의 증가 및 감소의 네 가지로 가정하였다.

또한 각 센서태그는 센싱환경에 설치된 후 이동이나 제거가 발생하지 않는 환경을 가정하였다.

4.2 성과 지표

4.2.1 데이터 필터링 시간

데이터 필터링 시간은 RFID 데이터와 센서 데이터를 미들웨어에서 필터링하고, 데이터베이스에 저장하는 시간을 말한다. 필터링 시간이나 저장하는 시간이 태그와 리더간의 통신 간격보다 오래 걸릴 경우 데이터의 처리가 지연되어 실시간 시스템 구현이 어려워진다. 본 논문에서 제안하는 데이터 필터링 알고리즘은 비교하는 알고리즘에 비하여 센서 데이터를 처리가 증가하기 때문에 전체적인 데이터 필터링 시간은 증가할 것으로 생각되며, 태그의 수와 그룹의 수가 증가할수록 반복 작업이 증가하기 때문에 큰 폭으로 시간이 증가하게 될 것이다.

4.2.2 데이터 저장 공간

데이터 저장 공간은 RFID 데이터와 센서 데이터를 저장하기 위한 데이터베이스 공간을 말하며 RFID /USN 데이터 특성상 데이터에 대한 로그를 장시간 보관하여야 하며 그 데이터의 양도 방대하기 때문에 저장공간에 대한 부담이 매우 크다고 할 수 있다. 본 논문에서 제안하는 알고리즘에서는 센서 데이터를 그룹 단위로 대표값을 결정하여 저장하기 때문에 데이터 저장 공간은 큰 폭으로 감소할 것으로 생각되며 한 그룹에 속한 태그의 수가 많을수록 감소하는 저장 공간은 커지게 될 것이다.

4.2.3 이벤트 및 오류 검출 정확도

본 알고리즘의 가장 중요한 성과지표로 센서 데이터의 분석을 통한 오류 및 이벤트 발생 검출의 정확도를 분석한다. 정확도는 시뮬레이션에서 랜덤으로 발생시킨 이벤트의 횟수와 필터링을 통해 검출한 이벤트의 횟수를 비교함으로써 그 정확도를 비교하고자 한다. 본 논문에서 제안하는 알고리즘은 시간 뿐 아니라 공간에 대한 고려도 함께 하기 때문에 좀더 정확한 이벤트 검출이 가능할 것이다.

4.3 성능 평가 결과

4.3.1 데이터 필터링 시간

미들웨어에서의 데이터 처리가 복잡해질수록 데이터의 처리속도 역시 증가하며, 증가 폭은 태그의 수가 증가할수록 처리속도역시 크게 증가하게 된다. 단순 필터링에서는 대표값 결정 및 범위 확인 작업으로 끝나는데 반해 복합 필터링의 경우에는 각 태그별 추세 확인 및 동일 그룹 태그의 추세 비교 등 추가 작업을

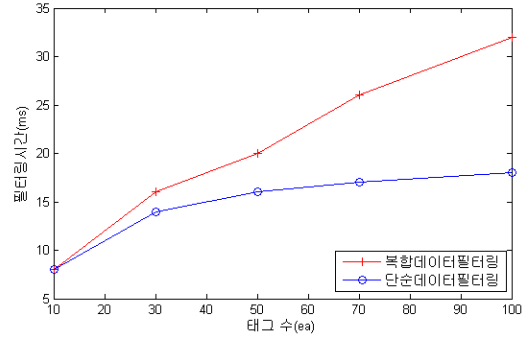


그림 2. 데이터 필터링 시간

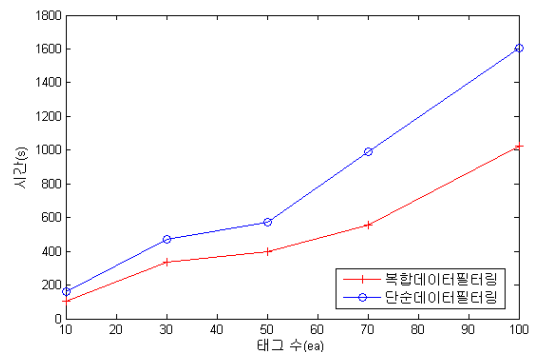


그림 3. 데이터 저장 시간

위한 시간이 많이 필요하다. 이러한 작업들은 태그가 증가할수록 작업량이 크게 증가하게 된다.

그러나 데이터 필터링을 통해 데이터 저장량이 감소하게 되면 데이터 저장 시간이 크게 감소하게 되는데, 이는 데이터 저장 시간의 경우 데이터베이스에 액세스하여 작업을 해야 하기 때문에 데이터 필터링 시간보다 더 오랜 시간이 걸리기 때문이다.

복합 필터링의 경우 복잡한 데이터 처리로 데이터 필터링 시간이 증가하게 되지만 전체 작업시간에서 데이터 저장 시간의 비율이 크기 때문에 저장되는 데이터를 감소시키는 것으로 작업시간의 향상을 볼 수 있으며, 빠른 작업 속도는 유비쿼터스 시스템의 특징인 실시간 시스템 구축에 매우 중요한 척도이기 때문에 핵심적인 성과지표라 할 수 있다.

4.3.2 데이터 저장 공간

일반적으로 태그의 개수가 증가하면 그에 대한 데이터양은 급격하게 증가하게 된다. 하지만 복합 필터링의 경우 센서 데이터를 그룹에 대한 대표값으로 저장하기 때문에 태그의 개수가 아무리 증가하더라도 저장되는 데이터의 수는 일정하다. 그렇기 때문에 태

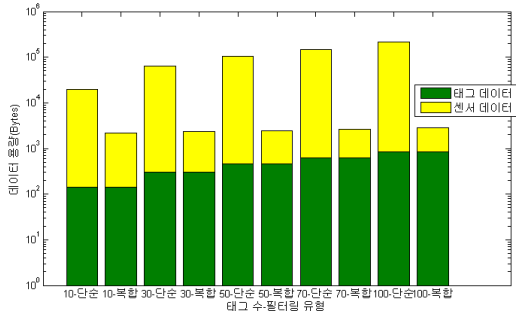


그림 4. 데이터 저장 공간

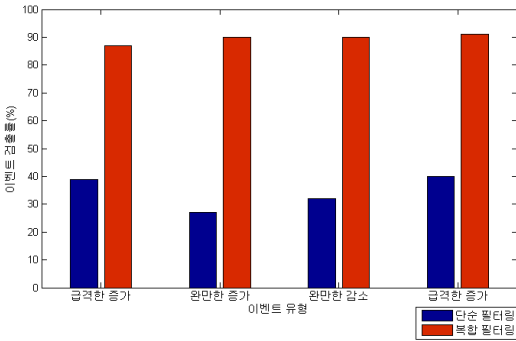


그림 5. 유형별 이벤트 검출 정확도

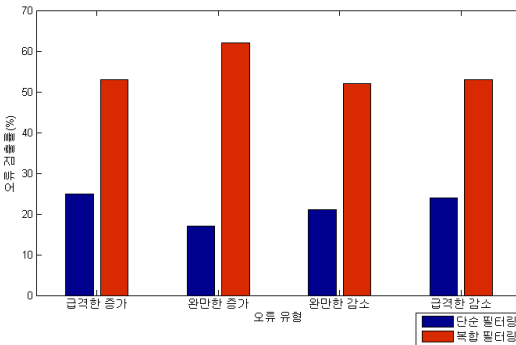


그림 6. 유형별 오류 검출 정확도

그의 수가 증가할수록 저장 공간의 효율이 더욱 증가하게 된다. 저장 공간의 효율은 로그데이터의 저장에 대한 부담을 줄여줄 수 있어 시스템 구축비용의 저장 공간에 대한 비중을 줄여줄 수 있다.

4.3.3 이벤트 및 오류 검출 정확도

현재 사용되고 있는 센서데이터 필터링에서는 센서 데이터가 일정 범위 내에 존재 하는지 벗어났는지만을 비교하기 때문에 센서태그의 환경에 대한 정확한 분석이 어려웠고, 모든 센서에서 이상을 감지하기 전

에는 정확한 오류를 인식하지 못하기 때문에 완만한 데이터 변화에는 쉽게 인식하지 못하였다.

그러나 복합 필터링에서는 시간축에 대한 비교 뿐 아니라 동일 시간대의 동일 그룹에 대한 데이터 분석과 데이터 추세분석도 함께 수행하기 때문에 오류나 이벤트 발생에 대해 보다 정확한 검출이 가능하다. 센서태그를 이용한 시스템에서 이벤트 발생 및 오류 검출의 정확도는 문제 발생 시 신속한 대처가 가능해진다는 장점이 될 수 있다.

V. 결 론

RFID/USN 기술은 유비쿼터스 컴퓨팅 환경을 구현 하는데 핵심 기술이라 할 수 있으며 센서 태그는 태그 정보와 센싱 정보를 효과적으로 수집할 수 있어 그 활용도가 매우 넓기 때문에 센서태그의 특징을 고려한 데이터필터링이 필요하다. 본 논문에서 제안하고 있는 통합 데이터 필터링의 성능평가 결과를 살펴보면 이동을 고려하지 않은 상태에서는 태그의 수가 많아질 수록 데이터 필터링 시간과 데이터 저장 공간에 대한 큰 이득을 얻을 수 있었다. 뿐만 아니라 이벤트 혹은 오류 검출에 대한 정확도가 향상되어 더욱 다양한 서비스에 응용될 수 있다.

본 논문에서는 태그 이동에 대한 고려가 없는 상태에서의 성능평가를 수행하였으며, 향후 연구에서는 센서 태그의 이동을 고려한 데이터 필터링에 대한 연구를 통해 본 논문에서 제안하는 통합 데이터 필터링이 실제 환경에서 얻을 수 있는 성능에 대한 연구가 필요하다.

참 고 문 헌

- [1] 김진수, 박찬흠, 김종근, 강병욱, “센서 네트워크에서 계층적 필터링을 지원하는 에너지 효율적인 데이터 집계연산,” 한국컴퓨터정보학회 논문집 제12권 제1호 pp.79-88, 2007.
- [2] 김현우, 류우석, 홍봉희, “능동형 센서 태그를 지원하는 미들웨어 플랫폼 설계,” 한국정보과학회 2009 한국컴퓨터종합학술대회논문집 Vol.36 pp.186-191, 2009.
- [3] 노진석, 복경수, 유재수, “전처리 필터링을 위한 효율적인 RFID 비즈니스 이벤트 검출 기법,” 정보과학회논문지-데이터베이스 제35권 제2호 pp.143-154, 2008.
- [4] 박노준, 현동준, 김명호, “센서 네트워크에서 집

계연산을 위한 적응형 필터링,” 정보과학회지 제 32권 제4호 pp.372-382, 2005.

[5] 이수련, 이채우, “채널 모니터링 기법을 이용한 RFID 리더 충돌방지 알고리즘,” 전자공학회논문지 제43권 TC편 제8호 pp.35-46, 2006

[6] 이준호, 류우석, 홍봉희, “센서 태그 데이터의 필터링을 위한 Edge Manager의 설계,” 한국정보과학회 2008 종합학술대회논문지 제35권 제1호 pp.138-143, 2008.

[7] 윤홍원, “RFID 데이터 스트림의 효율적인 필터링 기법,” 한국콘텐츠학회논문지 제7권 제10호 pp.27-35, 2007.

[8] 정재은, “센서 네트워크의 데이터 스트림 마이닝을 위한 온톨로지 기반의 전처리 기법,” 지능정보연구 제15권 제3호 pp.67-80, 2009.

[9] D. Lin, et al., “Data Management in RFID Applications,” Lecture Note in Computer Science, 2007, Vol.4653/2007, 2007.

[10] EPCGlobal, “Tag Class Definitions 1.0,” EPCGlobal Whitepaper, 2007.

[11] H. Gonzalez, et al., “Warehousing and analyzing massive RFID data sets,” Data Engineering, 2006. ICDE '06. Proceedings of the 22nd International Conference, 2006.

[12] J. Cha and J. Kim, “Novel Anti-collision Algorithms for Fast Object Identification in RFID System,” 11th International Conference on Parallel and Distributed Systems - Workshops - Vol.2, 2005.

[13] J. Xingyi, et al., “Efficient Complex Event Processing over RFID Data Stream,” Computer and Information Science, 2008. ICIS 08.seventh IEEE/ACIS International Conference pp.75-81, 2008.

[14] Y. Bai, F. Wang and P. Liu, “Efficiently Filtering RFID Data Streams,” CleanDB Workshop, 2006.

[15] 김세현, “현대경영과학-한글선형정수계획법 소프트웨어 K-opt활용,” 무역경영사, 2008.

류 승 완 (Seung-Wan Ryu)

중신회원



1988년 2월 고려대학교 산업공학과 졸업
 1991년 2월 고려대학교 산업공학과 석사
 2003년 뉴욕주립대(SUNY at Buffalo) 산업공학과 박사
 1991년~1993년 LG전자 영상미디어 연구소(주임연구원)
 1993년~2004년 ETRI 이동통신연구단 선임연구원
 2004년~현재 중앙대학교 정보시스템학과 교수
 <관심분야> 이동통신시스템 설계 및 성능 분석

오 슬 기 (Seul-Ki Oh)

중회원



2010년 2월 중앙대학교 정보시스템학과 졸업
 2011년 3월~현재 중앙대학교 정보시스템학과 석사과정
 <관심분야> 이동통신, D2D, 무선통신 시스템, RFID/USN

박 세 권 (Sei-Kwon Park)

정회원



1981년 2월 서울대학교 산업공학과 석사
 1985년 2월 Texas A&M 산업공학 박사
 1978년~1981년 ETRI 연구원
 1985년~1987년 ETRI 통신망계획부 선임연구원
 1987년~1990년 KREI 소프트웨어하우스 수석연구원
 1990년~현재 중앙대학교 정보시스템학과 교수
 <관심분야> 시스템공학

오 동 옥 (Dong-Ok Oh)

중회원



2008년 2월 중앙대학교 정보시스템학과 졸업
 2011년 2월 중앙대학교 정보시스템학과 석사
 <관심분야> 이동통신, 데이터 필터링, RFID/USN