

# 차량 충돌 사고에 대한 위치 확인 및 서비스 시스템

문 승 진<sup>†</sup> · 이 용 주<sup>\*\*</sup>

## 요 약

무선인터넷 기술의 발달과 응용의 확산으로 위치정보를 이용한 위치기반 서비스 형태는 더욱더 다양해 지고 있다. 특히, 언제 어디서나 사람과 사물 같은 객체의 위치를 인식하고 이를 기반으로 유용한 서비스를 제공하는 유비쿼터스 위치기반 서비스(Ubiquitous Location Based Services : u-LBS)가 중요한 서비스로 대두되고 있다. 이와 관련하여 본 논문에서는 차량 충돌 위치와 관련한 서비스 시스템을 제안한다. 제안된 시스템에서 사용된 GPS Packet에는 위치에 대한 정보와 차량 충돌에 대한 충돌 세기와 시간, 차량에 대한 NodeID 등으로 구성되어 있다. 이러한 데이터들을 이용하여 하나의 패킷이 만들어 지게 되고 차량 간 충돌이 발생할 경우 차량에서 Gateway로 전송된다. Gateway에서 Server로 전송된 패킷은 충돌 여부를 판단하여 위급상황으로 판단되면, 구급센터로 위치정보와 충돌측정여부에 대하여 알려주게 된다. 또한, 이러한 위급상황에 대해서는 외부에 있는 가족 등의 관련된 사람들에게 무선으로 무선단말기(PDA, 휴대폰)를 통해 알려주게 된다. Server에 들어 오게 되는 충돌 정보들은 Database에 저장되도록 구성하였다. 아울러, 제안한 u-LBS시스템의 유효성을 검증하기 위한 실험을 수행하였다.

키워드 : u-LBS, 유비쿼터스 센서네트워크, GPS, 위치추적시스템, 차량충돌시스템

## Location for a Car Crash and The Service System

Seung-Jin Moon<sup>†</sup> · Yong-Joo Lee<sup>\*\*</sup>

## ABSTRACT

The spread of wireless Internet technology development and applications with location information in the form of location-based services are becoming more diverse. In particular, where you recognize the location of objects such as people and things and to provide valuable services based on the ubiquitous and location-based services are emerging as an important service. The collision between the vehicle position measurement in this thesis and offers related service system. Used in the proposed system, the GPS PACKET with information about the location and time of collision for the vehicle crash, the vehicle consists of a NodeID. Cause a conflict between these data at the vehicle, the vehicle through the gateway from the server to decide whether to go on to determine that an emergency situation, Emergency Center, the location information and giving information about whether the conflict is measured. Also, for such an emergency, such as a family on the outside of the wireless terminal related to Wireless (PDA, Phone) is to let me know. The server to want to save the crash information to the database of configuration. Additionally, the proposed U-LBS system to verify the validity of the experiment was performed.

Keywords : u-LBS, Ubiquitous Sensor Network, GPS, Location Based Services, Car Crash Check System

## 1. 서 론

우리나라 교통사고 건수는 21만 1662건이 발생해 사망자 수 6166명을 기록하며, OECD 회원국 가운데 “교통사고 1위”라는 불명예를 유지하고 있다. 그 중 차량충돌 사고는 사망과 중상이 주된 원인으로 손꼽히고 있으며, 더욱이 문제

시 되는 것은 차량 사고에 대한 미흡한 대응과 부정확한 위치 파악, 차량 사고 후에 발견되어 신고 되는 시간동안의 시간 낭비 등의 문제가 있다. 이러한 문제점을 극복하기 위해, 현재 국/내외에서는 차량 간 사고에 대한 위치추적 시스템 기술을 적용하고 활용하기 위한 많은 연구와 사업이 추진되고 있다. 유비쿼터스 컴퓨팅의 개념이 도입되고 모바일 기기의 처리능력과 저장 공간들이 증가하고 무선 네트워크 기술의 발전으로 인해 사용자들은 자신의 주변 환경의 정보를 이용하여 다양한 정보를 제공 받을 수 있는 서비스를 요구하고 있다. 이러한 서비스들을 제공하기 위해서는 기본적으로 수행되어야 할 것이 바로 위치정보 기술이다[1-3].

\* 본 연구는 경기도의 경기도지역협력연구센터사업의 일환으로 수행하였음.  
[GGA0801-45700, U-city 보안감시 기술협력센터]

† 중신회원: 수원대학교 컴퓨터학과 부교수

\*\* 준 회원: 수원대학교 컴퓨터학과 석사과정

논문접수: 2009년 6월 16일

수정일: 1차 2009년 7월 22일

심사완료: 2009년 7월 31일

본 논문에서 제시하는 차량충돌 사고에 대한 위치확인 시스템은 GPS를 이용한 위치정보와 이를 이용한 서비스 시스템 연구이다. 차량에 부착된 충격센서와 위치추적 모듈인 GPS[8]의 데이터를 실시간으로 읽어 들여 차량의 충격 시 발생하는 데이터와 위치데이터를 알 수 있다. 따라서 본 논문에서는 GPS센서를 이용한 데이터 패킷은 위치정보, 충돌 정도를 측정할 수 있는 충돌 정보, 차량 사고 시간 정보를 포함하고 있다. 각 차량에는 노드에 대한 식별자(이하 NodeID)를 부여하여 식별할 수 있게 구성되어 있다. 차량의 충돌로 인하여 발생하는 센서의 이벤트를 실시간으로 수신하여 위치정보와 충돌세기 정도 등을 게이트웨이(Gateway)를 통하여 서버로 보내주게 된다. 이러한 데이터를 받은 서버에서는 충돌 정도를 측정하여 상황인식 알고리즘을 이용한 위급상황 여부에 대한 판단을 한다. 이에 대하여 위급상황으로 판단되면 구급센터에서는 디스플레이 PC의 맵뷰어(Map Viewer)를 통하여 간단하게 확인이 가능하고, 사고에 대한 빠른 대처로 인한 큰 인명피해를 막을 수 있다. 또한 위급상황에 대하여 외부의 앰블런스 기사, 119 응급센터, 112 응급센터는 PDA를 통한 사고를 알 수 있다. PDA를 통한 디스플레이는 간단한 좌표 등의 데이터를 나타내는 텍스트 형식이고 과거의 충돌정보에 대한 저장 및 활용이 가능하도록 u-LBS 데이터베이스도 제공 된다. 차량의 사고로 인하여 게이트웨이를 통한 서버로 들어오는 충돌정보와 위치정보, 차량 노드정보는 데이터베이스에 저장되어 진다. 이렇게 저장되어진 자료들을 토대로 사고 발생지점에서는 차량에게 위험신호를 알려 차량사고에 대한 피해를 줄여 주는 등 과거 데이터를 통한 다양한 서비스가 가능하다.

본 논문의 구성은 다음과 같다. 2장에서는 이와 관련된 관련연구에 대해 알아보고, 3장에서는 제안하는 시스템에 대한 전체적인 적용 방법 및 세부 구현에 대해 설명한다. 4장에서는 제안한 기술에 대한 시스템의 유효성을 검증하고, 마지막 5장에서 본 논문에 연구 내용을 요약하고 향후 연구 방향을 제시한다.

## 2. 관련 연구

### 2.1 LBS(Location Based Service)

LBS는 Location Based Service의 약어로서 위치기반 서비스로 통칭되며 이동통신망을 기반으로 사람이나 사물의 위치를 정확하게 파악하고 이를 활용하는 응용시스템 및 서비스라고 일반적으로 정의 된다[4].

휴대폰 속에 기지국이나 위성항법장치(GPS)와 연결되는 칩을 부착해 위치추적 서비스, 공공안전 서비스, 위치기반정보 서비스 등 위치와 관련된 각종 정보를 제공하는 서비스를 일컫는다. 즉 유선, 무선 통신망을 통해 얻은 위치정보를 바탕으로 여러 가지 서비스를 제공하는 것이 위치기반 서비스이다. 크게 이동통신 기지국을 이용하는 셀 방식과 위성항법장치를 활용한 GPS방식이 있다. 셀 방식은 2000년을 전후해 개발, 보급되기 시작한 방식으로, 위치의 오차범위가

수 Km까지 날수 있어 정확한 위치를 찾기 어렵다. 반면 중계기 등을 이용하기 때문에 건물 안이나 지하 등의 위치도 찾을 수 있는 것이 장점이다. GPS 방식은 위성에서 보내는 위치정보를 휴대폰에 내장된 칩이 읽어 기지국에 알려주는 방식이다. 셀 방식에 이어 등장한 서비스로, 위치의 오차 범위가 넓어야 수백m밖에 되지 않아 거의 정확하게 위치를 알 수 있다. 그러나 위성 신호의 특성으로 인해 반사, 굴절이 잘 되는 고층 건물이나 실내에서는 사용이 거의 불가능하다.

### 2.2 GPS(Global Positioning System)

GPS 또는 범지구 위치 결정 시스템은 현재 완전하게 운용되고 있는 유일한 범지구 위성 항법 시스템이다. 미국 국방부에서 개발되었으며 공식 명칭은 NAVSTARGPS(NAVSTAR-R는 약자가 아님)이다. 무기 유도, 항법, 측량, 지도제작, 측지, 시각동기 등의 군용 및 민간용 목적으로 사용되고 있다. GPS에서는 중궤도를 도는 24개(실제는 그 이상)의 인공 위성에서 발신하는 마이크로파를 GPS 수신기에서 수신하여 수신기의 위치벡터를 결정한다. 위치를 계산하는 방법은 위성으로부터 반송파에 실려 보내진 C/A 코드(Coarse/Acquisition 또는 Standard code)를 GPS 수신기가 감지하면 똑같은 코드를 생성해 두 코드의 시간차를 측정한다. 측정된 두 코드의 시간차에 전파의 속도를 곱하면 GPS 위성과 수신기간의 거리가 구해진다. 그러나 실제 전파 경로를 인한 오차, GPS 위성과 GPS 수신기에 내장된 시계의 오차, 수신기 내부 회로에서 발생하는 오차 등 내부적인 오차와 대기권오차 간섭 전파 등의 외부적 영향에 따른 오차 등이 있다.

## 3. 시스템 시나리오

본 논문에서 제안한 시스템은 모든 차량에 GPS와 충격 감지 센서가 장착되어 있고 이들 장비를 통해 얻어지는 위치정보와 충격정도정보 등에 대한 데이터들을 하나의 Packet으로 만들어 전송이 가능하게 하는 임베디드 장비가 있다고 가정을 한다.

차량충돌이 발생할 경우 GPS를 통해 얻어지는 위치에 대한 정보, 시간 등의 데이터와 충격센서를 통해 얻어지는 충돌정보를 임베디드 장비를 통해 하나의 Packet으로 만들어 Gateway로 전송한다.

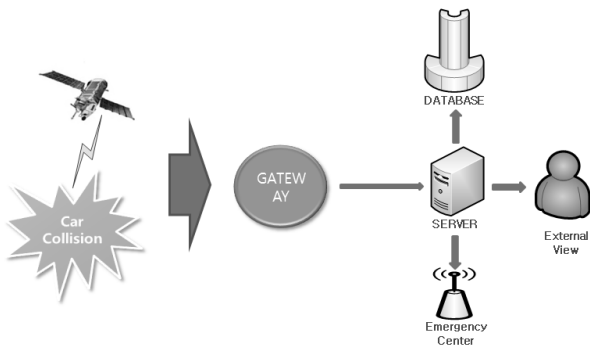
여러 곳에서 수신되어지는 Packet들은 무선을 통하여 Gateway에서 서버로 순차적인 송신을한다.

서버에서는 수신되어진 Packet을 분석하여 본 논문에서 제안되어지는 충돌측정 표와 비교하여 위급상황 여부를 판단하게 된다.

운전자나 위험하다고 판단이 될 경우 무선을 통하여 위치 정보와 충돌여부 등의 데이터를 응급센터로 송신하게 된다.

응급센터로 송신되어지는 데이터들은 데이터베이스서버에도 전송하여 저장한다.

차량 충돌이 발생한 사람과 관계된 사람의 뷰어(Display



(그림 1) 차량충돌 시스템 구성도

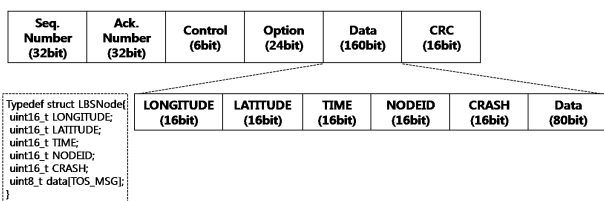
Viewer; 휴대폰, PDA)에서는 충돌 사실과 충돌강도 등의 정보를 서버를 통해 전송받아 확인이 가능하다.

차량충돌로 인하여 생성되는 패킷 데이터(Packet Data)에는 차량사고위치정보, 차량충돌세기정보, 날짜와 시간, 차량 식별이 가능한 차량의 노드ID 등이 포함된다. 이러한 데이터는 사고차량 주변의 게이트웨이에서 정보를 수집하고, 게이트웨이는 지역별 노드 서버에 데이터를 전송한다. 게이트웨이에서 서버로 대량의 패킷데이터가 전송이 될 경우 Packet loss가 발생하게 된다. 이러한 현상을 줄이기 위하여 신뢰성 있는 데이터 패킷 흐름 컨트롤(Data packet flow control)을 하게 된다. 모든 장비간 통신은 무선으로 이루어져 있으며 서버와 단말기만USB를 통한 유선 통신으로 이루어져 있다. 본 논문의 유효성을 실험하기 위해 교내 작은 범위 내에서 실험하였다. 실험을 위해 충격감지센서가 장착된 차량에 시속 0km에서부터 90km까지의 임의의 충돌을 발생시켜 <표 1>에 대한 충돌 측정 지표데이터를 얻을 수 있었다.

### 4. 구현 및 결과

#### 4.1 충돌발생 이벤트

차량 간의 충돌 또는 건물 외벽과의 충돌 등 어떠한 경우 에라도 차량에 충돌이 발생 하였을 경우에(SH-10모델의 충격 감지 센서 이용) 차량에 장착된 Hybus 사의 HMote-GPS가 포함된 센서와 xHYPER-270-TKU의 무선 랜카드가 포함된 임베디드(Embedded)보드를 이용하여 무선으로 패킷(Packet)을 전송한다. 차량 사고 위치정보 데이터(Location Check Data)와 차량 충돌시 발생하는 충돌세기 측정 데이터와 날짜 및 시간 데이터, 차량 식별이 가능한 차량 정보



(그림 2) 생성된 데이터 패킷 구조

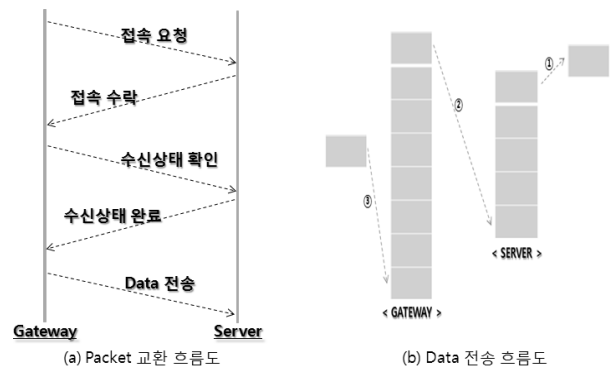
ID 등이 하나의 패킷으로 이루어져 게이트웨이로 전송하게 된다.

#### 4.2 GATEWAY

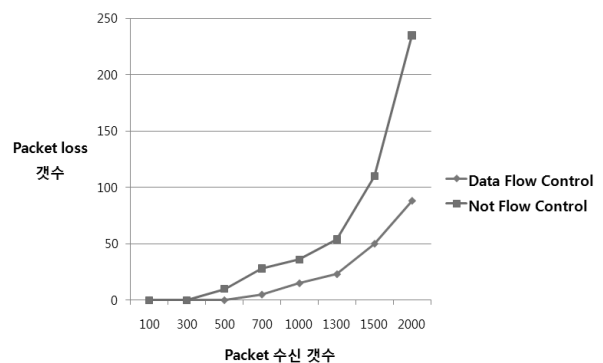
차량에 발생한 충격으로 인해 발생되어진 패킷은 게이트웨이를 거쳐 서버로 전송하고, 이때 게이트웨이에서는 서버로의 원활한 모든 패킷들을 전송하기 위한 데이터 패킷 흐름 컨트롤(Data packet flow control)을 하게 된다. 이는 패킷이 게이트웨이를 통해 서버로 송신될 때 서버의 패킷 과부하를 막으며 패킷의 손실을 막을 수 있다.

(그림 3)의 (a)와 같이 게이트웨이는 서버에게 접속요청을 하고 서버가 이를 수락하게 되면 다시 서버에게 데이터 수신이 가능한 상태인가를 물어보게 된다. 데이터 수신이 가능한지는 서버의 Data Buffer가 Full인지 아닌지로 판별한다. 서버가 수신 가능한 상태일 경우 데이터를 전송하게 된다. (그림 3의 b-②) 하지만 서버의 Data Buffer가 Full일 경우 게이트웨이에게 수신거부를 하게 되며 게이트웨이는 대기상태가 되어 일정시간마다 서버에게 질의를 Packet들을 Buffer에 저장하여 순차적으로 처리한다하게 된다. 서버는 게이트웨이를 통하여 받은 Packet들을 Buffer에 저장하여 순차적으로 처리한다(그림 3의 b-①).

본 논문에서 제안한 방법의 유효성을 검증하기 위하여 다수의 패킷들을 실시간으로 게이트웨이에서 서버로 전송하였다. (그림 4)와 같이 패킷 흐름 컨트롤을 했을 경우 그렇지



(그림 3) Gateway-Server간 Packet 통신



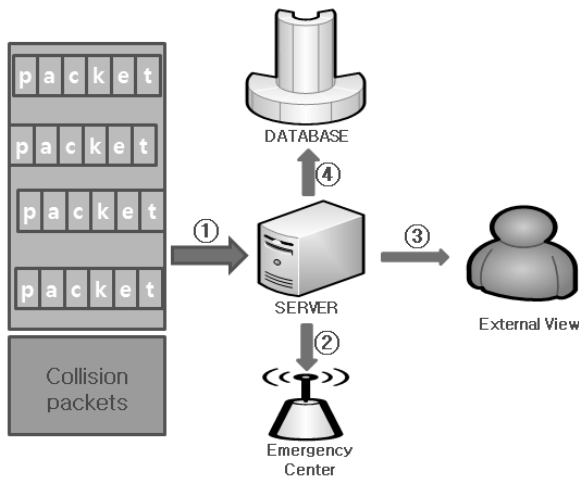
(그림 4) Data Flow Control vs Not Flow Control

않은 경우보다 2배 이상의 패킷 손실을 막을 수 있다는 결과를 얻을 수 있었다.

4.3 SERVER

(그림 5)와 같이 차량 충돌 후에 발생한 순차적인 패킷들을 게이트웨이를 거쳐 서버로 전송한다. (그림 5)의 ①과 같이 순차적인 패킷의 내용은 차량 충돌로 나타난 충돌의 세기, 충돌이 일어날 당시의 시간대, 위치정보 등을 포함(그림 2 참조)하고 있으며 이러한 패킷은 서버내의 최대 버퍼 크기(Size) 만큼 저장한다.

서버에 수신된 패킷내의 충돌 세기 데이터 표본은 <표 1>과 같다. <표 1>의 충돌 측정 등급은 충돌 세기에 의해 측정되며, 충돌 세기는 SH-10 모델의 충격감지 센서를 차량에 부착하여 실제 충격을 가해 데이터를 얻었다. 실험은 0km~90km 속도의 충격을 주었으며, 저속의 속도는 인체를 통하여(차량에 사람이 직접 충격을 주었다 - 도보 경보 달리기 등), 40km이상의 실험은 실제 차량 간 충격을 주어 데이터를 얻도록 하였다. 운전자의 안전성을 고려하여 일정 속도 이상에서는 추출된 데이터를 이용하여 예상치 데이터를 추출하였다. 이렇게 얻어진 충돌 측정 데이터는 값들이 크고 범위가 넓어 <표 1>과 같이 사용자가 알아보기 쉬



(그림 5) Server 구성도

<표 1> 충돌 데이터 분석 표

| 충돌 측정 등급 | 충돌 강도 | 충돌 측정 내용  |
|----------|-------|-----------|
| 5등급      | 1~3   | 운전자 매우 안전 |
| 4등급      | 4~6   | 운전자 안전    |
| 3등급      | 7~9   | 보통        |
| 2등급      | 10~12 | 운전자 위험    |
| 1등급      | 13 이상 | 운전자 매우 위험 |

```
void* receiveFromCrash(void *arg){
return arg;
for (;){
Thread.Sleep(50);
crashPort.Read(crashData, 0, TOS_Msg.Cosnt.TOSH_DATA_LENGTH);
isCrashed = true;
//packet
if (ne == null){
continue;}
byte crashing_info[5];
Console.WriteLine("Coordinates is not null " + ne.N + " " + ne.E);
string[] time = ne.T.Split('.');
byte[] msg = makeMSGtoRF(ne.N, ne.E, time[0], crashing_info);
---중략---
```



(그림 6) Server 데이터 처리

운 형태의 Data Matching을 수행 하였다.

위의 실험을 통해 얻어진 데이터 분석 <표 1>과 같이 충돌 측정 등급 중 2등급 이상일 경우는 (그림 5)의 ①의 응급상황센터(Emergency Center)에 전송하게 된다. 응급상황센터는 충돌 상황을 빠르게 파악하여 위치정보를 토대로 사고 장소로 출동하며, 응급상황센터로 전송되는 데이터들은 외부 단말기(PDA, 휴대폰 등)를 통하여 확인이 가능하고, 가족 혹은 관계된 사람에게 무선으로 통한 정보 전송을 (그림 5)의 ③과 같이 전송한다. 응급상황센터로 전송된 모든 데이터는 (그림 5)의 ④와 같이 응급상황센터내의 데이터베이스에 저장되어 다양한 위험감시 서비스 또는 사고다발지역의 GIS 서비스 구축에 이용된다.

4.4 DATABASE

응급상황센터의 데이터베이스에 저장된 데이터 필드는 충돌사고에 대한 위치정보, 시간, 충격세기, 각 차량들의 노드 ID 등으로 구성된다. 따라서 데이터베이스는 서버로부터 실시간으로 데이터를 받을 준비가 되도록 대기 상태에 있으며, 저장된 데이터를 이용하여 차량사고 다발지역, 차량사고에 대한 위험시간대 등에 대한 정보 추출이 가능하고, 이러한 정보를 이용하여 위험 지역 및 시간 예측 분석 시스템으로 서비스를 이용할 수 있다. 이러한 예측 분석 시스템은 지역 정보 통합 시스템인 GIS를 이용하며 본 논문의 결과와 GIS를 통합하여 u-EGS(Ubiquitous Emergency Gis Service)로 확대 가능하다. 본 논문에서는 소량의 데이터를 수집하고, 상황인식이 가능한 예측 분석만을 위하기 때문에 my-SQL로 데이터베이스를 구축하였으며, C#과 Embedded Visual C를 이용하여 PDA와 뷰어(u-Viewer)에서 확인이 가능 하도록 구현하였다.

```
string N, E;
N = crashInfo.N.Substring(0, 10);
E = crashInfo.E.Substring(0, 10);
crashInfo.showCrashInfo();// 충돌 정보를 받아옴
Console.WriteLine();
DateTime date = DateTime.Now; //시간저장을 위한 변수
string today = date.Year.ToString() + "-" + date.Month.ToString() + "-" + date.Day.ToString();
string query;
string time = crashInfo.hours + ":" + crashInfo.minutes + ":" + crashInfo.seconds;
// 노트에서 온 충돌 시간 분 초 정보를 합쳐서 string time 변수에 저장
string crash;
if(crashInfo.crashDgree < 4) // 충돌정도를나타냄 충돌 세기에 따라서 {
crash = "light"}
else if (crashInfo.crashDgree < 7){
crash = "middle"}
---중략---
```



(그림 7) Database 데이터 저장

4.5 응급상황센터

충돌 감지 후 위급상황이 발생할 경우에 빠른 대처를 위하여 응급상황센터에서 실시간 모니터링과 출동 대기를 한다. 서버에서 차량 사고로 인한 충돌 강도를 <표 1>의 2등급 이상으로 상황인식 시 응급상황이벤트가 발생하여 사고 차량의 인근지역의 응급상황에 대처 할 수 있는 112, 119, 가족에게 현재의 위급상황 <표 1>의 등급 정보와, 차량 충격의 세기, 위치 정보 등을 송신한다. 본 논문에서는 이러한 상황 데이터를 실시간으로 확인 할 수 있도록 맵 뷰어(Map Viewer)를 구현하였다. 본 논문에서 사용한 맵은 인터넷상에서 오픈되어 사용가능한 Google Map을 이용하여 구현하였다. 이러한 맵 뷰어를 이용하여 응급상황센터에서 보다 빠르고 정확한 위치 확인을 통한 대응이 가능하고 사고 후 인명피해의 최소화가 가능하다.

4.6 차량충돌 사고 위치확인 시스템 인터페이스

본 논문에서 제안한 시스템은 차량 충돌 발생 패킷(Packet), 게이트웨이(Gateway), 서버(Server), 데이터베이스(Database), 응급상황센터 등은 <표 3>과 같은 개발 환경으로 구현하였고, 각 에이전트(차량, 게이트웨이, 서버 등)별 주요 처리 변수는 <표 2>와 같이 정의 하였다.

차량 충돌로 발생된 패킷은 임베디드(Embedded) 장비를

```
class MappingFileHandler{
FileStream fs;
string fileName;
int markCnt;
public void init(string fileName){
this.fileName = fileName;
markCnt = 0;}
public MappingFileHandler(string fileName){
init(fileName);}
private void findStringFromFile(string str){
StreamReader sr = new StreamReader(fs);
while (!sr.EndOfStream){
string fline = sr.ReadLine();
fline.TrimStart();
fline.TrimEnd();
if (fline == str)
---중략---
```



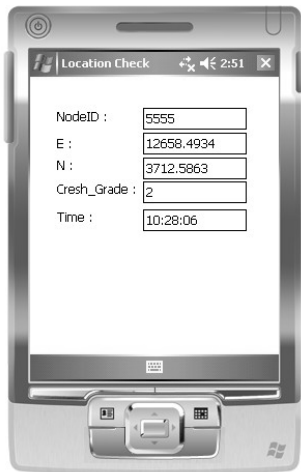
(그림 8) Map View

〈표 2〉 인터페이스 주요 처리 변수 정의

| 용도                        | 번호 | 처리 변수                            | 내용   |
|---------------------------|----|----------------------------------|--|
| 충돌 발생<br>(Car Collision)  | 1  | GPSDecoder_packetDecoder         | GPS로부터 읽어 들인 정보를 토대로 위도, 경도, 시간 등의 정보 획득           |
|                           | 2  | LBSNode_init                     | NODE의 초기화  |
|                           | 3  | LBSNode_makeMSGtoRF              | 무선 전송을 위한 packet을 생성                               |
|                           | 4  | LBSNode_packing                  | 게이트웨이에 NodeID, 위도, 경도, 시간 등의 정보를 전송하기 위한 packet 생성 |
|                           | 5  | LBSNode_receiveFromCrash         | 충돌 센서로 부터 정보 수신                                    |
|                           | 6  | LBSNode_receiveFromGPS           | GPS 센서로 부터 정보 수신                                   |
|                           | 7  | LBSNode_sendToRF                 | RF모듈로 packet 전송                                    |
| 게이트웨이<br>(Gateway)        | 8  | LBSGateWayRoutin_LBSGateWayMain  | 게이트웨이 주 처리기  |
|                           | 9  | CommRW                           | 통신 모듈들에 대한 공통된 인터페이스                               |
|                           | 10 | SocketCommRW                     | CommRW 인터페이스가 적용된 Socket 모듈, Server에 데이터 전송        |
| 서버<br>(Server)            | 11 | Server_Bridge                    | 게이트웨이의 접속을 수락                                      |
|                           | 12 | destinationServerMain            | 목적지(데이터베이스, 구급센터 등)에 정보 전달을 위한 처리기                 |
|                           | 13 | gatewayServerMain                | 게이트웨이로 부터 들어온 패킷 처리기                               |
|                           | 14 | CrashGrade                       | 충돌세기 관별로 인한 충돌 등급                                  |
|                           | 15 | CrashArray                       | 충돌세기에 대한 정보  |
|                           | 16 | LBSGateWayRoutin_pack            | 노드로 부터 얻은 충돌 정보를 패킷으로 생성                           |
| 데이터베이스<br>(Database)      | 17 | LBSDBServerRoutin_bridge         | 서버 접속 수락   |
|                           | 18 | LBSDBServerMain                  | database server의 주요 처리기                            |
|                           | 19 | packetAnalyze                    | 서버로 부터 받은 패킷 분석                                    |
|                           | 20 | sendToDB                         | 패킷 분석으로 얻은 정보를 데이터베이스에 저장                          |
| 구급센터<br>(Mapping Browser) | 21 | MappingRoutinMain                | 맵핑 처리기   |
|                           | 22 | MappingFileHandler_addMakr       | 사용자 추가   |
|                           | 23 | MappingFileHandler_clearMark     | 사용자 전부제거   |
|                           | 24 | MappingFileHandler_deleteMark    | 사용자 제거   |
|                           | 25 | MappingFileHandler_findMarkPoint | 마크의 추가 삭제를 위한 코드 삽입지점 find                         |
| 외부<br>단말기<br>(PDA)        | 26 | PDA_Bridge                       | 서버의 접속을 수락   |
|                           | 27 | Check_NodeID                     | 차량의 NodeID 확인                                      |
|                           | 28 | Location_Check                   | 차량 사고 위치정보   |
|                           | 29 | CrashGrade                       | 차량 충돌 세기   |
|                           | 30 | Time                             | 시간   |

〈표 3〉 인터페이스 개발 환경

| 사용 용도    | SOFTWARE NAME          | SOFTWARE 컴파일러      | SOFTWARE OS   | SOFTWARE 언어          |
|----------|------------------------|--------------------|---------------|----------------------|
| 차량       | hybus hmote-GPS 포함     | 터미널 : cygwin       | tinyOS        | nesC                 |
| 차량       | xHyper-270tku-무선랜카드 포함 | arm-linux-g++      | linux         | c++                  |
| Gateway  | PC                     | VS2008             | windowXP      | C#                   |
| Server   | PC                     | VS2008             | windowXP      | C#                   |
| Database | 노트북                    | VS2008             | windowXP      | C# , mysql           |
| 긴급센터     | PC                     | VS2008             | windowXP      | C#                   |
| 단말기      | PDA                    | .NET Framework 3.5 | Mobile Window | Mobile 5.0 pocket pc |



(그림 9) 외부단말기 View

통해 무선으로 근거리 게이트웨이로 전송하며 게이트웨이와 서버, 서버와 데이터베이스, 서버와 응급상황센터는 무선통신으로 서버와 외부단말기는 USB통신으로 차량 사고 지점에 대한 위치확인이 가능하다. 이러한 패킷 데이터는 충돌 세기의 <표 1>의 등급정보, 위치데이터, 사고시간, 차량식별 코드ID 등의 데이터를 <표 2>와 같이 각 상황에 맞도록 데이터를 처리한다. 충돌 발생시 GPS로부터 위도, 경도, 시간, 차량 코드ID에 대한 정보 패킷을 게이트웨이로 전송하고, 서버는 게이트웨이 패킷에 대하여 수신 후 기존 패킷에 충돌 측정 데이터들을 하나로 묶어 각 목적지에 정보 전달을 위해 사용한다. 정보 전달을 위해 처리기를 이용하며 이는 데이터베이스, 응급상황센터, 외부단말기와의 무선통신을 이용한다. 데이터베이스는 수신된 패킷을 분석 및 저장하며 응급상황센터의 GIS 매칭과 상황인식을 위한 분석 후 위치에 맞는 맵(Map)을 뷰어를 통하여 확인할 수 있다.

### 5. 결 론

본 논문에서는 나날이 늘어나고 있는 차량사고에 대한 인명피해와 위치추적 기술을 융합하여 사고에 대한 바른 위치 파악, 해당위치에 대한 쉬운 지도(Map)확인으로 인한 긴급 출동, 외부에서의 사고 확인이 가능하도록 차량 충돌 사고에 대한 위치확인 및 서비스 시스템을 제안하였다.

본 논문에서 제안한 기술은 GPS를 이용한 위치확인과 임베디드 시스템을 이용하여 무선통신으로 데이터를 송/수신한다. 서버에서 이러한 데이터를 기반으로 충돌정보 판단, 데이터베이스, 응급상황센터, 외부 단말기와의 정보를 송/수신 하고, 데이터 저장을 통한 사고통계조사, 외부 단말기와 응급상황센터의 위치정보를 실시간으로 확인하여 대처가 가능하고 GIS와 간단하게 연동하여 사고다발지역, 최대사고 시간대 조사와 지리적 사고통계 조사와 같은 분야에 이용이 가능하다.

논문에서 제안한 기술은 현재 큰 교통사고가 많이 발생하고, 발견이 가장 더딘 도심을 벗어난 인적이 드문 곳, 도로 등에 대해 초점을 맞추어 개발된 상태이다. 향후에는 도심 지역의 사고를 커버할 수 있는 실외 정밀 위치추적 기술을 연동한 시스템을 개발 할 것이다. 또한, 현재의 my\_SQL과 같은 소용량의 데이터베이스가 아닌 대용량, 실시간 데이터베이스를 이용함으로써 저장된 데이터를 통한 위험지역 예측 및 경계알림, 대규모 GIS를 통한 다양한 서비스 기술과의 융합이 기대되어 진다.

### 참 고 문 헌

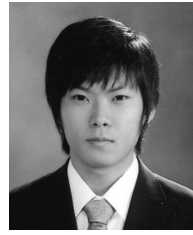
- [1] 김홍규, 문승진 “무선 센서 네트워크에서의 자기센서기반 이동 경로 추적과 데이터 처리 모듈”, 정보처리학회논문지C, 제 14-C권 제1호(2007.02).
- [2] 김병기, 정호민, 김진, 고영용 “크리켓 기반 위치추적 시스템 설계 및 구현”.
- [3] Hightower, J., Borrielo, G “Location system for ubiquitous computing” Computer Volume 34, Page(s):57-66, IEEE Computer Society Press(Aug. 2001).
- [4] 송유진, 한승현, 이동혁, “CS-RBAC 기반의 동적 Location Privacy 보호 구조 설계”, 정보처리학회논문지C, Vol. 13-C No.04 pp.0415-0426 2006.08.
- [5] 정보통신부, ‘u-센서 네트워크 구축 기본계획’, 2004. 2. 17.
- [6] 김완석, 박태웅, 이성국, ‘Ubiquitous Computing의 개념과 업계 동향’, 한국전자통신연구원, 주간기술동향 제 1035호, 2002. 2. 27
- [7] 남상엽, 송병훈, “Mote-Kit를 이용한 무선 센서 네트워크 활용”, 성학당, 2005.
- [8] P.Engel, and P.Misra, “Special issue on GPS: The Global Positioning System,” Proc. of the IEEE International Conf.on, Jan., pp.3-172, 1999
- [9] Kent K.C. Yu,N.r. Watson, and J.Arrillaga, Fellow, “An Adaptive Kalman Filter for Dynamic Harmonic State Estimation and Harmonic Injection Tracking,” IEEE, Vol.20, No.2, pp.1577-1584, 2005.
- [10] Per Engel, Todd Walter, Sam Pullen, Changdon Kee, Yi-Chung Chao, and Yeou-Jyh Tsai, “Wide Area Augmentation of the Global Positioning System,” In Proceedings of the IEEE, August, 1996.
- [11] Linnyer Beatrys Ruiz, Isabels G. Siqueira, Leonardo B. eOliveira, “Fault management in event-driven wireless sensor networks,” Communication of ACM. 2004, pp.149-156.
- [12] B. Karp and H.T. Kung, “GSPR: Greedy Perimeter Stateless Routing for Wireless Networks,” Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking(MobiCom), 2000.



**문 승 진**

e-mail : sjmoon@suwon.ac.kr  
1986년 1월 미텍사스 주립대 컴퓨터학(학사)  
1991년 2월 미플로리다 주립대 컴퓨터학(석사)  
1997년 2월 미플로리다 주립대 컴퓨터학(박사)  
1997년~현 재 수원대학교 컴퓨터학과  
부교수

관심분야: 실시간 임베디드 시스템, 실시간 센서네트워크 시스템, 실시간 데이터베이스



**이 용 주**

e-mail : susuk85@suwon.ac.kr  
2008년 2월 수원대학교 정보미디어학과  
2008년~현 재 수원대학교 컴퓨터학과 석사  
과정  
관심분야: 임베디드 시스템, 센서네트워크 위치추적, 실시간 센서네트워크 운영체제, 유비쿼터스