

# ITS 정보의 TPEG 변환 기법 설계 및 구현

\*이은진, \*김지윤, \*이권익, \*홍성욱 \*임재운, \*김홍수, \*\*김창윤, \*\*김준혁

\*제주대학교 통신공학과 \*\* (주)인터에프씨

\*gen2002@hanmail.net

## Tpeg Transform Method's Design and Implementation for ITS Informations

\*Eun-Jin Lee, \*Ji-Yun Kim, \*Kwoun-Ig Lee, \*Sung-Uk Hong, \*Jea-Yun Lim, \*Heung-Soo Kim

\*\*Chang-Yun Kim, \*\*Jun-Hyuk Kim

\*Dept. of Telecommunication engineering, Cheju University

\*\*Interfc Co., Ltd

### 요약

TPEG 메시지들은 바이트 기반 스트림 형태로 제작되고, 서비스 제공자의 데이터베이스에서부터 최종 사용자의 장비까지 교통 및 여행정보(TTI)관련 응용서비스 정보를 전달하기 위해 사용된다. 본 논문에서는 도로 교통정보에 대한 구간 및 지역별로 인터넷상에서 서비스 중인 지역 ITS정보에 대한 패킷을 분석하여 그 정보를 휴대용 단말기 상에서 구현하고자, TPEG 표준안 중 CTT 정보와 호환성을 갖도록 인코딩을 제작하였고, 수신된 TPEG 메시지에 대한 디코딩을 구현하였다. TPEG 디코더가 내장된 단말기 상에서 지역별 혼잡정보와, 구간별 소통정보 등을 지도정보와 매칭되어 실시간으로 표현되도록 구현하여, 현재 인터넷상에서 서비스 되고 있는 ITS 정보와 그 기능을 비교, 검토함으로써 네비게이션 단말기, PMP, PDA 등 휴대용 단말기에서 TPEG 기반 ITS 정보 제공서비스의 효용성을 입증하였다.

### 1. 서론

국내·외적으로 회자되고 있는 화두는 '기술융합'과 '유비쿼터스'로서 현재 ITS는 이미 기술융합과 유비쿼터스를 지향하고 있고, 이 두 기술이 구현되면 ITS는 보다 완벽한 ITS 서비스를 위한 정보수집 및 제공이 가능해져 ITS 서비스에 획기적인 변화가 올 것이다. 이러한 측면에서 ITS에서는 첨단 디지털 방송서비스 시스템인 DMB를 이용하여 유비쿼터스 개념에 부합하는 언제 어디서나 편리하게 이용할 수 있는 멀티미디어 기반 교통 및 여행자 정보 제공기술의 개발 및 이의 표준화에 깊은 관심을 갖고 있다.

DMB를 이용한 교통 및 여행자정보를 서비스하기 위한 기술인 TPEG이 ISO 표준으로 추진 중이고, 관심이 증가하고 있다 [1][2][3]. 각 방송사들은 DMB를 통한 교통정보 제공을 위하여 교통정보를 수집하는 개별 교통정보센터(한국도로공사, 경찰청, 서울시 등 개별 자치단체, 민간사업자 등)로부터 서로 다른 형식의 교통정보를 수집하여 이를 가공한 후 서비스를 제공하고 있다[4][5].

그런데 각 지자체별로 수집, 가공하여 독립적으로 제공되고 있는 ITS 정보는 정보의 활용도 및 중요성에 비해 시스템별로 상호호환성이 부족하므로 이를 TPEG 과 같은 표준 프로토콜로 변환하여 정보를 공유할 필요가 있다.

본 논문에서는 이와 같은 필요성에 근거하여 제주 첨단교통센터에서 서비스되고 있는 ITS 교통정보를 분석하여 이를 TPEG 규격에 적합하도록 변환하여 인코딩 하였으며, 이를 디코딩하여 실제 제주지역

지도정보와 결합하여 정보를 재가공하여 기존의 인터넷상에서 서비스 되고 있는 ITS정보와 실시간 비교를 함으로서 그 기능을 검증하였다.

### 2. TPEG 프로토콜 구조 및 CTT 구성도

TPEG 메시지들은 바이트 기반 스트림 형태로 제작되고, 서비스 제공자의 데이터베이스에서부터 최종 사용자의 장비까지 교통 및 여행정보(TTI)관련 응용서비스 정보를 전달하기 위해 사용된다. 그림 1은 TPEG 프로토콜에 대한 프레임의 일반구조를 나타낸 것으로서 서비스 컴포넌트 식별자를 통해서 특정 서비스에 대한 정보를 전송하게 된다.

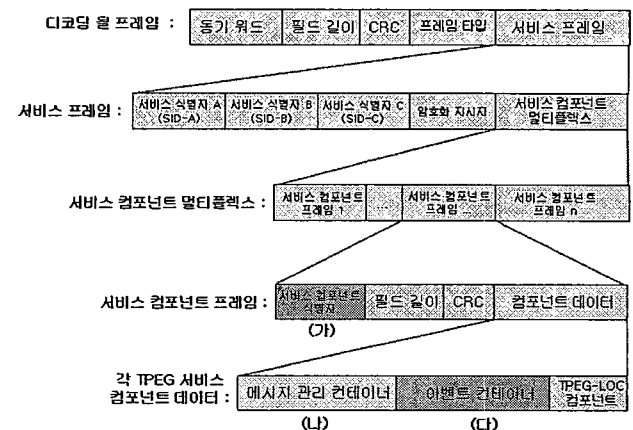


그림 1. TPEG 프로토콜의 기본구조

TPEG 프로토콜 중 혼잡정보인 CTT 컴퍼넌트 기법은 각 도로상의 혼잡정보를 표시하는 프로토콜로서 이중 LinkID를 이용한 위치참조구조를 그림 2에 나타내었다[4].

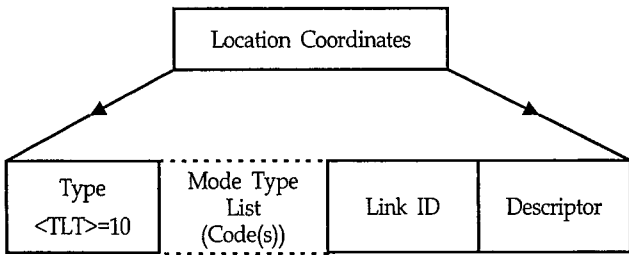


그림 2. CTT 에서의 LinkID를 이용한 위치참조

### 3. ITS 정보 제공 분석

제주 첨단교통정보센터에서 제공되는 ITS 서비스는 교통상황, 기상 정보, 대중교통정보 등을 서비스하고 있다. 교통상황 중에는 제주전역, 제주시, 서귀포시 등으로 구분하여 혼잡정보, 돌발정보, CCTV 영상 서비스 등과 노선별 소통정보를 서비스하고 있다. 이에 실제 인터넷을 통해 서비스 되고 있는 ITS 데이터를 분석하여 TPEG과의 연동방안에 대해 연구하였다. 제공되고 있는 ITS 데이터 형태는 지역별로 구분하여 노드별 고유 ID를 부여하여 속도정보, 통행시간, 혼잡정보를 포함하고 있다. 그림 3은 제주 전역에 대한 ITS 데이터 형태를 보여주고 있다. 제공되는 ITS 데이터를 분석하여 링크 ID 별 노드정보와 거리정보에 데이터베이스화 하였다. 이는 실제 TPEG CTT 정보는 링크 ID와 속도정보만을 제공하기 때문에 TPEG 디코딩 과정에서 인코딩된 파일을 해석한 데이터와 분석된 DB를 연동하여 구간별 통행시간을 표현하는데 사용이 된다.

```

traffic=C0,35,185, 01,,,|
      C1,40,407, 01,,,|
      C2,44,377, 01,,,|
      C3,33,195, 01,,,|
      C4,55,614, 01,,,|
      C5,59,573, 01,,,|
      :
      :
      C69, 0, 0, 00,,,|
      C70, 0, 0, 00,,,|
      C71, 0, 0, 00,,,|
&ldate=200609141553
    
```

그림 3. 제주 전역 ITS 데이터 형태

그림 3의 데이터를 수신한 후 내부에 저장된 도로정보와 연계하여 도로의 위치, 속도, 거리 및 수신여부 등을 추출하도록 프로그램화 하였으며 그 처리 결과를 그림 4에 나타내었다.

ID = C0	( 관양사거리 → 국립제주박물관 )	35 Km/h	185 sec	1.80 Km	정상 (1)
ID = C1	( 국립제주박물관 → 삼양리(보물관) )	40 Km/h	407 sec	4.52 Km	정상 (1)
ID = C2	( 삼양리(보물관) → 국립제주박물관 )	44 Km/h	377 sec	4.61 Km	정상 (1)
ID = C3	( 관양사거리 → 관양사거리 )	33 Km/h	195 sec	1.79 Km	정상 (1)
ID = C4	( 가문사거리 → 양조로교차로 )	55 Km/h	614 sec	9.38 Km	정상 (1)
ID = C5	( 가문사거리 → 가문사거리 )	59 Km/h	573 sec	9.39 Km	정상 (1)
ID = C6	( 가문사거리 → 삼양리(보물관) )	42 Km/h	358 sec	4.18 Km	정상 (1)
ID = C7	( 삼양리(보물관) → 가문사거리 )	45 Km/h	386 sec	4.28 Km	정상 (1)
ID = C8	( 이례수축장입구 → 신성마을 )	36 Km/h	318 sec	3.18 Km	정상 (1)
...					
ID = C64	( 표선 → 대천동사거리 )	58 Km/h	489 sec	7.88 Km	정상 (1)
ID = C65	( 대천동사거리 → 표선 )	65 Km/h	439 sec	7.93 Km	정상 (1)
ID = C66	( 대천동사거리 → 표선 )	37 Km/h	534 sec	5.49 Km	정상 (1)
ID = C67	( 표선 → 대천동사거리 )	48 Km/h	414 sec	5.52 Km	정상 (1)
ID = C68	( 시청입구 → 월드컵경기장 )	0 Km/h	0 sec	0.00 Km	No Data (0)
ID = C69	( 월드컵경기장 → 시청입구 )	0 Km/h	0 sec	0.00 Km	No Data (0)
ID = C70	( 시청입구 → 신시가지입구 )	0 Km/h	0 sec	0.00 Km	No Data (0)
ID = C71	( 신시가지입구 → 시청입구 )	0 Km/h	0 sec	0.00 Km	No Data (0)

그림 4. 제주 전역 ITS 데이터 분석 자료

### 4. 제주지역 ITS 정보의 TPEG 변환 및 구현

#### 가. TPEG CTT 프로토콜 구조

분석된 ITS 정보를 TPEG 프로토콜 구조로 변환하기 위해서 혼잡교통정보메시지(CTT) 응용기법을 활용하였으며, CTT 에 대한 메시지 프로토콜 구조를 표 1에 나타내었다.

표 1. CTT 프로토콜 구조

```

<ctt_message>:=
<intunli>(mid),           : 메시지 식별자
<intunti>(ver),          : 버전번호
<intunli>                  : 바이트로 표현되는 메시지길이
<bitswitch>(selector),   : 메시지 요소
if(selector=xxxxxxx1)<time_t>, : 메시지 발생시간
if(selector=xxxxxx1x)<intunlo>, : 미사용
if(selector=xxxxx1xx)<intunlo>, : 미사용
if(selector=xxxx1xxx)<intunlo>, : 미사용
if(selector=xxx1xxxx)<intunlo>, : 미사용
if(selector=xx1xxxx)<intunlo>, : 미사용
if(selector=x1xxxxxx)<intunlo>, : 미사용
if(selector=lxxxxxxx)<cti_components>,
                                : 혼잡교통정보메시지컴퍼넌트
    
```

#### 나. ITS 정보 CTT 변환 및 인코딩

그림 3에서 받은 ITS 패킷분석을 프로그램화하여, 각 고유번호를 4바이트의 LinkID 형태로 ID 화 하였으며, 이때 전송되어진 속도를 CTT의 구간평균 속도로 인코딩하였다. 그림 3과 같이 실시간으로 수신된 혼잡정보 중 첫 번째 정보에 대해서 TPEG으로 인코딩하여 전송한 후 이를 수신하여 상세 분석한 예를 그림 5에 나타내었다. 도로명을 4바이트의 텍스트형태의 LinkID로 변환 하였으며, 속도정보를 2진 데이터로 변환하였으며 수신시에는 자체에 보관중인 거리정보와 연계하여 소통시간을 추정할 수 있게 하였다.

```

0xA2, 0xA0, 0x00, 0x00, 0x19, 0x80, 0x02, 0x90, 0x00, 0x0E, 0x41, 0x00, 0x06,
0x0A, 0x0A, 0x10, 0x07, 0x01, 0x00, 0x04, 0x43, 0x30, 0x30, 0x30, 0x80, 0x00,
0x03, 0x00, 0x01, 0x23.
    
```

<< 1's CTT Message >>

```

Message ID       : 41632 ( A2A0h )
Version          : 0 ( 00h )
Number of bytes  : 25 ( 0019h )
Number of component : 2 ( 02h )
    
```

[ Location Container ]

```

-----
Length of Loc    : 14 ( 000Eh )
Default Language : Korean ( 41h )
Location Type    : Link ID ( 0Ah )
ID Type          : KS 링크식별자 ( 61h )
Link ID          : 43 30 30 30 ( C000 )
-----
구간평균속도    : 35 Km/h
    
```

그림 5. ITS 정보의 CTT 변환 및 분석 예

#### 다. TPEG 디코더 및 데이터 변환 기법

수신된 Binary TPEG을 분석한 후 이를 지도상에 표시하기 위해 플레시 기법을 사용하였으며, 수신된 Binary TPEG 데이터를 플레시 기법과 연동하기 위해서 그림 6과 같이 데이터를 변환하였다. 이때 각 도로에 대한 LinkID 명과 수신된 속도정보를 분석하여 소통원활, 혼잡, 정체, 정보 없음 등 4가지로 구별하여 데이터로 구성하였다.

```
&myC0=myCC0_0D&
&myC1=myCC1_0D&
&myC2=myCC2_0D&
&myC3=myCC3_0D&
&myC4=myCC4_0D&
&myC5=myCC5_0D&
&myC6=myCC6_0D&
&myC7=myCC7_0D&
&myC8=myCC8_0D&
:
&myC65=myCC65_0D&
&myC66=myCC66_0D&
&myC67=myCC67_0D&
&myC68=myCC68_0A&
&myC69=myCC69_0A&
&myC70=myCC70_0A&
&myC71=myCC71_0A&
&myLoop=72&
```

그림 6. 플레시 입력 데이터 변환

그림 6의 변환된 데이터를 입력으로 받아들이며 지도상에 지역별 혼잡 정보를 표현하기 위해 그림 7과 같이 프로그램 하였으며, 이를 기반으로 3개 지역에 대한 혼잡정보를 그림 8과 같이 지도상에 표현하였다.

```
Var_myC=new Array();

function loaded(){
    loop_num = Number(this.myLoop);
    for(i=0;i<loop_num;i++){
        Var_myC[i] = this["myC" + i];
    }
}

myLoadVars = new LoadVars();
myLoadVars.onLoad = loaded;
myLoadVars.load("dataC.txt");

if (myLoadVars.loaded == true) {
    for (var i = 0; i < loop_num; i++) {
        if(Var_myC[i] == "myCC"+i + "_" + "0A"){
            var ITS=_root["C_"+i]
            ITS.gotoAndStop("tr_1");
        }else if (Var_myC[i] == "myCC"+i + "_" + "0B"){
            var ITS=_root["C_"+i]
            ITS.gotoAndStop("tr_2");
        } else if (Var_myC[i] == "myCC"+i + "_" + "0C"){
            var ITS=_root["C_"+i]
            ITS.gotoAndStop("tr_3");
        } else if (Var_myC[i] == "myCC"+i + "_" + "0D"){
            var ITS=_root["C_"+i]
            ITS.gotoAndStop("tr_4");
        }
    }
} else { gotoAndPlay(_currentframe-1); }
```

그림 7. 플레시를 이용한 지도상의 표현 프로그램

#### 라. 제주지역 ITS 정보의 TPEG 변환 서비스

ITS 정보를 링크 ID와 속도정보를 갖는 CTT 형태로 인코딩을 수행하였고, 수신측에서 분석된 ITS 정보에 대한 DB와 연동하여 디코딩 과정을 통해 링크 ID, 속도정보, 통행시간, 혼잡정보를 갖는 데이터를 생성하였다. 이를 PC상에서 플레시프로그램을 이용하여 지역별 혼잡정보와 도로구간별 소통정보를 구현하였고, 인터넷에서 서비스되는 정보와 비교하여 ITS 정보의 TPEG 적용이 가능함을 확인하였다.

##### 1) 지역별 혼잡정보

제주지역 ITS 교통정보 데이터는 제주전역, 제주시, 서귀포시로 구분하여 3-5분 간격으로 실시간으로 송출되고 있고, 제주전역은 링크 ID가 C0 ~ C71까지 72 구간별, 제주시 지역은 A0 ~ A296까지 297 구간별, 서귀포시 지역은 B0 ~ B40까지 41 구간별로 서비스를 하고 있다.

이렇게 실시간으로 수집되어 가공 후 송출되는 제주지역 ITS 관련 데이터를 기반으로 그림 5와 같이 CTT 정보로 변환되어 송출된 후, 수신된 Binary TPEG 데이터를 실시간으로 분석하여 각 구간별 속도정보를 구별하여 그림 8과 같이 지도상에 표현해주었다. 또한 각 노선별 구간을 선택할 경우 해당 구간에 대한 도로명, 차량평균속도, 예상통과시간 등 구간별 상세 정보를 확인할 수 있게 구현 하였다.



a) 제주도 주요 도로에 대한 혼잡정보



b) 제주시 주요 도로에 대한 혼잡정보



c) 서귀포시 주요 도로에 대한 혼잡정보

그림 8. 제주지역 ITS 혼잡정보 서비스 구현

2) 구간별 소통정보

수신된 ITS 소통정보를 도로별, 구간별로 상세히 표현하기 위해 원래의 정보를 유지하면서 표현하는 기법만을 보완하여 그림 9와 같이 재구성하였다. 주요도로별로 구분하여 소통정보 서비스를 표현한 것으로 원하는 주요도로별 노선을 선택할 수 있으며, 선택된 노선의 혼잡 정보를 그림 9과 같은 방법으로 표현 하였다. 또한 각 노선 구간별 속도와 통과 시간 정보를 실시간으로 나타내주었다. 이는 현재 인터넷에서 서비스되고 있는 ITS 정보와 호환성을 유지할 수 있게 설계되었다.

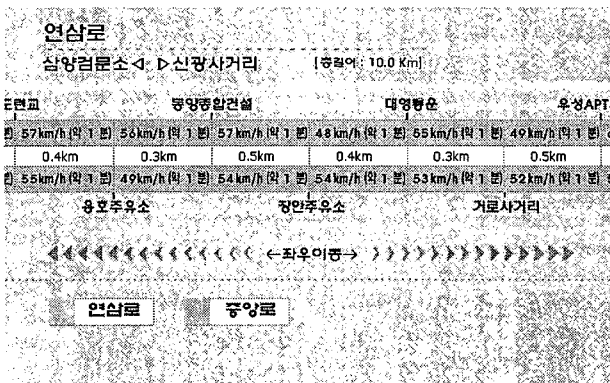


그림 9. 구간별 ITS 소통정보 서비스 구현

마. 종합 시스템 구성도

그림 10은 본 논문에서 제시한 시스템에 대한 전체적인 구성도이다. 기존의 인터넷 등을 통해 서비스 되고 있던 제주도의 주요 도로에 대한 혼잡정보를 동시에 받아서 이를 TPEG 형태로 재가공하여 인코딩을 하고, 방송망을 통해 전송한 후 단말기에서 이를 디코딩하여 해당 데이터 파일을 생성하고, 생성된 데이터 파일을 참조하여 플래시 기법을 활용하여 실시간으로 제주지역 도로 정보를 나타내었다.

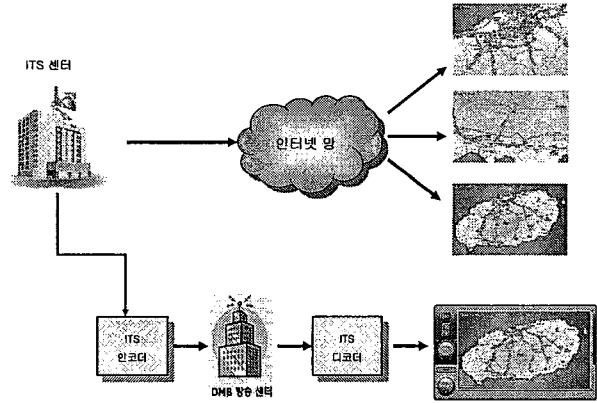


그림 10. ITS-TPEG 변환 시스템 구성도

5. 결론

본 연구에서는 제주지역에서 서비스 되고 있는 ITS정보를 TPEG 프로토콜로 변환하여 DMB 등 휴대용 단말기에서 호환성 있게 구현하기 위한 기법을 제안하여 현재 서비스되고 있는 지역 ITS 정보를 실시간으로 구현하기 위한 기법을 제안하였다. 동일하게 제공되는 실시간 데이터를 CTT 형태와 유사하게 구현하여 기존의 TPEG 서비스와의 호환성을 유지할 수 있도록 설계하였다. 향후 연구과제로는 ITS 별로 독립적으로 지정된 도로명을 LinkID 와 호환성 있게 구성하는 일과, CCTV 등의 정보를 실시간으로 전송하기 위한 프로토콜 연구 등이다.

[참고문헌]

- [1] EBU BPN 027-2 Transport Protocol Experts Group (TPEG) TPEG Specifications Part 2: Syntax, Semantics and Framing Structure
- [2] EBU BPN 027-3 Transport Protocol Experts Group (TPEG) TPEG Specifications Part 3: Service and Network Information Application
- [3] EBU BPN 027-6 Transport Protocol Experts Group (TPEG) TPEG Specifications Part 6: Location referencing for Applications
- [4] TPEG표준 혼잡교통정보 메시지 응용 : TPEG-혼잡교통정보메시지\_0.9/001, KS X ISO 2005
- [5] 김순철, 김건, 정영호, 조삼모, 안충현, “이동 멀티미디어방송 시스템에서 TTI응용서비스를 위한 뉴스정보 제공방법”, 2005 한국방송공학회 학술대회, pp.131-134, 2005년 11월
- [6] 정영호, 김순철, 조삼모, 김건, 안충현, 김환우, “지상파 DMB 기반의 POI 서비스 설계 및 구현”, 2005 한국방송공학회 학술대회, pp. 127-130, 2005년 11월
- [7] 임재운, 강용진, 이은진, 이권익, 홍성욱, 안충현, 김순철, “News/POI정보의 tpegML 제작 및 구현”, 2006 한국방송공학회 학술대회, pp. 235-238, 2006년 11월