

U-city를 위한 효과적인 Media 획득 및 전달 체계 : SLIM Framework

안성원^o, 한상범, 유혁
고려대학교 운영체제 연구실
{swahn^o, shan, hxy}@os.korea.ac.kr

Effective Media Acquisition and Streaming for U-City : SLIM Framework

SungWon Ahn, SangBeom Han, Hyuck Yoo
Korea university Operating System laboratory

요 약

앞으로 다가올 U-City (Ubiquitous City) 환경에서는 도시 내에 설치된 카메라를 통해서 각종 정보를 수집해야 하는 경우가 많다. 수집된 정보들은 그 양이 방대하고 복잡하므로 이를 효과적으로 저장하고 전달하기 위해서는 분산된 환경의 카메라 네트워크 시스템에 대한 연구가 필요하다. 본 논문에서는 도시 내에 분산된 카메라 네트워크를 효율적으로 관리하고, 수집된 정보를 필요로 하는 모듈 서버에게 효과적으로 분배하는 U-City 지향적 시스템에 대한 설계를 제시하고 그 기반 연구를 진행한다.

1. 서 론

1-1. 연구배경

최근 U-City의 구축과 그에 대한 관심이 증대되고 있다. U-City는 도시 속에서 언제 어디서나 사용자의 단말을 이용하여 컴퓨팅을 할 수 있고, 네트워크 자원을 사용할 수 있는 미래형 도시이다. 사용자는 U-City 안에서 전자공간과 물리공간의 서비스를 모두 이용할 수 있다. 때문에 사용자들은 기존의 도시에서는 받을 수 없었던 새로운 형태의 서비스를 받을 수 있으며, 보다 편리한 삶을 살 수 있게 된다. 이러한 이유에 의해서 U-City를 위한 많은 아이디어와 정책이 제시되고 있다.

U-City에서는 도심 곳곳에 설치되어 있는 다수의 카메라를 통한 각종 정보를 손쉽게 확인할 수 있는 일이 가능해질 것이다. 예를 들어 도시의 어느 특정 지역에서 월드컵 응원과 그에 따른 행사가 진행 중이라면, 현재 그 지역에 있지 않은 많은 사람들은 카메라를 통하여 응원과 행사를 관람할 수 있다.

뿐만 아니라 관광지, 공원, 각종 경기장 등의 공공장소에 설치된 카메라를 통해서 사용자가 관심 있는 각종 공연, 집회상황, 풍경, 경기상황들을 실시간 또는 비실시간으로 제공 받을 수 있다. 도로에 설치된 카메라를 통한 실시간 교통정보도 손쉽게 얻을 수 있게 된다. 또한 이러한 카메라 환경이 구축된다면 정부 차원에서의 각 지역의 환경 관리, 각종 사건 사고관리 등에 이용될 수 있다. 이러한 카메라 환경은 범죄나 교통사고 등에 대한 사후처리를 무척 용이하게 할 수 있는 능력을 제공한다.

1-2. 연구동기 및 목표

이와 같은 U-City를 구축하기 위해서, 다수의 카메라들에 대한 통합적인 관리 및 다른 시스템들에 응용할 수 있는 '카메라 네트워크 시스템'에 대한 연구가 필요하다. 기존의 감시카메라에 대한 연구나 시설로는 앞으로 구축될 U-City에 설치된 거대 카메라 네트워크를 관리하는 데에는 한계가 있다. 따라서 본 논문에서는 흩어져 있는 다수의 카메라 네트워크를 효율적으로 관리하는 방법에 대하여 설계하고, 이를 제시한다. 또한 카메라로부터 수집된 정보를 사용자의 요구에 맞게 전달하고, 이를 관리하는 모듈 서버를 효과적으로 관리 및 분배하는 시스템에 대한 연구와 그 기반연구를 진행한다.

2장에서는 본 연구와 관련이 있는 연구들에 대하여 간략하게 알아보고, 3장에서는 본 논문에서 설계, 제시한 SLIM(Scalable Live Media) Framework에 대한 개념에 대해 설명한다. 4장에서는 SLIM Framework가 지원해야 할 문제와 해결책에 대하여 알아보고, 5장에서는 제안한 SLIM Framework의 구조를 살펴본다.

2. 관련연구

본 논문에서 제시하는 내용에 관련된 최근 연구로는 대표적으로 스페인 발렌시아시에 적용된 TMS[참고문헌 1] (traffic - management system)과 유럽에서 진행 중인 Candela - Storage, Analysis, and Retrieval of Video Content in Distributed Systems[참고문헌2,3,4] 프로젝트가 있다.

TMS 에서는 유연한 비디오 스트리밍을 통해 도심의 교통량을 통제한다. 도로영상정보를 도로에 설치된 다수의 카메라로부터 습득하고 이 영상 값을 MPEG-4 의 방식으로 인코딩 한다. TMS 에서는 각각의 카메라들이 카메라 서버에 연결되어 있고 이 카메라 서버는 core network 에 연결된 형태의 방사형의 구조를 하고 있다. 코딩된 streaming data는 이 core network를 통하여 교통 중앙 처리 센터(Traffic management center)로 전달된다. data를 전달받은 센터에서는 video stream 을 저장하고 분석하여 'artificial vision system'의 학습 자료로 인용된다. 이렇게 publishing 된 video signal은 core network의 stream management server를 통하여 각 신호등 통제 장치에 전달되며 교통신호체계를 상황에 따라서 유동적으로 통제하게 된다.

Candela 프로젝트는 유럽 ITEA 프로그램의 일환으로 video data를 분산된 시스템에서 저장과 분석 등을 통하여 사용자 지향적인 서비스를 제공하는 데에 초점을 맞추고 있다. 이 프로젝트에는 세부 연구로 여러 가지의 주제를 두고 있다. 이 프로젝트 에서는 카메라로부터 생성된 방대한 양의 video data를 효과적인 방법으로 관리하고 사용자가 손쉽게 검색하여 이용하는 방법을 제시하고 있다. 그리고 프로젝트 내에서 제시된 Candela platform을 통하여 사용자의 서로 다른 mobile 단말에 대하여 media DB server의 video data에 접근할 수 있는 호환성 있는 지원과 관리를 제시한다.

3. SLIM(Scalable Live Media) Framework

고용량의 미디어 데이터를 효과적으로 코딩하여 기존의 네트워크망을 통해 다양한 사용자 단말에 전송하는 기술은 U-City 환경구축에 꼭 필요한 요소 이다. 다수의 카메라를 효율적으로 제어하며, 카메라로부터 생성되는 미디어 데이터를 효과적인 트랜스코딩을 통하여 QoS를 보장하면서 네트워크를 통해 요청된 자료가 사용자에게 전송이 되어야 한다.

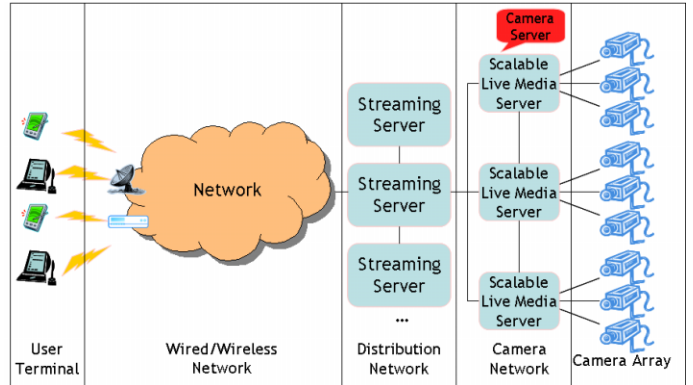
위와 같은 서비스를 보장하기 위해서는 첫째로, 다수의 카메라와 카메라 서버의 카메라 네트워크를 어떻게 효과적으로 관리하느냐가 관건이 된다. 이는 trans coding과 scalable coding을 통해 미디어 데이터의 1 차적인 가공을 해야 하는 문제 이다.

둘째로, 이렇게 가공된 데이터를 어떤 식으로 처리 할 것인가가 관건이 된다. 즉, 사용자의 다양한 요구조건을 만족시키며 데이터의 신뢰성과 QoS를 보장하는 중간처리 장치가 필요할 것이다.

셋째로, 사용자의 단말(핸드폰, PDA, PC, 노트북 등)은 서로 다른 다양한 device를 가지고 있는데 이렇게 다양한 device에 호환성과 적절한 데이터 수준을 보장해줄 수 있어야 한다.

본 논문에서는 이러한 조건을 만족시켜 줄 수 있는 U-City를 위한 효율적인 media stream 전달 기법으로 Slim(scalable live media) Framework를 제시한다.

본 논문에서 제시한 Slim Framework는 <그림 1>과 같다.



<그림 1> 제안된 SLIM Framework 구조

<그림 1>에서 보는 바와 같이 도심 곳곳에 설치되어 있는 카메라들은 하나의 scalable live media server (이하 camera server)에 여러개씩 묶어 그룹(camera network)을 이룬다.

그룹에 속해 있는 각각의 camera server들은 연결된 각각의 카메라들로부터 전송받은 영상 자료를 encoding 한다. 이때 영상자료들은 scalable encoding 이 되는데, 이것이 네트워크망의 상태나 사용자 단말의 다양한 성능에 따라 차별적인 지원을 함으로써 사용자의 요구를 만족시켜줄 수 있는 media stream 전송의 기반이 된다.

각각의 camera server는 camera network 내에서 상호간의 적절한 연동을 하고 한쪽 서버에 과부하가 걸리는 문제가 발생하게 되면 이웃한 서버에서 media stream 전송을 분담하는 로드 밸런싱(load balancing) 기법으로 과부하에 대한 부담을 덜어준다.

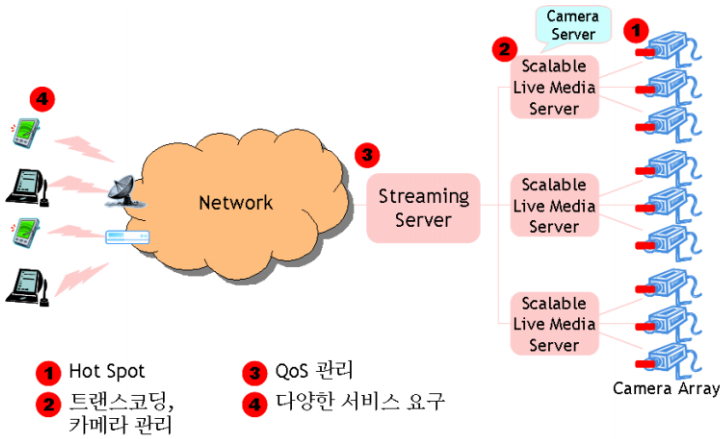
다수의 사용자가 특정지역의 카메라에서부터 생성된 영상정보에 관심이 있게 된다면 특정지역의 카메라와 연결된 camera server에 과부하가 생기는 것은 당연한 일이다. 차후 또 논하겠지만 이러한 문제를 해결하기 위함으로 본 연구를 진행하면서 서버간의 효과적인 연동을 구현 할 것이다.

camera server들로 이루어진 camera network는 streaming server들로 이루어진 distribution server와 연결되어 있다. streaming server는 camera server로부터 encoding 된 media stream을 받아 중간 네트워크 망을 통하여 데이터를 저장, 사용자들에게 적절히 분배, 전달하는 역할을 하게 된다.

streaming server는 다양한 사용자 단말의 능력과 device를 지원하기 위한 모듈을 내부에 갖추고 있다. 또한 현재 사용자가 접속되어 있는 중간 네트워크 망의 상태에 따라서 사용자에게 전송할 데이터의 수준을 결정하여 유연한 media stream 전송을 보장하게 된다.

User terminal(사용자 단말)은 streaming server로부터 받은 차별화된 media stream을 받아 재생할 수 있는 모바일 단말에 최적화된 light weight 한 player를 탑재 하고 있다. 사용자 단말은 또한 단말 자신에 대한 정보를 streaming server 에 전송함으로써 이러한 차별화된 data를 받아 볼 수 있게 되는 것이다.

4. SLIM Framework 가 지원해야 할 문제



<그림 2> SLIM Framework 가 지원해야 할 문제

4-1. HOT Spot 문제

제한된 프레임 워크에서는 카메라들 사이에 Hot spot 문제가 발생 할 수 있다. 즉, 도시 내에 어떤 특정 지역에 행사가 일어나고 있다면, 사용자는 해당지역의 정보에 관심을 가질 것이다. 이렇게 되면 자연히 특정 지역 내에 국한된 카메라와 카메라서버에 사용자접속수가 증가할 것이고, 사용자의 요구가 한곳에 몰리는 병목현상이 발생하게 된다. 이러한 병목현상은 서버에 과부하를 가져다주며, 원활한 정보지원에 차질을 빚게 된다. 따라서 이러한 문제를 해결해야 한다.

카메라를 통한 데이터의 획득 체계에 대한 효과적인 관리를 위해서는 카메라 네트워크 토폴로지를 구성하여 특정지역 소수 서버에만 사용자의 요청이 몰릴 경우, 이웃한 유휴서버들이 데이터의 일정량에 대한 처리를 담당함으로써 과부하를 분산시키는 로드 밸런싱 기법에 대한 연구를 하여 이를 해결할 수 있다.

4-2. 트랜스코딩

고용량의 비디오 데이터를 코딩하는 데에 있어서 어떤 식으로 코딩을 하는 것이 유동적인 데이터 전송에 큰 효과를 볼 수 있는지 관건이 된다. 카메라로부터 획득한 영상정보를 1차적으로 가공하는 데에 있어서 scalable coding 기법이 필요하다.

예를 들어 1분짜리 미디어 데이터를 1200 장의 슬라이드(초당 20프레임의 속도)로 코딩하여 데이터를 전송한다고 가정 할 때, 고속의 데이터 통신과 사용자 단말의 성능이 보장된다면 고화질을 보장하는 1200 장의 데이터를 모두 보내거나, 아니면 저속의 저사양의 성능을 위한 중간 중간의 데이터를 건너뛴 꼭 필요한(영상을 구성하는데 무리가 없는) 600 장의 데이터만을 보낼 수도 있다.

이처럼 사용자의 device 는 다양하며, 사용자가 이용하는 단말기의 성능 또한 천차만별이기 때문에 이러한 scalable coding은 사용자의 각기 다른 단말에 대하여 차별적인 media data의 전송을 가능하게 한다. 때문에 scalable coding 는 다양한 사용자의 환경에서 매우 큰 효과를 발휘 하게 된다.

4-3. QoS 관리

트랜스 코딩을 통하여 1차적으로 가공된 데이터를 전송하는 데에 있어서 서비스의 질은 어느 정도 보장되어야 한다. 때문에 앞서 설명했던 것처럼 제한한 SLIM Framework 환경에 적합한 QoS 관리자가 필요하다. 이는 사용자가 접속하는 유, 무선 네트워크의 상태에 따라서 유동적으로 조절할 수 있는 rate control 기법이라고 볼 수 있다.

시스템의 신뢰성을 향상시키기 위하여 오류발생 모듈에 대한 self-surgery 또한 고려해야 할 사항이다. 사용자들의 device의 능력에 따른 차별적인 서비스를 위하여 트랜스 코딩된 데이터를 전송하는 camera server 및 distribution server의 관리 기법도 빼놓을 수 없다. 각각의 distribution server 간에도 상호 유동적인 호환과 로드 밸런싱 기법을 추가할 수 있겠다.

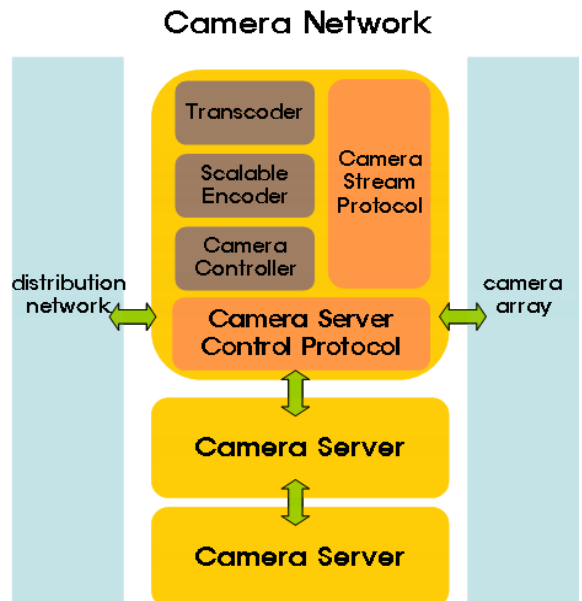
4-4. 다양한 서비스 요구

위에서 제시한 서비스를 받기 위해서는 사용자의 단말에도 위 기능들을 충분히 소화해 낼 수 있는 application 이 필요하다. 최저의 전력으로 최고의 화질을 낼 수 있는 경량화 된 미디어 플레이어 필요하다. 한정된 네트워크 자원에서 사용자의 각각의 요구사항을 만족시키기 위해서는 각 요구에 따른 우선순위 배분문제와 비슷한 요청을 한 사용자들에게 데이터를 그룹화하여 보낼 수 있는 그룹핑 매커니즘도 필요하게 된다.

5. SLIM Framework 의 구조

5-1. Camera Network

여러 개의 camera server(scalable live media server) 들을 엮어 놓은 군집을 본 연구에서는 camera network 로 정의 하였다. camera network의 구성요소인 camera server의 구조는 <그림 3>과 같다.



<그림 3> SLIM Framework 내의 camera network 구조

<그림 3>의 camera server는 5개의 독립적인 모듈로 구성되어진다. 먼저 camera array에서 들어온 영상 정보를 camera server에 전달하는 camera stream protocol이 필요하다.

카메라에서 생성되어진 일련의 stream들은 camera server내의 transcoder 와 scalable encoder에서 1차적으로 가공되어진다. transcoder 와 scalable encoder의 역할은 대용량의 영상정보를 적절한 단위로 코딩하는 것이다. 이렇게 인코딩된 영상정보는 2차적으로 사용자들에게 배분되기 위한 distribution network로 이동된다.

사용자의 요청에 따른 카메라 제어를 위해 기본적인 카메라의 시점전환, zoom view를 지원하는 camera controller도 탑재된다.

각각의 camera server들은 서로 유동적인 연결을 보임으로써 본 논문에서 차후 논하게 될 hotspot problem을 해결 할 수 있게 된다. 이러한 유동적인 연결을 위한 각 서버간의 camera server control protocol 또한 필요하다. 이 프로토콜은 distribution network 의 stream server에 의해서 통제되어진다.

다른 camera server와의 연동을 통해서 특정 카메라 집단과 그 집단을 담당하는 camera server에 과부하가 부여될 때 이를 해결하는 로드 밸런싱을 지원하는 토대가 된다.

5-2. Distribution Network

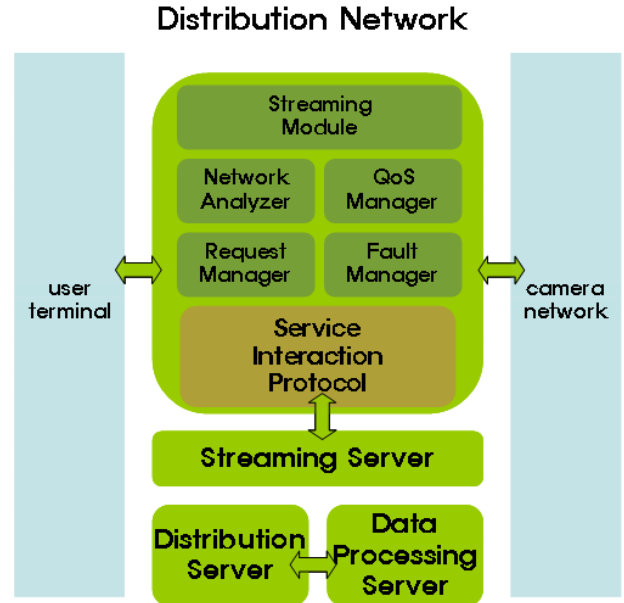
camera network에서 받은 media streaming을 2차적으로 처리하기 위한 distribution network의 집단을 본 연구에서는 distribution network 로 정의 하였다.

distribution network를 구성하는 streaming server의 구조는 <그림 4>와 같다.

distribution network에서는 기본적으로 camera network에서 받은 media streaming을 사용자의 요청과 요청한 사용자의 단말 능력에 따라 적절히 분배하는 역할을 한다. 또한 camera network에서 받은 media streaming을 일시적으로 저장하기도 하고, 필요할 경우 data processing server와 같은 추가적인 장치를 두어 데이터를 영구적으로 저장 할 수 도 있다. 여기에서 data processing server는 예를 들어 정부나 공공기관에서 distribution network를 중앙집중식으로 제어할 수 있는 장치로 점목될 수 있다. 본 논문에서 제시하는 Slim(scalable live media) Framework가 교통통제 시스템에 응용된다면, 각 도로에서 입수된 카메라 정보들을 다수의 user terminal이 아닌 도로교통 관리공단과 같은 하나의 기관에 제공되기 위해 운영하는 중앙통제 시스템이라고 볼 수 있다. distribution network를 구성하는 streaming server는 기본적으로 6개의 모듈을 탑재한다.

먼저 camera network에서 받은 media streaming을 저장하고 있다가 처리하는 streaming module이 있어야 한다.

사용자의 요청은 request manager에서 분석하여 일차적으로 network analyzer 모듈에게 사용자의 현재 네트워크 환경을 조사 할 것을 요청한다.



<그림 4> SLIM Framework 내의 distribution network 구조

이차적으로는 streaming module에게 사용자가 요청한 자료를 찾도록 한다.

사용자에게 요청된 자료를 제공하기 위해서는 사용자가 속해있는 네트워크 환경에 대한 분석이 필요하다.

이 분석을 network analyzer가 맡고 있다. network analyzer는 사용자가 속한 네트워크의 환경을 분석하고 그 상태에 따라서 scalable 하게 encoding 된 자료를 어떻게 보낼지 결정하는 QoS manager 에게 정보를 넘겨주게 된다.

QoS manager는 network analyzer로부터 받은 정보를 토대로 사용자가 속한 네트워크가 고속의 안정적인 네트워크라면 streaming module 로부터 완전한 streaming data를 보낸다. 만일 사용자가 속한 네트워크 환경이 저속이라면 사용자의 단말에서 화질이 다소 떨어지더라도 사용자가 데이터를 식별하는데 별 문제가 없도록 streaming module 로부터 선별된 꼭 필요한 streaming data만을 보내게 된다.

또한 QoS manager으로 인하여 사용자 단말의 device 와 성능에 대한 기본적인 정보를 받아 해당 단말의 device에 최적의 상태로 동기화 될 수 있는 streaming data를 기대할 수 있다. 사용자 단말의 data 처리 능력은 제각각 이다. 이러한 사용자 단말에게 같은 수준의 streaming data를 보낼 수는 없다. 때문에 QoS manager에 의한 차별화된 데이터 전송을 위한 작업이 필요하다.

streaming server는 자체적으로 오류를 복구 할 수 있는 능력을 지닌 fault manager 모듈 또한 탑재된다. fault manager 는 비디오 데이터의 누락 또는 왜곡에 대한 정보에 대한 복구 능력을 지원 할 것이다.

streaming server는 사용자의 요구에 적절히 대응하기 위한 streaming server간의 상호 협력도 필요하게 되는데 이는 service interaction protocol 을 통하여 구현 될 수 있다.

service interaction protocol에서는 사용자의 요청에 따른 능동적인 streaming server 데이터 처리를 보장할 수 있다. 각각의 streaming server들은 이 프로토콜을 통하여 서로가 가지고 있는 데이터를 공유 할 수 있고 사용자로 하여금 장소나 현재 접속해 있는 streaming server에만 국한되지 않게 자유로운 media data를 얻을 수 있다. service interaction protocol은 streaming server의 망을 구성할 수 있는 교두보 역할을 한다.

service interaction protocol은 이처럼 QoS manager 모듈과 더불어 각각의 사용자 단말에게 최적화 된 streaming data를 보내는 것을 보장하기 위한 역할을 한다.

5-3. User terminal

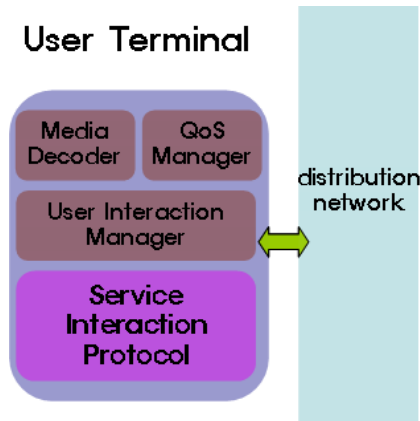
distribution network에 사용자가 자료를 요청하고 distribution network로부터 사용자가 자료를 받을 수 있는 사용자 단말 집단을 user terminal 로 정의하였다. SLIM Framework를 가장 효과적으로 지원할 수 있는 user terminal의 구조는 <그림 5>와 같다

user terminal은 4개의 모듈을 탑재하게 되는데, 기본적으로 light weight한 media decoder를 탑재하게 된다. user terminal이 유선이라면 문제 될 것이 없겠지만 대부분의 user terminal이 무선이라고 본다면 저 전력을 소모하는 light weight한 media decoder(player)는 필수적이다.

user terminal은 핸드폰, PDA, 노트북 등 매우 다양하다. 또 다른 종류의 핸드폰이라면 핸드폰의 성능과 기종에 따라서 핸드폰이 소화해 낼 수 있는 streaming data 능력은 다르다.

이런 지원을 가능하게 하기 위해서는 단말에도 QoS manager 모듈이 필요하다. QoS manager 모듈은 단말의 성능과 현재 속해있는 네트워크 상태에 따라 streaming server에게 알맞은 양의 streaming data를 요청한다.

사용자 단말에 탑재되어 있는 Service interaction protocol 은 streaming server 에 탑재된 service interaction protocol 과 연동하여 특정 서버에 구매받지 않는 호환성 높은 서비스를 보장 한다. 사용자가 언제 어느 위치에 있더라도 사용자는 원하는 서비스 media data를 받을 수 있다.



<그림 5> SLIM Framework 내의 user terminal 구조

User interaction manager 모듈은 추가적인 사항으로 사용자 단말의 사용자 편의사항이다. 사용자가 단말을 통하여 입력되는 정보는 키보드나 자판에 국한된 것이 아니다. 터치스크린을 통한 펜 입력, 음성정보 입력등도 있을 것이다. 현재 시중에 판매되는 많은 단말들은 다양한 interface를 제공하는데 user interaction manager 모듈은 이러한 interface와 호환되게 또는 새로운 interface를 지원하는 user terminal 환경을 가능하게 한다.

6. 결 론

U-city 구축의 일환으로 다수의 카메라에서 수집된 정보를 다양한 사용자에게 효과적으로 전달하는 기법에 대한 연구가 필요하다. 이를 위해 본 논문에서는 Slim Framework와 이를 구축하기 위한 기초 연구를 제안하였다. 제안한 Slim Framework는 고용량의 미디어 데이터를 각각의 다른 device를 갖는 사용자 단말의 요청에 효과적인 지원을 가능하게 해준다. Slim Framework는 다수의 카메라를 관리하는 camera network와 이를 분산해주는 distribution network로 구성된다. camera network에서는 scalable coding을 통하여 카메라로부터 수집된 영상 데이터를 1차적으로 가공하고 이를 토대로 distribution network에서는 사용자의 요청에 맞게 데이터를 2차적 가공 분산 시켜준다.

본 논문에서 제시한 바와 같이 향후 카메라로부터 영상자료를 얻는 camera server를 구축하고, camera server들 간의 상호작용과 로드밸런싱을 가능하게 하는 camera network를 구축한다. 그리고 camera network에서 받은 자료를 효과적으로 관리하며 분산된 camera server 제어를 위해 streaming server를 구축한다. distribution network는 streaming server 간의 상호작용을 통한 호환적이고 유연한 사용자 device를 지원한다.

잘 구축된 Slim Framework는 U-City의 다양하고 많은 분야에 매우 유연하고 호환성 있는 product가 될 것이라고 기대한다.

7. 참고문헌

- [1] Manuel Esteve and Carlos E.Palau Universidad Politecnica de Valencia, Spain "A Flexible Video Streaming System for Urban Traffic Control", Proc IEEE Int'l Conf, IEEE press January-March 2006, pp78-83.
- [2] P. Pietarila et al. "Candela—Storage, Analysis, and Retrieval of Video Content in Distributed Systems: Real-Time Video Surveillance and Retrieval," Proc. IEEE Int'l Conf. Multimedia and Expo (ICME), IEEE Press, 2005, pp. 1557-1560.
- [3] Paavo Pietarila et. "Candela—Storage, Analysis,

- and Retrieval of Video Content in Distributed Systems: Personal mobile multimedia management" VTT Electronics Oulu, Finland IEEE 2005.
- [4] P.merkus et. "Candela — integrated Storage, Analysis, and Distribution of Video Content for Intelligent Information systems" Proceeding of the European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology, EWIMT, London, U.K. 2004.
- [5] M.M. Trivedi, I. Mikic, and G. Kogut, "Distributed Video Networks for Incident Detection and Management," Proc. IEEE Conf. Intelligent Transportation Systems, IEEE Press, 2000, pp.155-160.
- [6] J. Nesvadba et. " Real-Time and Distributed AV Content Analysis system for Consumer Electronics Networks " Int. Conf. for Multimedia and Expo ICME, Amsterdam, The Netherlands, June 2005
- [7] F. de Lange et. "A Networked Hardware / Software Framework for the Rapid Prototyping of Multimedia Analysis Systems" Int. Conf. On Web Information Systems and Technologies (WEBIST), Miami, USA , 2005
- [8] W. Berkvens "Media Distribution in a Pervasive Computing Environment", IEEE Proc. Con. on Pervasive Computing and Communications, March 2005
- [9] S. Desurmont "A generic flexible and robust approach for intelligent real-time video surveillance systems", Proceedings of the SPIE – Real-time imaging VIII, Vol 5297, No 1, Jan. 2004
- [10] Xavier Desurmont, et al "A seamless modular image analysis architecture for surveillance systems" p 66-70 London, UK 2004.