

TMO기반의 실시간 다중 상대 원격 모니터링 시스템의 구현

정 학⁰ 정갑주¹

건국대학교

zealottm@gcslab.konkuk.ac.kr⁰ jeongk@konkuk.ac.kr¹

Implementation of

TMO-based Real-time Multi-target Tele-monitoring System

Xue Zheng⁰ Karpjoo Jeong¹

Konkuk University

요 약

오늘날에 있어서 e-Science는 점차 현실 생활에 침투되고 있는 추세를 보이며 많은 기기들의 발전과 더불어 온라인 원격 제어, 무선통신 등이 보다 쉽게, 많은 영역에서 지원되고 있다. 과학영역에서의 원격 실험은 세계적인 이슈로 부상되면서 여기에 관련된 각종의 구상과 기술들이 차츰 구축되어 나가고 있는 현실이다. 본 논문은 이런 시각에서 출발하여 원격 실험의 중요한 부분인 다중상대 원격 모니터링 시스템의 구현에 포커스를 두고 있다. TMO 기반의 실시간 다중상대 원격 모니터링 시스템은 지역적으로 떨어져 있는 여러 개의 관찰 대상을 실시간 미들웨어를 사용하여 보다 정확하게, 다각도, 고화질로 보여주는데 목표를 두고 있으며 보안적인 차원에서 Grid 환경과 접목을 할 예정이다. 본 논문에서는 로컬에 있는 카메라와 네트워크로 연결되어 있는 데이터 수집자(Data Collector)의 구현을 바탕으로 기술 되어 있다. 데이터 수집자는 카메라/센서에서 발생하는 RAW 데이터를 직접 받아서 동기화 해주고 실시간으로 보여주는 기능을 하며 특정된 파일 포맷으로 데이터를 저장해주는 역할을 한다.

1. 서 론

실시간 시스템(real time system)이란 진행 중인 현상과 동일한 속도로 입력 및 출력이 진행되는 시스템이다. 다시 말해 데이터의 발생과 출력 및 처리를 짧은 시간에 처리하는 방식으로, 은행의 창구 업무나 좌석 예약 업무에서처럼 요구가 발생한 즉시 결과가 나오게 되어 있다. TMO[1](Time-Triggered Message-Triggered Object)는 Kane.Kim 등에 의해서 개발된 실시간 미들웨어이다. 실시간 시스템의 척도는 시간이다. 즉 시스템이 제공하는 시간의 정밀도가 어떠한 단위까지 가능한지에 따라 성능이 평가되고 처리 할 수 있는 일의 특성이 변하게 된다. 실시간적 특성이 다른 시스템과 구별이 되는 한 가지가 정밀한 시간정보이다.

본 논문은 TMO기반의 다중 상대 실시간 원격모니터링 시스템의 핵심 부분인 데이터 수집자(Data Collector)의 구현을 중심으로 기술 할 것이다. 전반적인 시각에서 보는 실시간 원격 모니터링의 특징은 아래와 같다.

- ①. 실시간 차원에서 진행되는 원격 모니터링 시스템으로서 10ms 단위의 시간적 정밀도를 제공한다.
- ②. 분산되어 있는 복수개의 관찰대상(각종 센서, 카메라 등)을 동기화 하여서 보다 의미 있는 데이터를 제공한다.
- ③. TMO 미들웨어를 토대로 개발한 응용프로그램으로서, 잘 구축된 환경위에서 사용자 지향적인 프로그램의 개발에 중점을 두고 있다.
- ④. 여러 개의 관찰 상대의 고도로 정밀한 동기화를 주목적으로 하고 있으며 장치에 대해 유연한 확장성을 가진다.

본 논문의 아래와 같은 구조로 구성되었다. 제 2장에

서는 관련연구에 대해서 기술하고, 3장에서 TMO기반의 실시간 다중 상대 원격 모니터링 시스템의 설계에 대해서 설명한 후 4장에서 구현에 대해서 기술한다. 제5장에서는 결론 및 향후연구에 대해서 기술한다.

2. 관련연구

2.1 TMO

TMO는 Kane.Kim 등에 의해서 개발된 Object Structuring Scheme이다.

TMO는 기존의 객체 모델을 경성 실시간 시스템에서 높은 효율성을 보일 수 있는 객체 모델로 확장하기 위한 연구에서 나온 결과이다. 따라서 TMO는 실시간 시스템이 가지는 시간적인 특성과 행동을 쉽게 추상화 할 수 있는 구조를 가지고 있을 뿐 아니라, 적시 서비스 능력(timely service capability)을 시스템 설계 단계에서부터 보장할 수 있다.

TMO의 특징은 아래와 같이 요약할 수 있다.

①. 분산 컴퓨팅 컴포넌트

TMO 모델의 설계 개념 중 가장 두드러진 특징은 "RTCS(Real-Time Computing System)는 항상 TMO들로 구성된 네트워크의 형태를 취한다."라는 것이다. 다시 말해서, TMO들은 서버에 있는 서비스 메서드에 대한 클라이언트 호출을 통해서 서로 상호작용을 한다. 이때, 멀티노드의 TMO객체들은 non-blocking 형태의 RMI(Remote Method Invocation)를 통하여 분산 처리를 수행한다.

②. Spontaneous method (SpM)

Time-triggered method인 SpM은 클라이언트의 서비스 요청에 의해서 실행되는 SvM과는 달리 TMO 설계 시에 명세한 시간이나 주기가 되면 실시간 클럭(clock)에 의해 자동으로 실행되는 메서드다. SpM의 시간 조건은 디자인 시에 Autonomous Activation Condition(AAC)에 상수로 명세 된다.

SpM이 스케줄 될 수 있는 방법에는 두 가지가 있는데 프로그래밍 시에 AAC를 정의하여 SpM이 정적으로 스케줄 되도록 하는 정적인 방법과, 설계 시에 다 수의 AAC를 선언하고, 시스템 수행 중에 후보로 등록된 AAC중 하나를 선택하여 SpM이 수행될 수 있도록 하는 동적 스케줄 방법이 있다.

③. Basic concurrency constraint (BCC)

TMO들의 시간적인 서비스 능력을 보장하기 위한 제약 조건으로써, SpM과 SvM이 공유데이터 ODS를 동시에 접근하려고 할 때 발생할 수 있는 충돌을 방지하기 위한 수행 규칙이다.

④. 종료시간과 데드라인 보장

디자이너가 메소드의 시작시간, 종료시간 그리고 데드라인을 명세함으로써 시스템의 적시 서비스 능력(timely service capabilities)을 디자인 단계에서 보장할 수 있도록 지원 한다.

현재 TMO 객체모델은 실시간 처리를 필요로 하는 군사, 공장제어, 교통, 그리고 멀티미디어 등의 응용과 실시간 시뮬레이션 분야에 적용되어 활발한 연구가 진행 중이다

2.2 CxImage 영상 처리 라이브러리

CxImage는 Davide Pizzolato가 개발한 영상처리 라이브러리로서, 2001년도 코드 프로젝트(Code Project)에서 시작하여 오픈소스로 성장하였다.

CxImage는 영상 읽기, 저장, 보기, 영상 변환 등을 구현한 C++ 영상처리 클래스로 구성되어 있으며, API 자체가 매우 간결하면서도 빠르다. 이에 비해 다른 라이브러리인 OpenL, FreeImage, PaintLib는 강력하고 완벽하며, 업데이트 주기가 빠르나 C를 기반으로 한 플랫폼 종속적이며, 컴파일러마다 환경을 재설정해야 하는 단점을 갖고 있다. 그리고 CxImage는 무료라는 매력, 그리고 잘 알려진 TIFF, JPEG, PNG, ZLIB, J2K 오픈 라이브러리와 연계되어 있다.

본 논문에서 구현한 시스템은 일차적으로 JPEG을 사용하며 차후 JPEG2000으로 구현 예정이다.

JPEG는 Joint Photographic Experts Group(정지화상 전문가그룹)의 약자로 현재 가장 많이 쓰이는 정지화면 영상의 규격중 하나이다. 컴퓨터, 전자카메라, 컬러팩스, 컬러프린터 등에 응용되는 영상의 저장 및 전송을 위한 효율적인 압축에 관한 국제표준(ISO-IEC10918)으로서 이 표준화를 담당하는 작업반의 별칭이기도 하다.

JPEG압축방식은 크게 무손실 모드와 손실 모드로 나눌 수 있는데 무손실 모드는 원 데이터에 전혀 손상을 주어서는 안 되는 경우에 쓰이고, 손실모드는 시각적으로 못 느낄 정도의 손실을 허용하면서 압축률을 높이는 대부분의 응용분야에 채택하고 있다.

JPEG2000은 기존 이미지 포맷인 JPEG의 2000년 버

전을 말하는 것으로 서로 다른 특성을 갖는 다양한 형태의 영상에 하나의 통합된 부호를 부여한다.

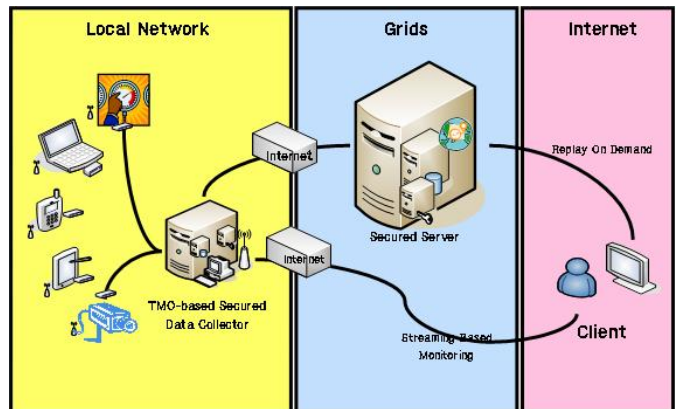
기존의 JPEG은 압축비율이 높은 반면 DCT를 사용하기 때문에 고주파의 컬러정보를 잃어버리거나 압축과정에서 형성된 블록들이 컬러의 경계선을 따라 나타나 영상을 훼손시킨다.

JPEG2000은 압축비율을 더욱 높이면서도 이미지 품질을 보존할 수 있는 방법으로 웨이블릿 기법을 이용해 DCT 압축으로 생기는 블록현상을 방지한다.

3. 시스템의 설계

3.1 TMO기반의 실시간 다중 상대 원격 모니터링 시스템

이 시스템은 크게 3개의 도메인으로 구성된다. Local Network, Grid, Internet 도메인에서 각각의 작업이 진행이 되며 중앙에 서버를 하나 두어서 중계 및 사용자의 요청에 응답을 준다. 기존에 있는 모니터링 시스템과 다른 점이라면 TMO 엔진을 토대로 그 위에 그리드 인증 체계를 더 함으로서 실시간 모니터링, 그리드 수준의 보안과 데이터 관리가 가능하게 해준다.

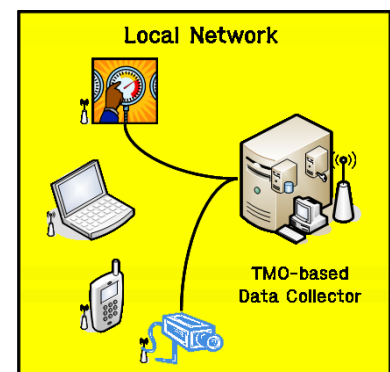


<그림 1>. 전체 시스템 구조

3.2 데이터 수집자

① 로컬 도메인의 다중 데이터 수집자.

TMO 엔진이 타임 매니저 역할을 을 하며, 10ms 단위의 정확도 제공을 목표로 한다. 이것은 모든 이동가능한

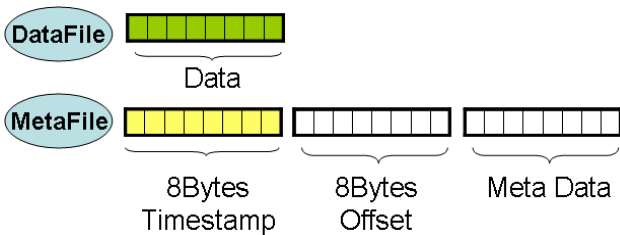


<그림 2>. 데이터 수집자 구조

센서를 쉽게 시스템에 연동할 수 있도록 있게 설계되었다. 이러한 다중 데이터 수집자는 GSI 인증을 통해 그리드 환경 상에 존재하는 데이터 저장소로 모니터링 데이터를 전송한다. 미세단위로 수집되는 데이터는 메모리에 버퍼링 되면서, 로컬 저장장치에 저장되며, 주기적으로 그리드 데이터 서버로 전송된다. 이렇게 하는 이유는 미세한 단위로 수집되는 데이터를 구조화하고 저장하는 데 있어서 TMO의 데드라인 제약과 네트워크의 전송시간 사이의 충돌을 피하기 위한 것이다. 파일 구조는 아래와 같다.

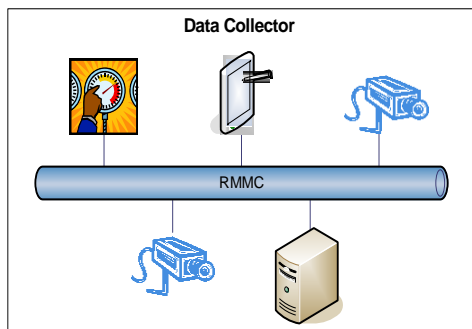
파일은 데이터 파일과 메타파일로 나뉘어 저장 된다. 여기서 메타 파일은 크게 세 가지 정보로 구성되는데 여기에는 영상 혹은 센서데이터의 Timestamp, 데이터파일 내부에서의 위치(Offset), 해당영상의 검색을 위한 메타 정보 등이 저장된다. 그 구조는 다음과 같다.

파일은 데이터 파일과 메타파일로 나뉘어 저장 된다. 여기서 메타 파일은 크게 세 가지 정보로 구성되는데 여기에는 영상 혹은 센서데이터의 Timestamp, 데이터파일 내부에서의 위치(Offset), 해당영상의 검색을 위한 메타 정보 등이 저장된다. 그 구조는 다음과 같다.



<그림 3>.파일 구조

구조적으로 볼 때 데이터 수집자는 하나의 기본 클래스를 가지고 하위 센서는 기본 클래스로부터 상속을 받는다. 실제 센서는 또한 기본 클래스 밑에 있는 센서 별 기초 클래스에서 상속을 받아 확장을 하여 센서의 추가를 최대한 쉽게 하였으며 시스템의 재개발 효율을 증가시켰다.



<그림 4> 데이터 수집자의 통신구조

통신방면에서 볼 때 센서와 데이터 수집자 사이에는 TMO 미들웨어가 제공하는 RMMC채널을 통하여 센서들이 데이터를 구조화 하여 주기적으로 채널에 공유하게 되고 이것을 데이터 수집자가 일정한 간격을 두고 읽어 가는 형식을 취하고 있다.

4.시스템의 구현

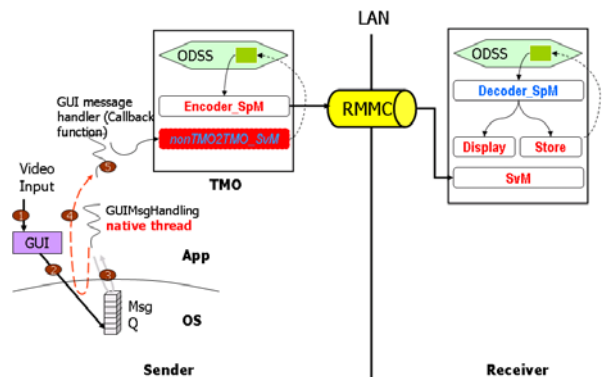
4.1 개발환경

운영체제: Windows XP with SP2 Professional
 H/W: Intel Core2Duo E6400, 2G RAM DDR2-6400
 영상장비: Logitech Quick Cam pro 4000
 TMO: TMO SL ver 4.21

동일한 사양의 PC 3대로 통신환경을 구성 하였으며 그 중 2대가 센서노드(영상 데이터 전송 자)역할을 하고 1대가 데이터 수집자 역할을 한다.

4.2 기본구조 및 구현

사용자의 영상데이터 입력은 아래와 같은 흐름에 의해서 처리된다.



<그림 5>.데이터 수집자의 기본 구조

데이터 수집자는 크게 송신측 하고 수신측으로 나뉘는데 영상장비/센서 하고 연결 되어 있는 부분이 송신측이 데이터 처리해주는 쪽이 수신측으로 된다.

사용자의 영상 데이터는 영상캡처 이벤트 발생시마다 TMO객체에 전송이 되며 전송된 영상은 Encoder_SpM을 걸쳐 메모리상에서 압축을 거치고 RMMC 채널로 데이터 처리노드 쪽으로 보내지게 된다. 수신측에 도달한 데이터는 Decoder_SpM을 통하여 복원되며 화면상에 보여주거나 저장된다.<그림 5>

여기서 송수신 될 때마다 각자의 시간 정보를 마이크로 초 단위의 정확한 Timestamp(표1)를 첨부하여서 보냄으로서 정확한 동기화에 튼튼한 토대를 마련해 준다.

```
class Timestamp{
int NodeID,MessageID;
.....
tmstp (int node_id, const C21_age& t = 0);
tmstp (const tmstp& tp);

void set (int node_id, const C21_age& t = 0);
void set (const tmstp& tp);
int get_node_id () const
void set_node_id (int node_id);
int get_message_id () const
void set_message_id (int message_id);
};
```

<표 1> Timestamp 정보

```
class TMO_Video_Send_Class: public CTMOBase
{
VideoCamODSSClass * m_pVideoCam;
WinCap_Video_Class VideoBuffer;
void CodecInit();
unsigned char FrameBuf[sizeof(SVideoFrame) +
size];
unsigned char CompressedBuf[Size];
unsigned char VideoMessage[MAX_MESSAGE_SIZE];
};
```

<표 2> 송신자 기본 클래스

표2는 송신자의 기본 클래스 구조로서 데이터 전송에 필요한 코덱정보, 메모리 버퍼 등에 대한 정의를 하고 있으며 수신자 측도 기본적으로 유사한 클래스 구조를 갖고 있다.

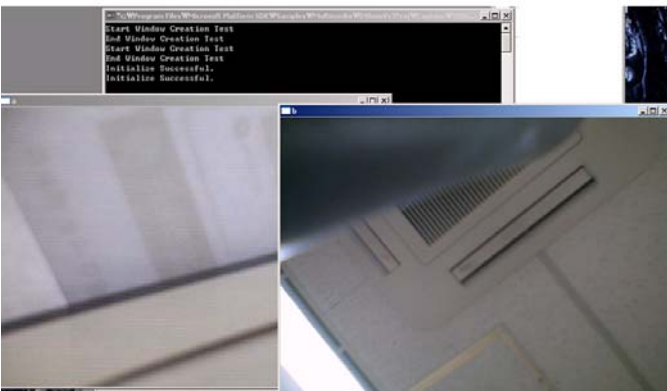
```
int TMO_Video_Recv_Class::TMO_Decoder_SpM()
{
tms Timestamp;
Recv(GetTimestamp,&pVideoFrame);

if(nFrameID > m_pDisplay->GetLastVideoFrameID())
{
m_pDisplay->SetLastID(pVideoFrame->nFrameID);
memcpy(CompressedBuf, nBytes);
// Set the codec format
pVCM_codec->SetCodecFmt();
// Decode a frame
pVCM_codec->Decode(FrameBuf, CompressedBuf);
// Display a frame
m_pDisplay->DisplayFrame((void*)FrameBuf);
}
}
```

<표 3> 수신자 디코더 클래스

수신자 쪽에서 주요하게 동기화에 필요한 Timestamp 정보와 영상의 환원에 그 기능을 하고 있다. 송신자로부터 전송된 시간정보는 Timestamp 객체 내부에 저장하고 수신된 영상정보는 디코딩을 거쳐서 화면에 보여진다.

4.3 실제 테스트



<그림 6>.데이터 수집자의 실행 화면

그림 6은 데이터 수집자의 실제 실행 화면이다. 두 개의 부동한 노드에서 전송되는 데이터는 하나의 수신자에 의해서 실시간으로 영상을 보여주게 된다.

4.4 시스템의 기술 문제점.

현재 개발 단계에 있는 본 시스템은 여러 가지 기술적 문제점을 갖고 있다.

첫째, 영상 처리의 오버헤드가 크다. 이는 주로 시간적인 면에서 민감한 실시간 시스템과의 문제다. 실시간 시스템은 짧은 주기 내에 신속하게 처리를 해주는데 현재 적용한 영상처리 메커니즘이 다소 시간을 많이 소요하는 병목현상을 보여주고 있다.

둘째, 미들웨어의 한계성이 존재한다. TMO로 전송 시 복합적인 환경적인 요소로 실제 성능과 이론적 성능의 차이를 보인다. 데이터의 사이즈가 커지면 통신이 원활하게 진행되지 않는다. 일정한 크기를 넘어가면 통신대상을 못 찾는 현상이 발생한다.

5. 결론 및 향후 연구계획

본 논문에서는 TMO기반의 실시간 멀티 타겟 모니터링 시스템 중 데이터 수집자의 구현에 포커스를 두고 있다. 영상/센서데이터를 목적으로 하는 수집자는 세밀한 동기화의 현실화에 전력을 다하고 있으며 다중 상대 모니터링에 대한 실시간 미들웨어에 기초한 시스템을 제안하고 개발하고 있다. 개발도중에 많은 제약점들이 발생하고 있지만 이것은 차후의 기술보완을 통하여 충분히 해결 가능한 문제들이다.

현재까지 만족할 수준은 아니지만 이 시스템은 멀티 타겟, 실시간 시스템, 그리드 환경 세 가지를 결합한 실제 응용모델로서 실시간 원격 모니터링에 대한 하나의 도전 과제이다.

향후 시스템의 안정성과 동적인 센서와의 연동, 초고속 통신망에서의 고화질 카메라(HD Camera)의 원격 모니터링에 대해서 연구할 것이며 또한 GPS를 통한 여러 개의 분산된 센서 망을 동기화하고 통합하는 방법에 대한 연구도 진행 할 것이다.

참고문헌

- [1] T M O S L _ v 4 . 2 1 _ M a n u a l
http://dream.eng.uci.edu/TMOdownload/
- [2]Kim, K.H., and Li, Y.Q., "Toward Easily Analyzable Sensor Networks via Structuring of Time-Triggered Tasks", Proc. FTDCS 2003 San Juan, Puerto Rico, May 2003, pp. 344-351.
- [3]Kim, K.H., Im, C.S., M.C. Kim, Y.Q. Li, S.M. Yoo, and L.C. Zheng, "A Software Architecture and Supporting Kernel for Largely Synchronously Operating Sensor Networks", in B. Kleinjohann et al. eds., 'Design Methods and Applications for Distributed Embedded Systems' Kluwer, pp.133-144
- [4]Kim, K.H., "Toward New-Generation Object-Oriented Real-Time Software and System Engineering", Invited paper, SERI Journal, Taejon, Korea, Vol.1, No.1, Jan. 1977, pp.1-23.

[5]Kim, K.H. and Subbaraman, C., "The TMO Structuring Approach and its Potential for Telecommunication Applications", Proc. ASSET '98 Richardson, TX, March 1998, pp.108-115.