

자율주행자동차 오픈플랫폼 온톨로지 구축을 위한 스마트디바이스 연구*

최 병 관**

A Study on Smart Device for Open Platform Ontology Construction of Autonomous Vehicles

Choi, Byung Kwan

〈Abstract〉

The 4th Industrial Revolution, intelligent automobile application technology is evolving beyond the limit of the mobile device to a variety of application software and multi-media collective technology with big data-based AI(artificial intelligence) technology. with the recent commercialization of 5G mobile communication service, artificial intelligent automobile technology, which is a fusion of automobile and IT technology, is evolving into more intelligent automobile service technology, and each multimedia platform service and application developed in such distributed environment is being developed. Accordingly, application software technology developed with a single system SoC of a portable terminal device through various service technologies is absolutely required. In this paper, smart device design for ontology design of intelligent automobile open platform enables to design intelligent automobile middleware software design technology such as Android based SVC Codec and real time video and graphics processing that is not expressed in single ASIC application software technology as SoC based application design. We have experimented in smart device environment through researches, and newly designed service functions of various terminal devices provided as open platforms and application solutions in SoC environment and applied standardized interface analysis technique and proved this experiment.

Key Words : SoC, Intelligent Autonomous Vehicles, Ontology, Smart Device

I. 서론

4차 산업혁명을 기반으로 인공지능자동차의 응용

기술은 이동수단의 한계를 넘어 빅데이터 기반으로 지능형자율주행과 패턴인식등 다양한 알고리즘을 통해 객체인식 기술과 연계한 첨단 디바이스와 응용 소프트웨어 등 멀티미디어 중심의 인공지능 융합기술로 진화하고 있다. 인공지능 기술의 진화에 따라 자동차 산업은 전장기술을 넘어 IT기술로 더욱더 지능화된 지

* 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW 중심대학지원사업의 연구결과로 수행되었음(2019-0-01219)

** 연세대학교 컴퓨터정보통신공학부

능형자동차서비스 환경으로 진화하고 있으며, 오픈소스 기반의 안드로이드를 탑재한 스마트폰 응용서비스라는 새로운 패러다임으로 많은 연구개발이 이루어지고 있다. 이러한 분산 환경에서 개발되어 있는 각각의 멀티미디어 플랫폼서비스와 애플리케이션이 개발되고 있음에 따라 다양한 서비스 기술을 휴대 단말장치 단일시스템, SoC로 개발되어져 응용 소프트웨어 기술이 한층 더 진화되고 있는 것이다[6].

본 논문에서는 지능형자동차 오픈플랫폼의 온톨로지 설계를 위한 스마트디바이스 설계를 통하여 단일형 ASIC 응용소프트웨어 기술에서 표현하지 못했던 안드로이드 기반의 SVC Codec과 실시간 동영상, 그래픽처리등 지능형자동차 미들웨어 소프트웨어 설계 기술을 SoC 기반의 응용 설계와 실험을 통해 스마트디바이스 환경에서 연구 하였으며, 오픈 플랫폼과 응용솔루션으로 제공되는 각종 단말장치의 서비스 기능들을 SoC(Software-on-Chip) 환경에서 새롭게 설계하고 표준화된 인터페이스 분석기법을 적용하여 연구를 하였으며, 지능형 오픈플랫폼 온톨로지의 다양한 융합환경을 실험과 분석을 통해 확인하였다.

지능형자동차 멀티미디어 융합플랫폼 연구의 필요성은 차량내 Embedded 통신 프로토콜을 통하여 차량과 무선통신망이 서로 결합된 대표적인 자동차 응용기술로 차량의 안전과 진단, 그리고 GPS/GPRS를 위치 기반서비스를 통해서 서비스 시장이 확산되어 급성장 하고 있는 것이다. 이러한 멀티미디어 기반의 차량 네트워크 기술을 적용하여 차량간 무선통신의 발전과 모바일 융합을 통한 서비스의 다양성이 확산되고 자동차와 IT기술이 융합되는 차량중심의 통합망 시스템 운용에 대한 오픈플랫폼의 온톨로지(Ontology) 구현을 통한 상호 인프라의 연동성을 추출하고 GUI를 통해 적용하는 다양한 토폴로지(Topology)를 구성하는 테스트 실험 환경을 가지고 있어야 한다.

지능형자동차의 오픈플랫폼 연구는 기존에 연구

<표 1> 지능형자동차 기술 분류

구 분		기술분류(중분류)
차량 안전 / 차량 정보화	액티브 안전시스템	예방안전, 충돌방지, 차량이격장치
		사고회피, 차선이탈감지, 졸음방지
		자율주행, 속도감지, 도로상황인식
		패턴인식, 자동주차, 사물인식
	액티브 패시브 통합차량 이동통신시스템	사고경감 및 탑승자, 보행보호
		4G/5G Mobile Communcation, ITS, 텔레매틱스, 차량간통신, 네비게이션(GPS/GPRS),
차량정보시스템	지능형정보시스템(운전정보시스템)	
	차량탑재, 네트워크 기술	
공동기반 기술	AI, HMI, HCI, Human Factor, Embedded S/W, H/W, HUD/HMD	
편의성	엔터테인먼트 시스템	디지털차량 AV시스템, 음원, VOD CAR-Infotainment,
	편의성 및 보안성	차량보안 및 탑승자 편의성

되어진 프로토타입 응용기술의 한계를 넘어 스마트 모바일과 오픈플랫폼 온톨로지(Ontology)를 적용하여 구성된 새로운 실험 연구를 적용함으로써 보다 다양한 Application의 기능을 효과적으로 확인하고 효율성에 대한 높은 성과를 도출할 수 있으며, 실험을 통한 결과값에 대한 비교 분석을 실용성 연구에 적용하여 그 결과 값에 대한 실증 연구 값을 얻어내어 입증을 할 수 있다.

본 논문의 구성은 지능형자동차 응용시스템에 대한 스마트디바이스의 기술을 정의하고 관련 오픈플랫폼에 온톨로지를 적용한 분석내용과 디지털 운행 기록장치 설계와 지능형자동차 응용Application 그리고 융합플랫폼 S/W구조의 구성방안등을 응용서비스별 스마트모바일 플랫폼서비스의 방법론으로 정의하고 제안을 하였으며, 지능형자동차의 원격제어를 통한 인공지능 스마트-카(Smart-CAR)의 관련연구와 각각의 R&D 시스템사업 모델등 서비스구축 내용을 제

안한다.

II. 프로토콜 오픈프레임

본 연구는 인공지능의 효율성과 기능성을 적용하여 안드로이드 기반의 운영체제(O/S)를 기반으로 차량 내 네트워크를 지능형자동차 오픈 플랫폼으로 설계하고, 차량내 통신방식은 직렬 데이터버스를 통해 분산된 전자 모듈간 데이터 전송 방식으로 설계를 하였다[5].

지능형자동차 모듈 간의 네트워크 방식은 전용방식인 P-to-P(Point-to-Point)통신 직렬버스 방식을 적용하여 한 가닥의 선으로 시분할 멀티플렉싱을 통해 신호를 조합함으로써 차량내 필요한 통신선의 수를 줄일 수 있다. 그리고 정보는 ABS, 시트제어, 온도조절, 대쉬보드 디스플레이 등 각 기능을 제어하는 각각의 제어 모듈로 보내어 처리한다[5].

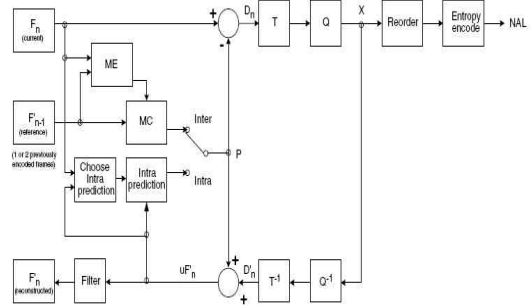
2.1 스마트디바이스 영상 압축

영상기술의 초기 영상 압축기술은 MPEG-1을 표준으로 압축하고자 하는 영상의 전체를 압축하여 하나의 프레임으로 적용하여 사용 하였으나 영상기술의 다양한 진화와 고 사양의 응용프로그램과 플랫폼 다양성에 따라 진보론(Progressive)적인 방식을 채택해서 인터페이스 방식으로는 TV영상을 저장할 수 있을 정도의 단순한 효과를 나타낼 정도의 기초적인 효과만을 나타내지 못했다[8].

그러나 다양한 서비스와 플랫폼의 등장으로 HCI(Human Computer Interaction)프로토콜과 알고리즘이 연구되고 산업에 활용되면서부터 영상분야의 프로세스 적용방식을 Interface 방식의 영상을 효과적으로 처리할 수 있도록 한 표준 MPEG-2가 제안되어 기존의 NTSC TV방송용 자료 뿐만 아니라, 위성TV,

CATV 및 DVD영상등을 압축하는 표준으로 널리 사용되게 되었다. 영상 압축 표준에서 그 압축률은 표준을 사용하는 응용의 경제성을 좌우한다. 수백개 채널의 영상이 실시간으로 네트워크를 따라 전송되고 가공되며, 최종적인 가정으로 전달되는 과정에서, 압축률의 향상은 영상전송 및 저장에 요구되는 비용을 절감할 수 있다.

영상압축 기술은 압축률 향상을 위해서 마이크로소프트사에서 비디오 압축 표준으로 개발한 VC-1(공식명칭 SMPTE 421M)을 사용하여 환경시험에 적용을 하여 실험을 하였으며, 압축 효율은 H.264/MPEG 4 AVC와 비슷하고 모바일 기기에서 고선명 텔레비전(HDTV)까지 적용할 수 있도록 다양한 프로파일과 레벨을 지원하여 고선명 DVD (HDDVD) 및 블루 레이, X박스 360 등에서 사용될 수 있도록 표준으로 채택하여 제안 되었으며, JVT에서는 또 다른 압축 표준인 MPEG-4 Part 10 (H.264) 이 제안되었는데, 이들 새로운 표준은 MPEG-2에 비해서 2배 이상의 압축률을 제공한다.



<그림 1> MPEG-4/H.264(PART-10) Encoder 구성

VC-1은 미국의 SMPTE(Society of Motion Pictures and Television Engineers)에서 표준 비디오 압축 방법의 하나로 2006년에 승인되었고, H.264는 2003년 ISO/MPEG의 공동 표준 방법으로 승인된 후 몇 번의 수정단계를 거쳐서 오늘에 이르고 있다.

2.2 지능형자동차 온톨로지 신호구성

본 논문에서의 지능형자동차 스마트디바이스 오픈 플랫폼 온톨로지 구성은 각각의 신호단말 기술에 있어서 서로 불편함 없이 오픈 플랫폼 환경에서 기술을 적용하여 자동차의 Automotive-IT 기술을 적용한 Smart-CAR 오토 드라이빙 기술을 각각의 단말 플랫폼 장치들에 적용을 하였고, 이들 프로토콜과 API를 연동하여 지능형자동차 테스트베드 시스템에 적용하여 플랫폼 설계에서 사용하였다.

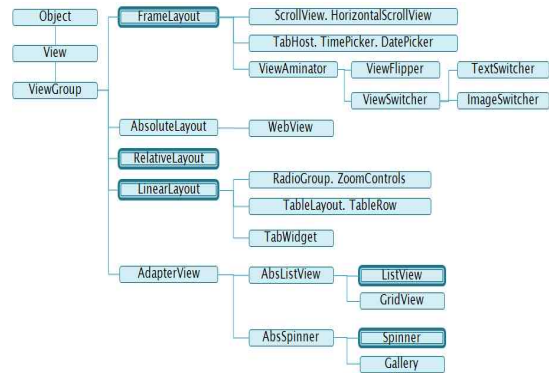
<표 2> 지능형자동차 Ontology Symantic 구성

Smart-CAR Ontology	Symantic Agenda	Ontological Engineering Item
차량 간략화 모델	가속엔진	<ul style="list-style-type: none"> 전형적인 가솔린 엔진의 경우 다음과 같은 간이모델을 사용할 수 있는데 계산된 토크와 실제 토크의 유사성 확인
	트랜스미션, 기어, 휠	<ul style="list-style-type: none"> 트랜스 미션과 파이널 기어는 기어비가 i, 효율은 η 인 덩어리 부품(lumped component)으로 모델링실행
	차량역학 방향	<ul style="list-style-type: none"> 오른쪽 부분은 각각 휠에 작용하는 힘, 공기저항력, 구름저항력, 경사 저항력행
에너지 효율기반 연비계산	등가연료 기반 구간연비	<ul style="list-style-type: none"> 배터리의 SOC 변화량을 등가연료로 계산하여 총 연료사용량에 반영 적용 배터리 SOC변화량, 모터과워 대비 연료의 변화량을 lookup table 구성
	에너지 효율기반순간연비	<ul style="list-style-type: none"> 엔진의 연료 소모량, 배터리의 SOC 변화량, 차량의 운동에너지 변화량을 종합하여 차량 에너지를 계산한 식이다. 차량에너지를 연료의 발열량(kWh/kg)과 연료밀도 (kg/l)를 이용하여 적용
	CASE	<ul style="list-style-type: none"> 일정시간동안 이동거리, 초기속도, 최종속도가 나타날 수 있는 속도레직
E-CHO Driving	최적의 속도	<ul style="list-style-type: none"> 타행과 역행에 대한 후방 속도레직을 구한 다음 실행
	Driving 패턴	<ul style="list-style-type: none"> 최적의 속도레직과 실제속도 레직에서의 에너지소모량 예코드라이빙 지수

<표2>는 지능형자동차 Ontology Symantic Web 구성을 표로 나타낸 것이다. 각각의 단말장치에서 제공하는 기기데이터의 Message Agenda를 세분화하여 구분하고 Engineering에서 필요한 데이터유형의 분류를 보여주고 있다.

2.3 미들웨어플랫폼 온톨로지 View

안드로이드 기반의 인공지능자동차 오픈플랫폼은 Symantic Web을 통해 스마트디바이스와 미들웨어로 온톨로지 View를 설계하였다. 지능형자동차의 적합한 미들웨어는 Embedded 응용프로그램을 기초로 설계하여 적용 하였으며, 지능형자동차 내에서의 멀티 미디어 플랫폼과 정보를 동시에 제공할 수 있도록 연구를 하였는데, 시스템 환경은 자율주행, 멀티미디어 콘텐츠와 네비게이션, 전자결제시스템, 도로교통 정보등의 교통정보망(ITS)시스템서비스를 연동하여 실험, 테스트환경을 제공한다. 아래 <그림3>은 온톨로지 플랫폼의 전체 View Group을 나타내었다.



<그림 3> 온톨로지플랫폼 View Group

최근 스마트폰 애플리케이션들이 다양하게 출시되고 있음에 따라 지능형자동차 카-인포테인먼트시스템과 이를 활용하기 위한 미들웨어플랫폼등 모바일 단말장치 수요에 대한 다양한 Embedded S/W설계와 주변 기기들과 연동할 수 있도록 표준화된 API를 적용하여 테스트베드(Testbed)환경에 적용하여 안드로이드 기반의 미들웨어를 Smart-CAR에 맞는 온톨로지를 구현하고 다양한 연구를 통해 실험, 적용하게 됩니다[7]. 아래의 <그림4>는 온톨로지 S/W Firm ware 프로그램을 나타내었다.

```

This program contains SPI routines in firmware (not for SSP).
Written for Hi-Tech PICC compiler vX.XX
*****
#include <pic.h>
#include "SPI.H"
*****
* Function Name: WriteSPI
* Return Value: None
* Parameters: Single data byte for SPI bus
* Description: This routine writes a single byte to the SPI bus.
*****
void WriteSPI(unsigned char DataOut)
{
    unsigned char n;
    mSCK = 0; //Make sure clock is low
    for(n=0; n<8; n++)
    {
        mSCK = 0; //
        if((DataOut << n) & 0x80) //Clock a 1 DFFE
            mSQ = 1;
        else //Clock a 0
            mSQ = 0;
        mSCK = 1; //Pulse clock
        // mSCK = 0; //
    }
}
*****
* Function Name: ReadSPI
* Return Value: Data[n] value
* Parameters: void
* Description: Read single byte from SPI bus.
                Note: Clock comes in low and slave is
                already driving data.
*****
unsigned char ReadSPI( void )
{
    unsigned char n, DataIn = 0;
    mSCK = 0; //Make sure clock is low
    for(n=0; n<7; n++)
    {
        mSCK = 1;
        DataIn = DataIn + mSI;
        DataIn = (DataIn << 1);
        mSCK = 0;
    }
    //last bit
    mSCK = 1;
    DataIn = DataIn + mSI;
    mSCK = 0;
    return DataIn;
}

```

<그림 4> 온톨로지 Firm ware S/W프로그램

2.4 인공지능 자율주행 오픈플랫폼

인공지능자동차내 Embedded를 구성하는 카-인포테인먼트(CAR-Infotainment)시스템은 차량에 OBD 전자식(Digital) 운행 기록장치의 단자에 연결되는 ELM327 칩(Chip)을 통해 인공지능의 자율주행 기반이 되는 차량이탈방지장치, 졸음방지장치, 자율주행장치, HUD/HMD, 패턴인식, 형태소분석등 기타 차량정보 수집모듈과 연계되는 안드로이드 기반의 차량정보시스템을 통해 뷰어 애플리케이션 모듈로 구성하여 연동되도록 설계하였다.

안드로이드 미들웨어를 통해 스마트폰과 태블릿 PC등과 같은 모바일 단말장치의 운영환경과 연동하

<표 5> 자율주행 셀프 오토드라이빙

항 목	세부 내용
Applications(Java/HTML/JavaScript)	<ul style="list-style-type: none"> 안드로이드 플랫폼의 내장 애플리케이션
Application Framework	<ul style="list-style-type: none"> 안드로이드 플랫폼 라이브러리 (android.* 패키지) 표준 자바 라이브러리(java.* 패키지) 외부(Third-party) 라이브러리 (org.apache.* 패키지)
Android Runtime (Java, Native)	<ul style="list-style-type: none"> Dalvik 버추얼 머신 코어라이브러리(dalvik.*, java.lang)
Libraries (Native)	<ul style="list-style-type: none"> 안드로이드 플랫폼의 내장 라이브러리(Webkit, SQLite, etc)
HAL(Native)	<ul style="list-style-type: none"> 그래픽, 오디오 · 비디오, USB
Kernel Driver (Native)	<ul style="list-style-type: none"> 안드로이드 리눅스 커널 안드로이드 드라이버

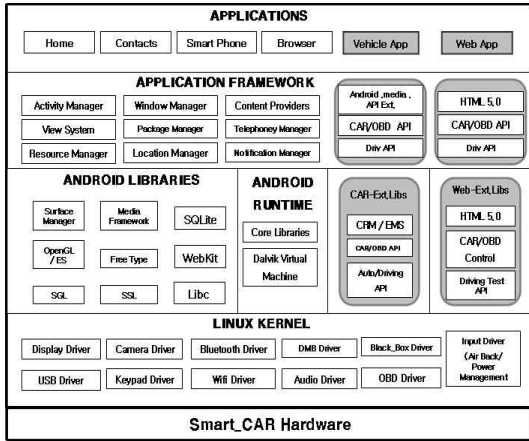
여 리눅스 커널과 함께 탑재되는 소프트웨어 프레임워크를 제공하고 이를 통하여 인공지능과 자율주행에 필요한 서비스를 제공한다[2].

최근 세계 최초로 국내의 5G 이동통신망 통신서비스가 개통됨에 통신 인프라에 따른 안드로이드 기반의 미들웨어와 오픈플랫폼을 차량정보시스템과 연동하여 스마트폰과 태블릿PC, 기타 안드로이드 플랫폼 장치의 모바일 통신과 연계한 단말장치를 연결함으로써 다양한 주변기기와의 연동 활용할 수 있는 장점을 확인 할 수 있다.

2.5 지능형자동차 오픈플랫폼 시험환경

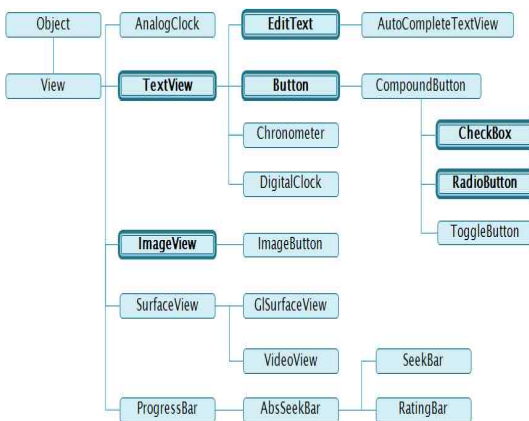
본 연구에서는 인공지능 지능형자동차 오픈플랫폼을 설계하고 온톨로지를 구현하기 위하여 안드로이드 기반의 지능형오픈플랫폼 적용하여 스마트디바이스 미들웨어시스템을 구축하여 연구를 하고, 지능형 자동차에서 필수적으로 적용되어야 할 기능들을 설계하여 실험환경을 구축하여 적용하였다.

실험 방법은 다음과 같이 두 가지의 패턴을 기반으



<그림 6> 실험중인 안드로이드 미들웨어

로 설계를 하였다. 첫째는 지능형자동차의 운행정보를 위해 확장해야 할 모듈들을 신규로 설계하였으며, 둘째는 안드로이드 기반의 오픈플랫폼을 연동하여 온톨로지 구현을 기본적으로 포함된 웹 기반의 플랫폼이 지능형자동차 모듈과 연동한 인터페이스가 되도록 신규 API를 설계 하였다. 본 연구에서는 안드로이드 기반의 지능형자동차 미들웨어를 직접 설계 개발하여 스마트폰 애플리케이션과 연계한 자율주행운행정보에 대한 연구를 SoC 기반에서 실험하였다.



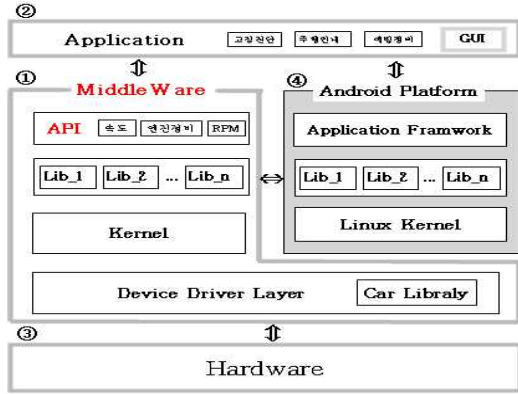
<그림 7> 지능형자동차 Ontology Widget

스마트디바이스 안드로이드는 리눅스커널 위에 안드로이드 애플리케이션 구동을 위한 소프트웨어 계층(라이브러리, 안드로이드 런타임, 애플리케이션 프레임워크)이 올라간 형태로 구성되어 있다[8]. 일반적으로 안드로이드를 탑재한 리눅스커널(Linux Kernel)과 애플리케이션 프레임워크(App lication Framework), 라이브러리(Libraries) 임베디드 안드로이드 소프트웨어 계층을 통칭하여 “안드로이드”라고 한다[10].

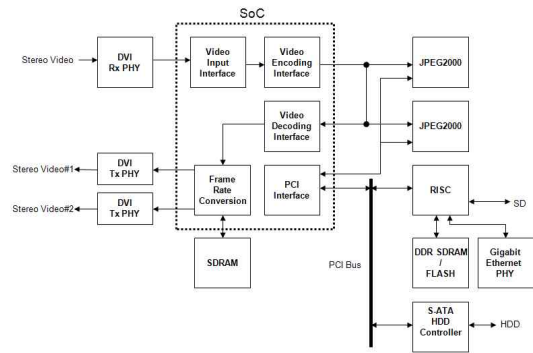
안드로이드를 플랫폼에 특정한 스마트-카 (Smart_CAR) 하드웨어에 이식하기 위해서는 리눅스 커널과 안드로이드를 S/W 환경을 준비해야 한다. 안드로이드 기반 플랫폼은 가장 윗 부분은 애플리케이션으로 이루어져 있으며, 지능형자동차의 각종 단말장치와 스마트폰용 애플리케이션등을 서비스를 나타내고 있으며, 그 아래 Application Framework와 Android Runtime과 Linux Kernel로 이루어져 있다.

가장 아래 부분에는 자동차의 차체 플랫폼인 Smart CAR Hardware로 이루어져 있는데, 자율주행자동차의 오픈플랫폼 미들웨어 구현을 위해서 본 논문에서는 안드로이드 각 계층과 지능형자동차용 모듈을 연계시키기 위하여 추가로 API를 설계하였다 [9].

본 연구에서 설계한 안드로이드 오픈플랫폼을 이용하여 지능형 자율주행자동차의 소프트웨어와 하드웨어플랫폼 각 계층에 지능형 자동차에서 사용 될 안드로이드 계층별 미들웨어 모듈과 API를 설계하여 정의하였다. 지능형자동차 오픈 플랫폼 기술은 IT기술과 접목하면서 기능면에서 다른 산업의 제조기술에 비해 발전속도가 상당히 빨라지고 있다. 특히 전장기술은 최근 모바일 이동통신의 기기서비스 발달 등으로 스마트-폰에서 사용되는 각종 Application과 연계하여 CAR-Infotainment 인공지능 기술로 발전하고 있는데, 그러한 원인은 다음과 같다.



<그림 8> 지능형자동차 오픈플랫폼 구성요소



<그림 9> JPEG-2000 Codec SoC 구성도

첫째, 운전자 습관분석과 고장진단 정보수집 및 사고 영상정보 저장과 사고발생 정보를 전송하는 Trip 리포트 통계 기능을 적용한 DataBase를 설계하고 적용한다[7].

둘째, 자율주행시 에코드라이빙을 위한 예방정비 기능과 고장진단 안내등 지능형자동차에서 제공할 수 있는 위급상황에 대한 응급처치 및 e-Call 서비스를 지원하는 스마트폰 패드기능을 설계하여 차량내 인포테인먼트(Infortainment)를 위한 표준 오픈 플랫폼과의 연계된 운행자의 운전습관이력에 대한 분석 정보 통계와 차량관리 및 사고 발생에 대한 응용 처리시 대책방안을 미리 수립하고 위급상황에 대한 상태정보를 운전자에게 스마트폰으로 문자안내를 제공하도록 확인하고 설계한다.

셋째, 자율주행자동차(Autonomouse Vihicle) 실현을 위한 차체정보 모니터링, 자가진단, 차량내공조 제어등의 기능을 제어할 수 있는 ‘OBD 전자식(Digital) 운행기록장치’ 적용방안과 스마트폰 애플리케이션연계하여 S/W로 분석한 에코드라이빙 데이터 값을 운행분석 프로그래밍 처리를 할 수 있도록 사전에 설계하여 구현하였다.

아래의 <그림 9>는 JPEG-2000 Codec SoC구성도를 나타내고 있다.

III. UX EMS시스템 구조

본 연구에서는 자율주행 통합운용관리를 위하여 사용자경험 디자인을 적용한 임베디드 소프트웨어의 UXI(User Experance)Element Management System시스템을 설계하고 프로그래밍하여 외부로부터 모바일 (Mobile) 이동통신망에 접속되어 있는 망관리 접속을 시스템 전체에 통합 운용할 수 있도록 상태정보를 Alam Event를 구성하여 관리를 한다[11].

5G통신망에서 제공하는 각각의 통신 아키텍처를 분석하고 Mager와 Agent는 P-to-P(Peer_To_Peer) 개념으로 GUI상에서 이들간의 통신 프로토콜 메시지를 SNMP를 사용하여 시그널을 내려주고 상호 통신을 한다. EMS는 UX(User Experence)의 사용자경험을 통한 통합망 관리의 논리적 체계를 계층구조 측면에서 EMS(Element Management System)에 해당되도록 GUI통하여 전체 시스템을 통합, 운용 관리하고 시스템의 안정성과 확장성 그리고 서비스 운용에 대한 용이성 측면에서 자율주행이 순조롭게 이루어지도록 프로세스와 알고리즘 체계화하여 설계하였다.

3.1 UX EMS 기본설계

본 연구에서 프로토타입으로 시험중인 지능형자동차 시스템의 운용설계는 EMS의 망 구성을 기본으로 탑재 설정하고, 통합망시스템을 이용한 기본설계는 UX/GUI를 이용하여 통합망 관리시스템을 통해 EMS를 체계적으로 조작하고 운영하도록 Manager의 기능과 성능을 보장하며, Manager의 영역에서 담당하고 있는 관리 자원에 대해서는 종합적이고 체계적인 망 관리 운용 능력을 가질 수 있도록 고급 프로그래밍 언어인 JAVA Script를 이용하여 설계하고 각각의 어플리케이션 실행 환경은 XHTML, ECMA Script, CSS 등의 해석기와 Mib API 구현을 통하여 구성되도록 설계하였다.

소프트웨어 프로그래밍 설계에 있어서 공통 인프라스트럭처는 JAVA기반의 어플리케이션 실행환경과 웹기반 어플리케이션 실행환경을 지원하기 위한 공통요소로 UX EMS를 구성하며, 각각의 관리기능과 DBMS의 연결포인트는 차량내부 통신망인 TCP/IP와 CAN통신등 IPC 방식을 이용하여 통신프로토콜과 연동하고 실행할 수 있도록 구성을 하였다.

또한 일반 관리(GM: General Management)는 어플리케이션 실행후 GUI의 운용을 위한 웹 기반 어플리케이션과 각종 프로세서의 관리를 수행할 수 있도록

설계가 되어 있으며, 감시결과는 DBMS에 저장되어 설정된 기준을 위반하면 사고발생에 대한 알람경보(Alarm Event)를 발생시키고, 결과는 그래픽(Graphic) 처리하게 된다.

<그림 10>는 UX EMS와 시스템간의 인터페이스의 구성도를 나타내고 있다.

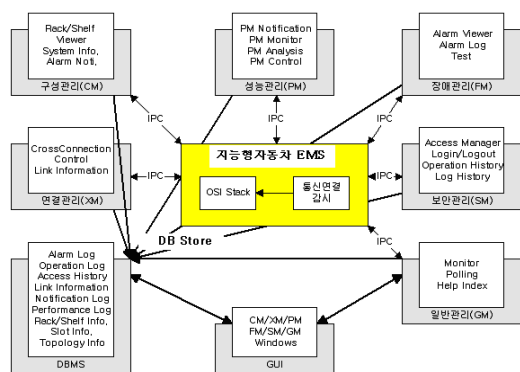
IV. 실험방법 및 절차

4.1 테스트베드 실험

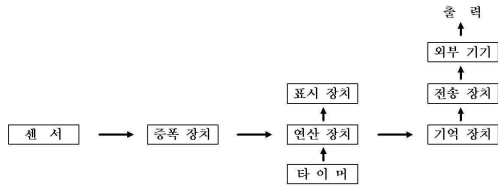
본 실험은 자율주행자동차 오픈플랫폼의 안드로이드 운영체제를 기반으로 온톨로지를 설계하고 지능형자동차 오픈 플랫폼을 차량 내부의 제어신호와 차량 정보를 실시간으로 신호를 전달하는 스마트폰용 미들웨어를 탑재하여 하였다. 본 실험에서는 자동차 디바이스플랫폼에 OBD 디지털(Dital)운행기록 장치를 장착하여 디지털기록장치(OBD-II), 테스트베드 플랫폼을 차량 내부 네트워크 출력장치인 STM3210C_EB보드 테스트베드(Testbed)에 연동하여 디지털(Dital)운행기록과 자율주행의 오픈 플랫폼을 연결하는 신호체계 시그널 장치를 장착하여 차량내 데이터 통신흐름을 파악하고 자율주행시 지능형자동차 운영을 감지하고 기록하는 주행정보의 흐름을 확인하고 기록하도록 설계를 하였다.

<그림 11>은 자율주행 기록장치의 신호처리 구조에서 자동차전장 S/W, H/W와 시스템의 통합 운행 정보를 저장, 관리하는 ECU의 “센서 연동모듈”을 나타내고 있으며, 자동차의 On/Off 신호제어 기술과 “증폭장치 모듈”을 통해서 차량의 상태정보와 출력 장치의 인지 정보를 감지하게 된다.

자율주행자동차의 지능형모듈은 디지털운행기록장치와 스마트폰과의 통신방식은 Bluetooth 통신을 통해서 신호체계를 연동하며 본 연구를 위해서 개발한



<그림 10> UX EMS 시스템인터페이스 구성도



<그림 11> 자율주행 기록장치 신호처리 구조

전용 미들웨어를 차량에 탑재하여 인공지능으로 디지털기록장치의 운행기록 정보를 감지하고 프로토타입 운용에 대한 주행실험을 연동하여 “연산장치 모듈”과 “타이머 모듈”, 그리고 “표시장치 모듈”등의 신호 처리를 통해 자율주행 애플리케이션이 오픈플랫폼을 통해 순조롭게 실험항목들을 수행 할 수 있도록 소프트웨어를 프로그래밍하고 플랫폼을 설계하여 나타내었다.

4.2 SoC 신호구성

본 실험에서 설계한 Multimedia용 SoC는 기존의 ASIC과 VLSI는 시스템 레벨의 설계가 여러 반도체에서 이루어 질 경우 시스템이 요구하는 각 반도체의 스펙에 맞춰서 설계가 이루어졌다. 그러나 본 연구에서 설계한 SoC는 기존의 프로세서에 음성, 화상, 영상, 그래픽 등을 효과적으로 처리하기 위한 하드웨어 가속 기능이 추가되고, 멀티미디어 데이터들의 입출력을 위한 표준화된 인터페이스를 포함하는 특화된 설계기법을 적용하여 실험에 적용하였다.

<그림 12>의 ①은 실험을 위한 지능형자동차 전용 미들웨어(Middleware)를 나타내고 있다. 내부의 시스템 설계기준 방식은 자동차전용 라이브러리를 생성하고 관리해주는 “Device Driver Layer”가 있으며,

Video, TCP/IP Protocol 스택, H.264 SVC Codec Driver 및 Display Driver를 관리해주는 “Kernel Layer”, 있다. 자동차의 속도_API, 엔진정비_API, RPM_API, 온도_API등 차량의 각종 Application

<표 3> 멀티미디어 Soc 신호처리의 특징

항 목	세부내용
입체영상 신호처리	<ul style="list-style-type: none"> Full HD 입체영상 자동 주 시각 제어 왜곡보정, 영상선명도 개선, 저조도 보상 다양한 입체 모니터지원(Full Side by Side, frame Sequential, half Side by Side, Top and Bottom format) 및 포맷 변환
입체영상 저장 및 전송	<ul style="list-style-type: none"> 입체영상 압축 / 저장 / 전송 / 복원 Full HD H.264 Codec (60fps) Full HD급 실시간 영상녹화 기능
Video Standard 영상입력 및 출력	<ul style="list-style-type: none"> DVI/Analog RGB -> PC, 영상카메라, 실시간 압축 원격지 전송기능 Composite/S-Video/Component/HDMI USB / 1394 (PC Mother Board)
User Interface	<ul style="list-style-type: none"> USB Front Remocon 1024 x 768 LCD with Touch Screen
Gigabit Ethernet	<ul style="list-style-type: none"> PC Mother Board
Security Engine	<ul style="list-style-type: none"> AES-128(PC S/W)

Layer를 담당하는 지능형자동차 전용 미들웨어가 실행된다[4, 12].

<그림 12>의 ②는 지능형자동차의 전장기술과 주행기능으로 차량정보에서 나타낼 수 있는 고장진단, 운행정보, 차량성능, 예방정비, 등을 나타낼 수 있는 각종 Application 서비스정보 기술을 처리한다.

<그림 12>의 ③은 지능형자동차 하드웨어 플랫폼에 대한 차량 차체의 Hardware Layer를 정의했다.

<그림 12>의 ④는 스마트단말장치를 통해 차량의 상태정보를 독립적인 "Android Platform"을 실험환경을 통해 각종 API와 View정보등 UX를 통해서 나타낼 수 있도록 성능 및 상태정보의 인공지능 운행기록에 대한 실험, 정보를 나타낸다.

4.3 SoC UX 설계구성

가. 스마트디바이스 온톨로지

자율주행 스마트디바이스 온톨로지의 설정값과 플

랫폼을 SoC환경에서 자율주행을 통해 운행정보기록에 대한 변수값과 실험결과를 실시간 데이터를 통해 수신처리 하도록 안드로이드 미들웨어를 통해 스마트폰 애플리케이션과 디바이스 모듈 오픈플랫폼으로 미들웨어의 운행정보 기록내용을 실시간으로 분석하게 된다.

그 결과값은 빅데이터를 통해 테스트베드 백업서버에 저장처리 하며 자율주행시 운행기록을 통해 정상적인 실험값과 다양한 환경값은 변수데이터를 추출하도록 오픈플랫폼을 설정하고 항목별 실험을 통해 약 2,500건의 자율주행 테스트 시험값은 주행실험을 통해 다양한 운행정보 값을 얻어내어 저장할 수 있었다.

나. 스마트디바이스 운행기록, 저장처리

본 연구를 통해 실험한 지능형자동차 오픈플랫폼은 온톨로지 구축을 위한 스마트디바이스와 안드로이드 기반의 디지털 운행기록장치 오픈 플랫폼을 통해 SoC환경에서 자율주행자동차의 패턴인식과 운행기록에 대한 주행정보를 스마트디바이스 온톨로지를 설계하여 실험에 적용하였다. 본 실험은 자율주행 실험결과에 대한 스마트디바이스 온톨로지 출력 값은 아래<표 4>의 자율주행 실시간 운행정보처리 결과값을 통해 축적된 데이터값을 빅데이터에 저장하여 자율주행 토폴로지 실험등 클라우드 환경으로 저장 관리하여 실험에 활용될 것이다.

• 지능형자동차 SoC기반 Application 적용실험, 연구



<그림 12> 스마트디바이스 자율주행 온톨로지

<그림 12>과 같이 스마트디바이스 지능형자동차는 디지털운행기록장치와 스마트-폰 미들웨어의 연결을 통해 전체 애플리케이션을 스마트-폰을 통해 운행기록장치의 정보상태를 안드로이드 미들웨어를 통해서 실시간으로 전송받고 운전자가 어플리케이션으로 직접 운행기록을 확인할 수 있도록 자바(JAVA) 언어로 설계하였다.

<표 4> 실험결과 데이터베이스 관리

실험 결과	데이터베이스 저장/내용
운행 기록/결과	▶ 과속 분석보고서, 급제동 분석보고서, 장기과속 분석보고서, 급가속 분석보고서
블랙박스 전송데이터	▶ 운행내역, 통계, 운전성향, 고장이력, 일별운행 기록, 최고속도, 평균속도, 엔진 공회전 시간, 정체시간, 시동 On/Off시간
사고기록	▶ 필수저장 데이터항목, 선택저장데이터 항목
운행기록	▶ 일 주행(24시간)분석조회, 구간 상세조회
운행속도 분석보고서	▶ 운전성향 및 주행속도에 따른 분석보고서, 구간 상세보고서
운행기록 저장	▶ 주행기록 DataBase 저장
운행분석 프로그램	▶ 운전성향에 대한 그래프 출력, ADT-Viewer 차량별 운행 상황표
네비게이션 운행분석	▶ 네비게이션 운행에 대한 ADT-Viewer 운행구간 데이터 저장

V. 실험연구

5.1 시험환경

본 논문에서 설계한 안드로이드 기반의 오픈 플랫폼 미들웨어 구성 요소를 검증하기 위한 다음과 같은

실험 환경을 적용하여 실험을 하였다. <표 5>는 실험 연구시스템 환경 내용을 나타낸 것이다.

<표 5> 실험연구 시스템 환경

내용	설 명
실험 플랫폼	실험에 연구에 적용한 자동차디바이스플랫폼에 OBD 디지털(Dital)운행기록 장치를 장착하여 테스트베드(Testbed) 플랫폼으로 실험장착 하였다.
검증 내용	안드로이드 기반의 오픈 플랫폼디지털 운행기록장치와 연계한 카인포테인먼트 영상데이터 조회기능, 수신, 저장, 에코드라이빙 주행안내 서비스등, 운행기록 관리가 원활하게 동작 하는지를 검증 함
적용 칩셋	Microchip MCP6541/2/3/4 레퍼런스보드 MMDL914Ta / High-Speed Switching Diode
운영 체제	OSEK/VDX 차량전장 전용 운영체제 Microsoft Windows 2018/ 데스크탑 PC서버
메모리	Samsung DDR4 16GB PC4-21300
PC	Intel_코어8세대 i7-8700 HP/Server
주변 장치	디지털기록장치(OBDII), STM3210C_EB보드 UNICORN(IDE & 600 SATA컨버터, 케이블)

5.2 시험구성

본 연구에서 디지털운행 기록계의 성능은 주행 속도계부의 허용 오차 즉, 표준주행속도계 지시값의 $\pm 2\%$ 이내로 적용 하였으며, 운행 기억 장치의 각 부에 이상이 없도록 환경구성을 하였다. 또한, 표준 주행 속도계 지시값이 $\pm 2\%$ 이내이어야 하나 운행거리 기록의 기록오차는 운행 거리 100 km에 대하여 2 km 이내로 셋팅하였다.

5.3 온도, 전압특성

본 연구에서 지능형자동차 디지털 운행기록장치의 지시 속도는 주위 온도 $20\pm 2^\circ\text{C}$ 에서 60km/h가 되도록 하고, 온도를 $-20\sim 60^\circ\text{C}$ 의 범위로 변화시켜, 공칭 전압이 12V인 것은 11~15V, 24V인 것은 22~30V의 범위에서 전압을 변화시켜 시험형식을 나타낸 것이다. <표 6>는 온도-전압 특성 값을 나타낸다.

<표 6> 온도-전압특성

순간속도 기록의 변동	운행거리 기록의 변동	운행시간 기록의 변동	주행속도의 지시 변화
60km/h에서 6km/h 이내	100km에 대하여 1km 이내	24시간에 대하여 2분 이내	6km/h 이하

5.4 노이즈(Noise)

본 연구에서 지능형자동차의 디지털 운행기록 장치의 속도 측정은 약 40km/h에 상당하는 속력을 준 상태에서 출력 임피던스 50Ω의 펄스 값을 발생시켜 해당하는 조건값 충격성 잡음을 운행 기록계의 전원 단자와 접지 단자(또는 몸체 케이스) 사이에 접촉시켰을 때 디지털-운행기 기록장치가 정상적으로 작동 하는지의 기술적인 안전방식을 측정할 수 있도록 기준점을 설정 하였다. <표 7>에 디지털 운행기록 장치의 노이즈 시험조건을 나타낸 것이다.

<표 7> 노이즈 시험 조건

항 목	조 건
펄스의 높이	350V
펄스의 폭	1s
펄스의 상승 시간	1ns
펄스의 주파수	상용 주파수와 동일
펄스의 극성	+ 및 -
펄스의 위상	0°에서 360°까지

5.5 내진성 / 내구성

본 연구의 내진성 실험에서는 지능형자동차의 디지털운행 기록장치를 정규 부착 상태로 진동 시험대 위에 부착하고, 최고 눈금의 80% 또는 100km/h에 상당하는 속도를 주고, 진동수 33Hz, 전체 진폭 2mm의 진동을 상·하 방향에 대하여 4시간, 전·후 방향에 대하여 2시간, 좌·우 방향에 대하여 2시간 실험하고, 실험 중 또는 시험 후 장치의 기능과 성능이 정상적으로 작동하는를 확인하고 적용하였다. <표 7>은 디

지털은행 기록장치의 내진성 및 내구성을 나타낸 것이다.

VI. 결 론

본 논문은 지능형자동차의 운행기록 정보를 오픈 플랫폼을 이용하여 스마트디바이스 온톨로지로 표현하고, 단말장치에서 나타나는 다양한 디바이스 정보를 스마트모바일 환경에서 안드로이드 미들웨어를 통해 SoC Chip Set과 연동하여 운행기록장치의 주행 실험 값을 나타내고 확인하였다. 일반적인 안드로이드 스마트-폰에서 지능형자동차 SoC기반의 오픈플랫폼을 연구하는 것은 현재까지 개발 사례없고, 특히 연구개발에 있어서 전통적인 연구방법인 ASIC설계 기법을 벗어나 국제표준의 SoC 통합설계 방식을 적용함에 따라 통신속도가 4G통신망 속도보다 20배 빠른 5G 통신망을 테스트베드로 활용하여 실험을 하였다는 사실과 스마트디바이스 안드로이드 미들웨어를 설계하고 코딩하여 플랫폼의 오픈소스와 연계하여 차세대 인공지능자동차 오픈플랫폼 디바이스 설계의 모델로 다양하게 활용될 수 있을 것으로 판단하고 있으며, 실험 결과에 만족하고 있다.

본 연구에서 실험한 안드로이드 기반의 지능형자동차 SoC 미들웨어 설계 기술은 일반적인 자동차응 통신프로토콜과 개별적인 ECU Controller의 기능서비스를 벗어나 본 연구에서는 개별적인 ECU 시그널 신호처리를 SoC를 통해 안드로이드 기반의 오픈 플랫폼으로 적용하여 스마트디바이스와 연계한 운용기법을 통해 자율주행 지능형자동차 플랫폼에 적용하고 이를 통해 디지털신호처리 값을 자율주행실험을 통해 다양한 미디어 영상 값과 자율주행 정보의 실험 데이터 값을 확인 하였다.

향후의 연구목표는 현재, 연구중인 SoC 기반의 통합 단말시스템을 보다 상용화 수준으로 연구를 추진

하여 4차산업혁명의 핵심 산업인 자율주행 지능형자동차 산업발전과 기술 표준화 연구에 적극 활용하고, 세계최초 5G 이동통신망서비스 기술을 상용화한 국내통신 기술을 이용하여 국제경쟁력을 갖춘 인공지능 자율주행 기술의 최고 기술로 발전할 수 있도록 제품화 연구에 노력할 것이다. 마지막으로 본 논문을 통해 연구한 SoC 실험, 연구를 통해 확인된 결과내용은 국내 인공지능자동차 연구와 오픈플랫폼 연구개발에 활용된다면 디지털미디어를 활용한 지능형자동차 응용 발전에 많은 도움이 될 것으로 확신하며, 5G 이동통신 통신 인프라서비스 기술과 함께 국내의 자율주행 인공지능자동차 상용화 연구를 앞당기는 데 큰 도움이 될 것으로 기대한다.

VII. 약어정리

1. SoC: System On Chip
2. HCI: Human Computer Interaction
3. 3. SVC: Switched Virtual Connection
4. 4. P-to-P: Peer to Peer
5. 5. HUD: Heads Up Display
6. 6. HMD: Heads Mount Display
7. 7. ITS: Intelligent Transport System
8. 8. OBD: On Board Diagnostics
9. 9. ECU: Electronic Control Unit
10. 10. UX: User Experience
11. 11. EMS: Element Management System
12. 12: GUI: Grapic User Interface
13. 13. API: Application Program Interface

참고문헌

- [1] H-Y Cheng, B-S Heng, P-T Tseng, K-C Fan, "Lane Detection with Moving Vehicles in the Traffic Scenes," IEEE Trans. Intell. Transp. Syst. Vol.7, No.4, 2016, pp.571-582.
- [2] C-Y su, G-H Fan "An Effective and Fast Lane Detection Algorithm," Advances in Visual Computing, Vol.5359, 2015, pp.942-948.
- [3] Y. Zhou, R. Xu, X. Hu, Q. Ye, "A robust lane detection and tracking method based on computer vision," Measurement Science and Technology, Vol.17, 2014 pp.736-745.
- [4] 정도현, "미국의 지능형자동차 개발 및 실용화 지원 프로젝트(4)-IntelliDrive 프로젝트를 중심으로," 오토저널, 제31권 제4호, 2015, pp.66-71.
- [5] 양승한 · 송봉섭 · 엄재용, "레이더와 비전센서 융합을 통한 전방차량 인식 알고리즘 개발," 제어 로봇시스템 학회지, Vol.16, No.7, 2017, pp.639-645.
- [6] Y. Wang, E-K Teoh, D. Shen, "Lane detection and tracking using B-Snake," Image and Vision Computing, Vol.22, 2017, pp.269-280.
- [7] ZuWhan Kim, "Robust Lane Detection and Tracking in Challenging Scenarios," IEEE Trans. Intell. Vol.9, No.1, 2009, pp.16-26.
- [8] Radu Danescu and Sergiu Nedevschi, "Probabilistic Lane Tracking in Difficult Road Scenarios Using Stereovision," IEEE Trans. Intell. Transp. Syst. Vol.10, No.2, 2016, pp.272-282.
- [9] H. Yoo, U. Yang, K. Sohn, "Gradient Enhancing Conversion for Illumination-Robust Lane Detection," IEEE Transactions on Intelligent Transportation Systems, 14(3), 2013, pp.1083-1094.
- [10] Z. Sun, George Bebis, Ronal Miller, "On-Road Vehicle Detection: A Review," IEEE Trans. Pattern Anal. Mach. Intell., Vol.28, No.5, 2015, pp.694-711.
- [11] R. Labatrade, D. Aubert, and J. P. Tarel, "Real Time Obstacle Detection in Stereovision on Non Flat Road Geometry Through 'V-disparity' Representation," in Proc. IEEE Intelligent Vehicle Symposium, Vol.2, 2017, pp.646-651.
- [12] S. Boyer, C. Decamme, M. Hourdebaigt and O. Noel, "La Securite Des Pietons En 2009: etude Sectorielle," Observatoire Observatoire national interministeriel de securite routiere(France), ISBN 2-11-003686-9. 2016.
- [13] 최병관, "영상객체 sPFACS ASM 알고리즘을 적용한 얼굴인식에 관한 연구," 디지털산업정보학회 논문지, 제12권 제4호, 2016, pp.1-12.
- [14] 정성욱, "퍼지논리를 이용한 옷놀이 인공지능 플레어 연구," 디지털산업정보학회 논문지, 제13권 1호, 2017, pp.1~12.

■ 저자소개 ■



최 병 관
(Choi Byungkwan)

2019년 9월~현재
연세대학교/전임교원
과학기술대학 컴퓨터정보통신공학부

2019년 8월~중앙대학교/전임교원
ICT융합공학 인공지능학전공

2003년 12월 ㈜LG-CAP ICT 첨단기술연구소
기술중앙연구소장/상무

1993년 04월 ㈜한화/한화정보통신
기술중앙연구소/수석연구원

1981년 04월 한국과학기술원 시스템공학연구소

관심분야 : 인공지능, 로봇공학, 지능형자동차,
HCI, AR/VR, SoC 3D융합미디어,
E-mail : zcbk@yonsei.ac.kr

논문접수일	: 2019년 5월 7일
수정일	: 2019년 7월 23일
게재확정일	: 2019년 7월 30일