

**KOROSKOR
OROSKOR
ROSKOR
OSKOR
SKOR
KOR
OR**

KOROS

서비스 로봇을 위한 모듈 - 2-3 부 :
비전 기반 인식 지능 모듈을 위한
정보 모델

KOROS 1148-2 : 2021

한국로봇산업협회

2021년 05월 28일 제정

<http://www.KOROS.or.kr>

심 의 : 지능형로봇표준포럼 운영위원회

	성명	근무처	직위
(운영위원장)	박홍성	강원대학교	교수
(운영위원)	이순걸	경희대학교	교수
	우종운	한국로봇산업진흥원	센터장
	곽관웅	세종대학교	교수
	조영조	한국전자통신연구원	책임연구원
	김동한	경희대학교	교수
	정영숙	한국전자통신연구원	책임연구원
	전진우	한국로봇산업진흥원	실장
	임성수	경희대학교	교수
	문인혁	동의대학교	교수
	권용관	대림대학교	교수
	김규로	국가기술표준원	국장
	김석중	하이젠모터	이사
	김승훈	전자부품연구원	책임
	김혜지	국가기술표준원	주무관
	단병주	LG전자	수석연구원
	류영선	한국컴포짓	대표
	문전일	한국로봇산업진흥원	원장
	문승빈	세종대학교	교수
	박종환	넬판	대표
	서준호	한국로봇산업협회	팀장
	서태원	로보테크	부장
	심재홍	한국산업기술대학교	교수
	이병우	세화교정기술원	전문
	이용국	현대로보틱스	책임연구원
	임우철	두산로보틱스	책임
	장민	유진로봇	이사
	정병찬	한화정밀기계	수석연구원
	지수영	한국전자통신연구원	책임연구원
(간사)	김현근	한국로봇산업협회	주임

표준작성 기여자

	성명	근무처	직위
(과제제안자)	정영숙	한국전자통신연구원	책임연구원
(작성참여자)	송병열	한국전자통신연구원	책임연구원
	장철수	한국전자통신연구원	책임연구원
	박홍성	강원대학교	교수
	김미숙	강원대학교	책임연구원

목 차

1	적용범위	1
2	인용표준	1
3	용어와 정의	1
4	비전 기반 인식 지능 모듈을 위한 정보 모델 표현	2
4.1	개요	2
4.2	모듈 이름 정보	3
4.3	버전 정보	4
4.4	속성 정보	4
4.5	서비스 정보	7
4.6	인프라 정보	8
5	비전 기반 인식 지능 모듈을 위한 정보 모델의 클래스	8
5.1	개요	8
5.2	모듈 ID와 모듈 유형 클래스	10
5.3	속성 클래스	10
5.4	상태 및 건전성 조건을 위한 클래스	11
5.5	서비스를 위한 클래스	11
5.6	인프라를 위한 클래스	12

서 문

1 표준의 목적

이 표준은 서비스 로봇에 사용되는 비전 기반 인식 지능 모듈의 특성을 명시하는 정보 모델에 대해 정의하고, 모듈에서 제공하는 인식 지능 서비스 혹은 메소드 호출과 응답에 대한 내용을 포함한다. 이 표준에서 제시하는 서비스 로봇을 위한 비전 기반 인식 지능 모듈의 정보 모델은 모듈의 속성과 서비스 API들을 가진 클래스로 구성되며, 로봇 설계자 및 로봇/AI 응용개발자가 다양한 환경에서 개발한 비전 기반 인식 지능 모듈의 통합을 용이하게 한다.

본 표준에서 필수 항목은 “~하여야 한다,” 형태로, 권고사항은 “~하는 것이 좋다.” 혹은 “~하여야 할 것이다.” 형태로, 허용 항목은 “~해도 된다.” 형태로, 마지막으로 가능성 항목은 “~ 할 수 있다.” 형태로 문장을 서술한다. 이 외에 대등한 문장은 KS A 001 Annex H를 참조하여 작성한다.

2 타 표준과의 관련성

2.1

타 국내표준과의 관련성

KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항

KOROS 1148:2020, 서비스 로봇을 위한 모듈-2-1부-모듈을 위한 공통 정보 모델

2.2

타 국제표준과의 관련성

ISO DIS 22166-1, Modurlarity for service robots – part 1 — General requirements

3 지적재산권 인지 관련사항

해당사항 없음

4 적합인증 관련사항

해당사항 없음

5 표준의 이력

본 표준은 2021년 05월 28일에 개최된 제59차 지능형로봇표준포럼 운영위원회에서 제정되었다.

KOROS 1148-2 : 2021

서비스 로봇을 위한 모듈 - 2-3부 - 비전 기반 인식 지능 모듈을 위한 정보 모델

Modularity for service robots – Part 2-3 – Information Model for Vision based Recognition Intelligence Modules

1 적용범위

이 표준은 서비스 로봇에 사용되는 비전 기반 인식 지능 모듈의 특성을 명시하는 정보 모델을 정의하고 모듈에서 제공하는 인식 지능 서비스 혹은 메소드 호출과 응답에 관한 내용을 포함한다. 이 정보 모델은 고객인 로봇제조업체에 납품하는 모듈 제조업체, 모듈 통합자, 로봇 및 AI 응용 개발자가 다양한 환경에서 개발하는 비전 기반 인식 지능 모듈을 대상으로 한다.

2 인용표준

다음 문서는 전체 또는 일부가 이 문서에서 규범적으로 참조되며 해당 응용 프로그램에 반드시 필요하다. 날짜가 기입된 참고 문헌에 대해서는 인용된 판만 적용한다. 기한이 지난 참조의 경우 참조 문서의 최신 버전 (모든 개정 포함)을 적용한다.

KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항

KOROS 1148:2020, 서비스 로봇을 위한 모듈-2-1부-모듈을 위한 공통 정보 모델

3 용어와 정의

이 표준의 목적을 위하여 다음의 용어와 정의를 적용한다.

3.1 정보모델 (IM: Information model)

IM

동작환경, 속성 및 함수 그리고 서로 간의 관계를 가진 객체의 추상화와 표현

[출처: KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항]

3.2 공통 정보 모델 (CIM: Common Information model)

CIM

모든 모듈에 공통으로 적용되는 정보 모델

[출처: KOROS 1148:2020, 서비스 로봇을 위한 모듈-2-1부-모듈을 위한 공통 정보 모델]

3.3 모듈 (module)

시스템 디자인과 통합 그리고 상호운용성 및 재사용성을 위하여 속성 프로파일에 정의된 인터페이스

로 복합 또는 결합한 컴포넌트

[출처: KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항]

3.4 모듈 속성 (module property)

모듈의 특성

[출처: KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항에서 수정]

3.5 소프트웨어 모듈(software module)

순전히 프로그래밍된 알고리즘으로 구성된 모듈

[출처: KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항]

3.6 인식 지능 모듈(recognition intelligence module)

인간 로봇 상호 작용의 수단으로 환경을 인식하고 해석하는 기능을 구현한 소프트웨어 모듈

3.7 비전 기반 인식 지능 모듈 정보 모델(IMVR: Information Model for vision based Recognition intelligence modules)

IMVR

비전 기반으로 구현된 인식 지능 모듈에 적용되는 정보 모델

3.8 인프라(Infrastructure)

모듈 및 시스템의 운영을 지원하는 구조화된 리소스

[출처: KOROS 1149-1:2020, 서비스 로봇을 위한 모듈화 1부: 일반요구사항에서 수정]

4 비전 기반 인식 지능 모듈을 위한 정보 모델 표현

4.1 개요

비전 기반 인식 지능 모듈을 위한 정보 모델(IMVR)은 XML 형식으로 기술한다.

비전 기반 인식 지능 모듈을 위한 정보 모델은 모듈 이름(module name), 모듈 설명(description), 제조자(manufacturer), 모듈 식별자(module ID), 모듈 유형(module type), 예시(examples), 소프트웨어 특성(software aspects), 모듈 속성(module properties), 입력, 출력, 인프라(Infrastructure), 보안(security) 등 KOROS 1148에서 정의한 공통 정보 모델(CIM)의 모든 속성을 상속하고 인식 지능 모듈을 위한 추가 속성 정보와 데이터 유형 정보를 포함하여야한다.

표 4.1 비전 기반 인식 지능 모듈을 위한 정보 모델 예시

요소	내용(예)
모듈 이름	PCT_ObjRecg_PersonalBelongings
버전 번호	1.0
모듈 설명	(Optional)This is an implementation of Personal Belongings Detection and Recognition Module. The module has two main parts, object detection and instance classification.

	License: GNU GPLv3
제조사	제조사 정보(이름과 이메일)
모듈 ID	고유한 모듈 식별자
모듈 유형	BTS_RECG(Recognition Basic SW module)
예시	(Optional)모듈의 활용 예시
모듈 속성	<p>사용언어: python 2.7, python 3.6</p> <p>라이브러리: CUDA 8.0 or higher, OpenCV 3.2</p> <p>AI 프레임워크: Pytorch 1.4.0</p> <p>학습용 데이터셋: PASCAL_VOC 07+12, COCO, Visual Genome, Imagenet</p> <p>GPU: GeForce</p> <p>사용위치: local edge/cloud/robot platform</p> <p>입력 유형:</p> <ul style="list-style-type: none"> - Image(jpg, png, bmp, tiff) video stream(mp4, avi, mkv, asf, mpv, wmv) <p>출력 유형:</p> <ul style="list-style-type: none"> - object(class) name: "bottle", "cell phone", "cup", "remote" - object_location of top-left point(x1, y1) - object_width_height(width, height) - object_detection accuracy
인프라	- 데이터베이스의 위치(경로, IP, 포트번호, URL 등)

4.2 모듈 이름 정보

공통 정보 모델의 **GenInfo**는 모듈 이름, 모듈에 대한 설명, 제조자와 모듈의 사용 예시 등의 속성으로 구성된다. **GenInfo** 에서 모듈 이름은 기본 모듈 유형과 상세 모듈 유형으로 구성되어 '기본 모듈 유형_상세 모듈 유형'의 형식으로 기술한다. **KOROS 1148:2020**에서 상세 모듈 유형은 상세 규격에서 정의하도록 제시함에 따라 비전 기반 인식 지능 모듈을 위한 상세 모듈 유형은 인식 지능 유형과 사용자 지정 유형으로 구성하여 기술하여야 한다. 인식 지능 유형은 표4.2과 같이 분류하고 정의한다. 사용자 지정 유형은 사용자에게 의해 정의된다.

이에 따라 비전 기반 인식 지능 모듈의 모듈 이름은 '기본 모듈 유형_인식 지능 유형_사용자 지정 유형'으로 기술한다. 사용자 지정 모듈 명칭은 인식지능유형의 세부 유형으로 사물 인식 중 개인 소지품 인식을 예로 들 수 있다.

예) **RECG_ObjRecg_PersonalBelongings**: RECG는 기본 모듈 유형인 인지/인식 모듈(RECG), 인식 지능 유형인 사물인식 유형(**ObjRecg**), 사용자 지정 유형(**PersonalBelongings**)으로 구성된다.

표 4.2 비전 기반 인식 지능 모듈을 위한 상세 모듈 유형

인식 지능 유형	설명	심볼 또는 태그
사물 인식	다양한 사물 객체를 인식한다(사람도 객체가 될 수)	ObjRecg

(Object Recognition)	있음).	
사람 감지 (Human Detection)	사람만을 감지한다.	HumanDetect
장애물 감지 (Obstacle Detection)	장애물을 감지한다. 사물 인식과는 비슷할 수 있지만 다름.	ObstacleDetect
얼굴 인식 (Face Recognition)	사람의 얼굴을 식별하여 누구인지를 확인한다. 이 항은 예로 얼굴 검출과 얼굴 비교 등을 포함한다.	FaceRecg
제스처 인식 (Gesture Recognition)	사람의 제스처를 인식한다.	GestRecg
행동 인식 (Action Recognition)	사람의 행동을 인식한다.	ActRecg
객체 추적 (Object Tracking)	사물의 위치를 인식하고 추적한다.	ObjTracking
사람 추적 (Human Tracking)	사람만을 대상으로 위치를 인식하고 추적한다.	HumanTracking

4.3 버전 정보

비전 기반 인식 지능 모듈의 버전 정보를 **GenInfo**에 포함하여야 한다.

그림 4.1은 4.2에 의해 정의된 모듈 이름과 모듈에 대한 설명, 제조자의 이름과 이메일, 모듈 사용 예, 버전 정보 등을 적용한 **GenInfo** 정보의 예를 보여준다.

```
<?xml version="1.0" encoding="UTF-8"?>
<GenInfo>
  <ModuleName> RECG_ObjRecg_PersonalBelongings </ModuleName>    <!-- string -->
  <Description> description of module </Description>    <!-- string -->
  <Manufacturer email="e-mail address"> manufacturer name </Manufacturer>    <!-- string -->
  <Examples> list of use cases </Examples>    <!-- string -->
  <VersionNo> 1.0 </VersionNo>    <!-- string -->
</GenInfo>
```

그림 4.1 비전 기반 인식 지능 모듈의 **GenInfo** 정보 예시

4.4 속성 정보

비전 기반 인식 지능 모듈에 사용되는 속성(properties)을 표 4.3과 같이 정의한다. 모듈의 속성은 모듈에서 사용한 언어, AI 프레임워크, 라이브러리 정보와 OS, CPU, GPU 등이 하드웨어 환경, 모듈의 계산량을 고려한 사용 가능 위치, 입력과 출력 정보 유형과 학습에 사용된 데이터 셋 등의 정보를 포함한다. 그림 4.3은 비전 기반 인식지능 모듈의 속성 정보를 xml로 기술한 예를 보여준다.

표 4.3 비전 기반 인식 지능 모듈 속성

속성	내용(예)	심볼 또는 태그
----	-------	----------

사용언어	Python 2.7(3.6), Cython, C, C++, C#, Java, Scala, CUDA 등 list type으로 버전을 포함	Language
AI 프레임워크	Keras, Pytorch, TensorFlow, MxNet, Caffe, Caffe2, Weka 등 list type으로 버전을 포함	AIFramework
라이브러리	CUDA 8.0 or higher, OpenCV 3.2, Scikit-learn, 사용 언어별 기계학습 라이브러리 등 list type으로 버전 포함	ALLibrary
사용가능위치	robot , edge, cloud 중 택일	Location
OS 종류	OSType_##, 여기서 ##는 OS의 버전 번호임	OSType
CPU 종류	INTEL_16/32/64, ARM_16/32/64 등	CPUType
GPU 종류	GeForce, Quadro, GMA, Intel HD Graphics, Iris, Iris pro, Radeon Rx/Pro/Instinct 등	GPUType
입력 이미지 유형	모듈에 사용되는 입력 이미지에 관한 정보로 자세한 내용은 표 4.4. 참조	InputImageForm
출력 유형	모듈이 분류하는 클래스 종류("bottle", "cell phone", "cup", "remote"): 이 내용과 같이 enumeration type 으로 적음. 이 때 bottle, cell phone인 경우 각각 0, 1 값이 결과로 나옴 모듈에 사용되는 출력 정보에 관한 자세한 내용은 표 4.5 참조	OutputObjForm
학습용 데이터 셋	PASCAL_VOC 07+12, COCO, Visual Genome, Imagenet, 자체 제작 data set	TrainedDataset

표 4.4와 같이 비전 기반 인식 지능 모듈의 이미지 입력 유형을 정의한다. 지원하는 입력 이미지의 데이터포맷, 데이터 표현 방식, 이미지 데이터의 크기(넓이, 높이), 인식되는 객체의 최소 픽셀 크기 등의 정보를 제공하여야 한다.

표 4.4 태그 <InputImageForm>의 내용 예시

종류	설명	심볼 또는 태그
이미지 데이터의 포맷	이미지 데이터의 포맷으로 raw, bmp, jpeg/jpg, gif, png, tiff, psd, tga, ai, svg 등으로 지원하는 포맷을 나열한다. 2개 이상의 포맷을 지원하는 경우는 빈칸 혹은 콤마(,)로 분리하여 나열한다.	ImageFormat
이미지 데이터의 표현	Raw 이미지 데이터인 경우 이미지 표현이 컬러인지 흑백(bw)인지, 8비트 혹은 16비트인지를 표현하고, 컬러인 경우 bgr, bgra, rgb, rgba 인지를 구분한다.	ImageType

이미지 데이터의 크기	데이터의 크기는 (넓이, 높이)의 쌍으로 표현된다.	ImageSize
인식되는 객체의 최소 픽셀 크기	이미지 내에서 인식되는 객체, 사람, 얼굴의 최소 픽셀크기로 x,y의 순서쌍으로 표시함. 여기서 x는 x축의 최소 픽셀 크기이며, y는 y축의 최소 픽셀 크기이다.	MinRecgObjSize

지능 모듈의 출력 정보는 인식 모듈의 종류에 따라 다를 수 있다. 여기서는 일반적인 인식 모듈의 내용을 활용한다. 인식 관련 출력 정보는 인식된 객체에 대한 정보들이며, 인식된 객체는 하나 이상일 수 있다. 표 4.5와 같이 인식된 객체의 정보는 인식된 객체의 종류, 인식 정확도, 객체의 위치이다. 객체의 위치는 일반적으로 사각형으로 표시되는데 이는 다음의 데이터 구조로 표현할 수 있다: 왼쪽 모서리의 픽셀 위치(x,y), 높이(height)와 폭(width). 또한 인식된 객체의 종류를 정의하는 것은 Enumeration 형태로 정의하며, 해당 Enumeration의 집합은 정보로 제공한다.

또한 이러한 정보는 실제로 속성 정보로 제공되지만, 실제로 인식된 객체의 정보 출력으로 표시해야 한다. 따라서 인식된 객체의 정보 출력을 위하여 인식된 객체의 종류, 인식 정확도와 위치를 제공하여야 한다.

표 4.5 태그 <OutputObjForm>의 내용 예시

종류	설명	symbol 혹은 Tag
인식 가능한 객체의 종류 리스트	인식 가능한 객체의 종류를 표시하며, Enumeration 형으로 나열한다. 예로 {"bottle", "cell phone", "cup", "remote"} 이다. 이 내용의 결과가 인식된 결과 값으로 출력됨	RecgObjClass
인식정확도	해당 모듈의 정확도 평균을 제공한다.	Accuracy
객체의 위치	4개의 요소로 구성되며, 객체가 인식되어 출력 가능한 위치를 표시한다. 이 내용은 선택사항으로 언급이 없다면 입력 이미지의 전체를 표시한다.	x,y,width,height

```
<?xml version="1.0" encoding="UTF-8" ?>
```

```
<Properties>
```

```
<Property>
```

```
<OperatingEnv>      <!-- operating environments or conditions are provided  -->
```

```
<Language> Python 3.6 </Language>
```

```
<AIFramework> Pytorch 1.4 </AIFramework>
```

```
<AILibrary> CUDA 10.1, OpenCV 3.2, Numpy, </AILibrary>
```

```
<Location> edge </Location>
```

```
<OSType> WIN64_11 </OSType> <!-- 64bits Windows version 11 -->
```

```
<CPUType> INTEL_64 </CPUType>
```

```
<GPUPType> GeForce </GPUPType>
```

```
<TrainedDataset> COCO </TrainedDataset>
```

```
</OperatingEnv>
```

```

<InputImageForm>
  <ImageFormat> raw, bmp, jpg, gif, png, tiff </ImageFormat> <!--Image format -->
  <ImageType>
    <colortype> bgr </colortype> <!-- color type: bw, bgra, rgb, rgba -->
    <bitdepth> 8 | 16 </bitdepth> <!--bit depth of color channel -->
  </ImageType>
  <ImgeSize> (width, height) </ImageSize>
  <MinRecgObjSize> (x_min, y_min) </MinRecgObjSize>
</InputImageForm>

<OutputObjForm>
  <RecgObjs>
    <RecgObj>
      <RecgObjClass> {"bottle", "cell phone", "cup", "remote"} </RecgObjClass>
      <Accuracy> float32 </Accuracy>
      <x> int </x>
      <y> int </y>
      <width> int </width>
      <height> int </height>
    </RecgObj>
    ...
  </RecgObjs>
</OutputObjForm>
</Property>
</Properties>

```

그림 4.3 비전 기반 인식 지능 모듈의 속성 정보 xml 예

4.5 서비스 정보

Service 속성에 사용되는 기능(역량)을 위한 정보는 KOROS 1148:2020를 따르며 그림 4.4은 모듈의 서비스 정보 예를 보여준다. 정보는 C++, 자바, 파이선과 같은 프로그래밍 언어로 작성된 프로그램에서 사용된다. 그래서 전달 인자들의 순서를 결정하는 것과 인자들을 통하여 추가적 반환 값을 받는 것이 중요하다.

```

<?xml version="1.0" encoding="UTF-8" ?>
<Services>
  <Function>
    <name> detect </name> <!-- service/function name -->
    <Mandatory> TRUE </Mandatory> <!--Mandatory(=TRUE), OPTIONAL(=FALSE) -->
    <arguments>
      <1st>
        <Aname> imageFilePath </Aname> <!-- argument name -->
        <type> String </type> <!-- data type of argument -->
      </1st>
    </arguments>
    <ReturnValue type= "OutputFormatRecgObj" /> <!-- data type of return value 그림 5.8 참조 -->
    <Description>
      check the classified type of the input image, the recongnition accuracy, \
      and the location of the recognized objects
    </Description>
  </Function>
  <Function>
    <name> detect </name> <!-- service/function name -->

```

```

<Mandatory> FALSE </ Mandatory > <!--Mandatory(=TRUE), OPTIONAL(=FALSE) -->
<arguments>
  <1st>
    <Aname> imageData </Aname>      <!-- argument name -->
    <type> uint8[] </type>          <!-- data type of argument -->
  </1st>
</arguments>
<ReturnValue type= "OutputFormatRecgObj" /> <!-- data type of return value 그림 5.8 참조 -->
<Description>
  check the classified type of the input image, the recognition accuracy,\
  and the location of the recognized objects
</Description>
</Function>
</Services>

```

그림 4.4 비전 기반 인식 지능 모듈의 서비스 정보 예시

4.6 인프라 정보

인프라(Infra) 속성은 데이터베이스 관련 정보로 그림 4.5와 같이 제공되어야 한다. 이 내용에는 지능 모듈을 훈련시킨 데이터 파일들을 포함한다. 이러한 파일에는 당연히 인식된 객체의 레이블도 포함되어 있어야 하며, 얼굴인식의 경우에는 얼굴인식과 관련된 사람 이름도 포함되어 있다. 얼굴 인식 혹은 객체 인식을 위해 사용되는 데이터 집합에 대한 내용을 제공한다.

```

<?xml version="1.0" encoding="UTF-8" ?>
<Infra>
  <!-- 사용된 DB에 따라 다음 내용중 1개만 선택하여 사용 -->
  <DBpath> path_DB </DBpath> <!-- 로컬 컴퓨터에 있는 경우 -->
  <DBURL> URL_DB </DBURL> <!-- 원격에 있는 경우 -->
</Infra>

```

그림 4.5 비전 기반 인식 지능 모듈의 인프라 정보

5 비전 기반 인식 지능 모듈을 위한 정보 모델의 클래스

5.1 개요

비전 기반 인식지능 모듈을 위한 정보 모델은 KOROS 1148-2020에서 정의한 공통정보모델(CIM)의 속성을 상속받으며 또한 모듈의 특성에 따른 속성 정보를 추가하여 만든다. 그림 5.1은 비전 기반 인식지능모듈을 위한 정보 모델과 공통정보모델과의 관계를 보여준다.

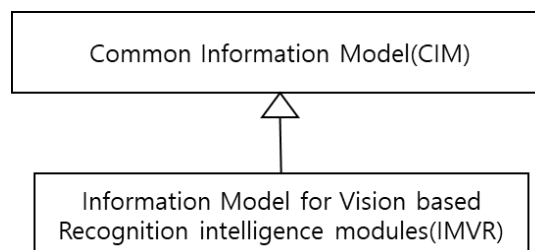


그림 5.1 CIM과 IMVR의 관계

공통정보모델은 IDnType, Properties, IOVariables, StatusHealth, Services, Infra, SafeSecure과 Modelling 의 이름을 가진 클래스들을 가지고 있으며, 그림 5.2와 같다.

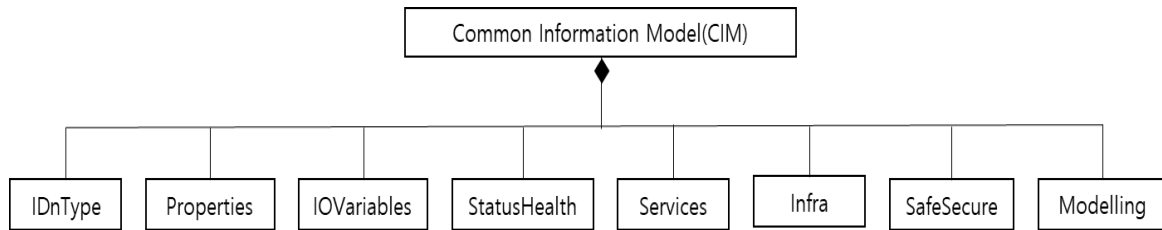


그림 5.2 공통 정보 모델 클래스

이미 4절에서 IMVR이 가지는 정보 속성을 제시하였다. 이 정보 속성들을 활용하면 IMVR에는 IOVariables와 modelling 관련 정보 속성이 없다. 따라서 이 2개의 클래스들은 IMVR에서는 사용하지 않는다. 따라서 IMVR 정보 모델용 클래스 다이어그램은 그림 5.3과 같이 만들 수 있다. 그림 5.3에서 oXXXXX 기호는 객체 변수를 의미한다.

또한 모듈의 사용 방법에 따라 StatusHealth 클래스를 사용할 수도 있고 사용하지 않을 수 있다. StatusHealth 클래스는 해당 모듈이 독립적으로 수행할 수 있을 경우에 사용되는 클래스다. 만일 모듈이 독립적으로 수행되지 않고 독립적인 수행모듈의 함수 혹은 기능으로 존재한다면 이 클래스는 사용하면 안된다.

주1 : 현재 버전은 1.0이다.

주2 : 속성은 다음 표식 중 하나를 사용한다: private (-), protected (*), public (+). 속성의 이름이 먼저 나오고, 속성의 데이터 타입을 그다음에 표시한다. 속성의 이름과 속성의 데이터 타입의 구분자는 ‘:’ 이다. 만일 속성이 public 멤버로 선언 된 경우 해당 속성에 접근하기 위한 함수를 정의할 필요가 없다.

Class IM
- ModuleName : String - Description : String - Manufacturer : String - Examples : String - InformationModelVersion : String + oIDnType : IDnType + oProperties : Properties + oIOVariables = NULL + oStatusHealth : StatusHealth + oServices : Services + oInfra : Infra + oSafeSecure : SafeSecure + oModelling = NULL
+ getModuleName() : String + putModuleName() : Boolean + getDescription() : String + putDescription() : Boolean + getManufacturer() : String + putManufacturer() : Boolean + getExamples() : String + put Examples() : Boolean

그림 5.3 IMVR 클래스 다이어그램

5.2 모듈 ID와 모듈 유형 클래스

IDnType 클래스는 모듈 ID와 모듈 유형 정보를 가지고 있다. 모듈 ID는 유일한 값을 가져야 하며, 모듈 유형은 복합(composite) 모듈형과 기본(basic) 모듈형으로 구분되지만 현재는 인식모듈은 기본 모듈형으로만 고려한다. 즉 여러 개의 지능 모듈로 구성되는 복합 모듈은 현재는 고려 대상이 아니다. HW 관련 내용이 존재하지 않기 때문에 HWAAspect 관련 내용은 존재하지 않는다. 그리고 기본 유형의 모듈만 고려하기 때문에 SWAspects 내용은 존재하지 않는다. 이 경우 그림 5.4와 같은 클래스 다이어그램을 가진다. 모듈 ID는 UUID를 생성하는 방법을 통하여 만든다.

Class IDnType
+ moduleID : String + oHWAsects = NULL + oSWAspects = NULL

그림 5.4 IdnType 클래스 다이어그램

5.3 속성 클래스

속성(Properties) 클래스는 그림 5.5와 같이 인식 모듈의 실행과 동작 환경 및 조건을 위해서 필요한 정보를 포함한 모듈의 속성을 제공해야 한다. 모듈의 실행에 필요한 정보는 모듈의 초기화를 위한 값들과 관련이 있다. 그림 5.5의 내용은 4.3절의 속성 정보로부터 추출되어 만들어져야 한다. 특히 InputImageForm과 OutputObjForm에 사용되는 클래스는 각각 그림 5.6과 그림 5.8에서 보여준다.

Class Properties
+ Language[]: String // list of Programming Language used in the module + AIFramework: String // AIFramework used in the module + AILibrary[]: String // list of AILibrary used in the module + Location: Enumeration // one of following values {ROBOT, EDGE, CLOUD} + OSType : String; // OS type used for the modules + CPUType : String // CPU type used for the module + GPUType : String // GPU type used for the module + InputImageForm : InputFormatImage // Input format of Image for Module, See Figure 5.6 + OutputObjForm: OutputFormatRecgObj //Output format of recognized object, See Figure 5.8 + TrainedData : String // trained data set used for training of the module

그림 5.5 Properties 클래스 다이어그램

Class InputFormatImage
+ ImageFormat : String // Image data format e.g. bmp, jpeg/jpg, gif, png, tiff, psd, tga, ai, svg + ImageType : Enumeration // one of following values {bw, bgr, bgra, rgb, rgba}, bw: black and white + ImageSize: Size // width by height of input image. See Figure 5.8 + MinRecgObjSize : Size // minimum width by minimum height of input image

그림 5.6 InputFormatImage 클래스 다이어그램

Class Size

+ Width : uint16 // width of Image + Height : uint16 // height of Image + MinRecgObjSize : Size // minimum width by minimum height of input image
--

그림 5.7 Size 클래스 다이어그램

Class OutputFormatRecgObj	
+ RecgObjClass: int16	// enum value of Object category e.g. {bottle, cellphone, cup, remote}
+ Accuracy : float32	// accuracy of object recognition
+ x : int16	// left top x pixel of recognized object
+ y : int16	// left top y pixel of recognized object
+ Width : int16	// width of rectangle of recognized object
+ Height: int16	// height of rectangle of recognized object

그림 5.8 OutputFormatRecgObj 클래스 다이어그램

5.4 상태 및 건전성 조건을 위한 클래스

모듈의 상태 및 건전성 조건을 위한 정보는 **StatusHealth** 클래스에 정의되어야 한다. 이 클래스는 모듈이 독립적으로 수행할 때 사용되는 클래스이다. 현재 고려대상인 인식 모듈은 다른 독립 수행 모듈이 호출하는 형태로 동작하도록 설계되므로 독립적 수행 모듈이 아니다. 따라서 그림 5.3의 클래스 다이어그램에서 해당 객체 내용은 **NULL**로 설정되어야 한다.

5.5 서비스를 위한 클래스

인식 지능 모듈의 서비스는 서비스 클래스 안에 정의해야 한다. 서비스 클래스는 모듈이 지원하는 인터페이스 함수를 **CIM**이 제공하는 기본 형태인 그림 5.9와 같이 제공하여야 한다. 인터페이스 함수는 4.6절의 서비스 속성에서 정의한 내용을 사용하여 정의해야 한다.

Class Services	
+ NoOfBasicService : int8	// Number of Basic services provided
+ NoOfOptionalService : int8	// Number of Optional services provided
+ BasicServDes[] : String	// Descriptions of Basic services provided
+ OptionalServDes[] : String	// Descriptions of Basic services provided
// basic(or Mandatory) Services	
+ serviceName_b1(ArgName1: dataType1, ..., ArgNameN: dataTypeN) : type of return data	
..	
+ serviceName_bM(ArgName1: dataType1, ..., ArgNameN: dataTypeN) : type of return data	
// optional(or Special) Services	
+ serviceName_s1(ArgName1: dataType1, ..., ArgNameN: dataTypeN) : type of return data	
..	
+ serviceName_sM(ArgName1: dataType1, ..., ArgNameN: dataTypeN) : type of return data	

그림 5.9 Services 클래스 다이어그램

4.6절에 정의된 내용을 사용하여 서비스를 위한 클래스를 만들면 그림 5.10와 같다.

+ NoOfBasicService =1 : int8	// Number of Basic services provided
+ NoOfOptionalService=1 : int8	// Number of Optional services provided


```

+ BasicServDes[] : String           // Descriptions of Basic services provided
+ OptionalServDes[] : String
// basic(or Mandatory) Services
+ detect(imageFilePath : String) : ClassRecog
// optional(or Special) Services
+ detect(imageData[] : uint8) : ClassRecog

```

그림 5.10 4.6절의 예제 기반 **Services** 클래스 다이어그램

5.6 인프라를 위한 클래스

인프라스트럭처 관련 정보에는 데이터베이스, 데이터버스, 전원 정보 등인데, 인식 지능 모듈 관련 내용에는 데이터베이스 관련 정보만 포함되어야 하며, 이 내용이 그림 5.11에 제시되어 있다.

Class Infrastructure
// types of DataBase mgt system + dbType : String // DBMS type used in module. e.g csv, oracle, sql, ... + DBlocation : Enumeration // location od data , choose one of following values : {Local, Remote} + DBpath : String // path of Data or URL of data

그림 5.11 비전 기반 인식 모듈용 **Infrastructure** 클래스 다이어그램

지능형로봇표준포럼

서비스 로봇을 위한 모듈 -2-3부 -
비전 기반 인식 지능 모듈을 위한 정보 모델
Modularity for service robots – Part 2-3 –
Information Model for Vision based Recognition Intelligence Modules

KOROS 1148-2 : 2021

제 정 자 : 지능형로봇표준포럼 의장

제 정 : 2021년 05월 28일

지능형로봇표준포럼 사무국
서울시 용산구 한강대로 31 금영빌딩 8 층한국로봇산업협회
전화 : (02) 780-3060