Ontology-Based Framework of Robot Context Modeling and Reasoning for Object Recognition

Wonil Hwang[1], Jinyoung Park[1], Hyowon Suh[1], Hyungwook Kim[2], and Il Hong Suh[2]

[1] Department of Industrial Engineering, KAIST, Daejeon, Korea
{onil, zknowledge, hw_suh}@kaist.ac.kr
[2] Graduate school of Information and Communications, Hanyang University, Seoul, Korea
{hwkim, ihsuh}@incorl.hanyang.ac.kr

Abstract. This paper introduces Multi-layered Context Ontology Framework (MLCOF) for comprehensive, integrated robot context modeling and reasoning for object recognition. MLCOF consists of six knowledge layers (KLayer) including rules such as an image layer, three geometry (1D, 2D and 3D geometry) layers, an object layer, and a space layer. For each KLayer, we use a 6-tuple ontology structure including concepts, relations, relational functions, concept hierarchies, relation hierarchies and axioms. The axioms specify the semantics of concepts and relational constraints between ontological elements at each KLayer. The rules are used to specify or infer the relationships between ontological elements at different KLayers. Thus, MLCOF enables to model integrated robot context information from a low level image to high level object and space semantics. With the integrated context knowledge, a robot can understand objects not only through unidirectional reasoning between two adjacent layers but also through bidirectional reasoning among several layers even with partial information.

1 Introduction

Intelligent service robot has to understand and carry out user's requirements. For this, object recognition is essential to robot. In the early researches of image processing, objects are recognized using pattern matching with low level image data extracted from sensors. However these approaches are limited in object recognition. To solve this problem, some researchers have been focusing on context based object recognition. They modeled context information such as objects and their geometric shape. But, the approaches do not have concrete reasoning mechanism. In addition, the context information is limited in integrating low level information. Moreover, the context information describing robot environment is very complex and not well organized. Thus systematic framework is needed to represent it.

This paper introduces Multi-layered Context Ontology Framework (MLCOF) for comprehensive integrated context modeling and reasoning for object recognition in a robot environment. This proposed approach is based on ontology. By constructing ontology for context information such as image, geometries, objects and spaces, the robot environment knowledge can be managed comprehensively and synthetically, and object reasoning with partial context information is possible.

L. Wang et al. (Eds.): FSKD 2006, LNAI 4223, pp. 596 – 606, 2006.

In this research, context information includes geometries, objects and spaces. In a robot environment, the use of multi-layered context information can enable the accuracy of object recognition. This proposed framework is represented with first-order logic to maintain integrated uniform representation.

In section 2, we discuss previous researches, and basic concept of MLCOF is provided in section 3. The details of MLCOF based object ontology model is discussed in section 4. Finally in section 5, a summary and consideration for future research are provided.

2 Related Works

There are several knowledge-based image retrieval systems related to object recognition such as a computer system for interpreting scenes [2], a knowledge based image understanding system [4], a knowledge based galaxy classification system [6]. These systems use data schemas for context information, which are limited in containing sufficient and structured context information. So, ontology based systems have been developed.

Ontology-based image processing: S Liu et al. [5] introduced a framework for building ontology to provide semantic interpretations for image contents. Their ontology is focused on visual information such as color, texture and edge features which are extracted from an image. They define general concepts and properties in the ontology. However their ontology is developed only for visual concepts. N Maillot et al. [3] proposed to use visual concept ontology to hide the low-level image layer. The mapping between objects and image is based on the visual concept ontology. These ontological concepts are linked both to object knowledge and to low-level vision numerical descriptors. This ontology can be considered as a guide that provides the vocabulary for the visual description of objects. J Vompras [7] also used ontology to reduce the gap between low-level visual features of images and high-level human perception of inferred semantic contents. The ontology framework contains image primitives as well as semantic information. However, the above studies are limited in using full ontology, and their reasoning algorithms are not well organized.

Ontology-based Context Modeling: Go and Sohn [9] studied how to model context information in a robot environment. They developed the context model using rules and ontology. Rules are used for modeling dynamic information such as current user's location, current robot dictator and so on. Ontology is mainly used for describing static information about most parts of device, space, person and artifact. Their research focuses on modeling object, space and activities. They do not concern low-level context which is generally used to recognize their context of interest in practical application. CONON [8] proposes OWL encoded context ontology for modeling context and supporting logic based context reasoning. The context ontology includes location, person, activity and composite entity. Furthermore they developed context reasoning process. Using this reasoning mechanism, implicit context (activities such as sleeping, cooking) can be deduced from explicit context (space, person, device and so on). However their approach does not provide sensor-driven image or geometry information. They only modeled objects, spaces and activities, and defined rules for reasoning between them.

3 Basic Concept of MLCOF

MLCOF consists of six knowledge layers (KLayer): image, 1D geometry, 2D geometry, 3D geometry, object and space. For managing integrity and consistency of this framework, geometry, object and space layer have meta-ontology, ontology and ontology-instance layer while image layer has ontology and ontology instance layer. We use a 6-tuple ontological structure including concepts, relations, concept hierarchies, relation hierarchies, relation functions and axioms. The rules are used to specify or infer the relationships between ontological elements at different KLayers. In Appendix, for the formality of MLCOF, seven ontological definitions are defined based on KAON ontology [1].

MLCOF enables us to model integrated context information from a low level image to high level object and space semantics. The uses of various forms of context information such as geometry, object, and space enable more accurate object recognition. Thus, if we use multi-layered approach containing several forms of context information, systematic information management and effective reasoning process are possible. Figure 1 shows overall architecture of MLCOF.

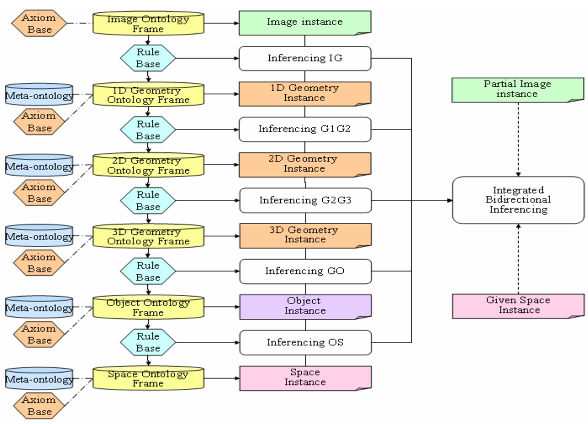

Fig. 1. Architecture of MLCOF (Dynamic View)

4 MLCOF-Based Context Ontology Modeling

MLCOF provides a framework for integrated and systematic context ontology modeling and reasoning. We can model all the context information related to visual objects and infer an object with an MLCOF based context ontology. For MLCOF-based context ontology modeling and reasoning, we need to define usage mechanism of MLCOF. First, we need to describe role and usage of each layer. Second, building rules and axioms has to be discussed. Finally, we need to describe not only unidirectional reasoning between two adjacent layers, but also bidirectional reasoning among several layers with partial information.

4.1 Descriptions of KLayers

We divide KLayer into a meta-ontology layer, an ontology layer and an ontology instance layer. Meta ontology layers capture general features of context entities. They support sharable and reusable framework to represent the entities such as geometries and objects. We construct meta-ontology layers using the 6-tuple ontology structure. Ontology layers are for describing domain specific entities like a refrigerator in the object layer. When we construct an ontology layer, we use a meta-ontology layer. Ontology instance layers are the instances of ontology layers, which are recognized results by reasoning process based on MLCOF from an input image.

Image Layer: it represents information of an image including pixel information as shown in figure 2. In addition, an image layer may include low-level features extracted from an image such as edge features, SIFT feature points and etc. Thus, an image layer needs to have additional entities to represent them. In the example in figure 2 (a), an image layer has SIFT feature points as well as its pixel information. A set of SIFT feature points in an image layer is directly matched to an object in an object layer. In this case, a geometry layer is not necessary. However, a set of pixels is recognized as a set of edges, and the edges are then recognized as lines or curves in geometry layer by edge detection rules and geometry reasoning rules. Thus, geometry layer are generally necessary.

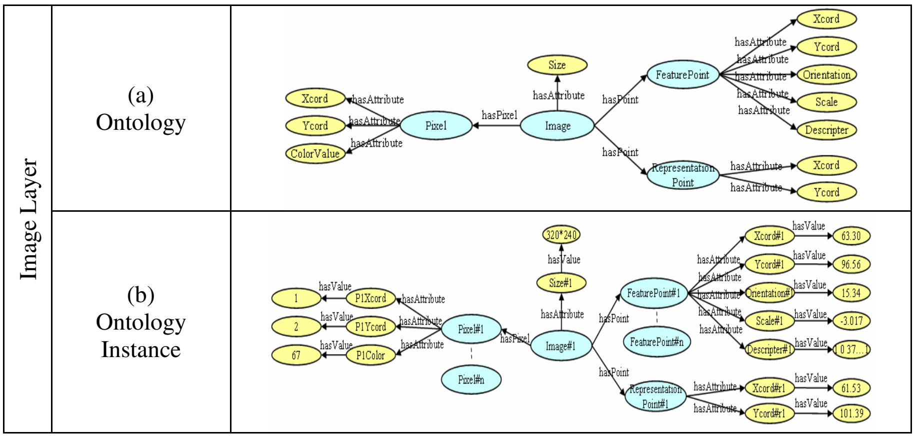

Fig. 2. Example of image layer

Geometry Layer: it represents geometries obtained from a set of pixels in an image layer. Geometry meta-ontology layer describes representative geometry concepts, attributes, and their relations as shown in figure 3 (a). Geometry concept is divided into 1-dimensional geometry, 2-dimensional geometry, 3-dimensional geometry, and spatial relations between the geometries are defined. Geometry concepts also have attributes and geometry reference image to represent a related image in the image layer. Geometry ontology layer represents specific geometry such as 1D geometry-line, 2D geometry-rectangle, 3D geometry-hexahedron, etc. Thus, we classify geometry ontology layer into three KLayers such as 1D, 2D and 3D geometry ontology layer as shown in figure 3 (b). This classification supports flexible definitions of geometries. Geometry instance layer represents an instance of a geometry ontology layer. Figure 3 (c) shows an example of geometry instances for a quadrangle example. This instance can be obtained from an image instance by edge detection and other rules, which will be discussed later.

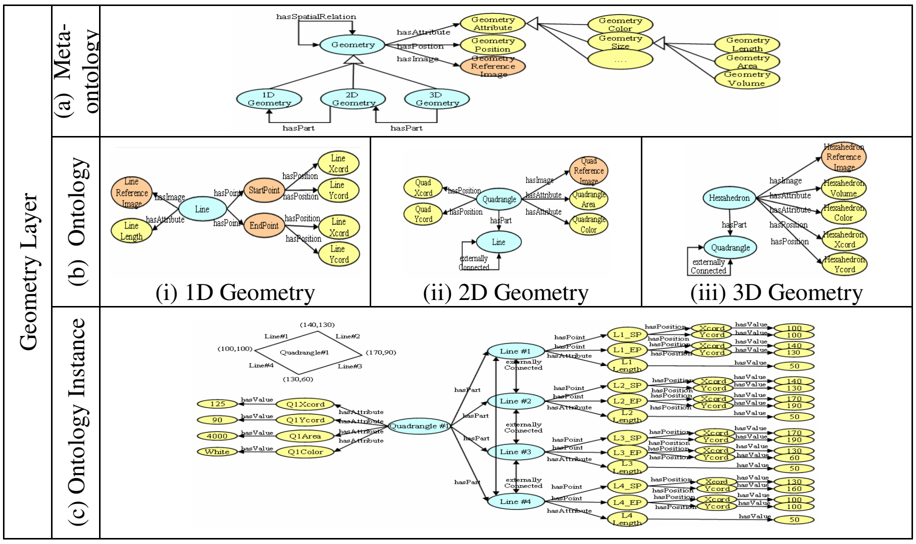

Fig. 3. Example of geometry layer

Object Layer: it represents object and its properties. Objects are divided into top objects and child objects. Top object is the target object which we want to recognize, and it usually consists of several child objects. For example, we regard a refrigerator as a top object while we consider its door and knob as child objects. Child object is a functionally significant subset of top object or another child object. Object meta-ontology layer defines object, object position, object reference image, object function and object attribute as concepts. In object ontology layer, we represent specific real world object such as refrigerator, chair, etc. All concepts and relations in this layer are defined according to the object meta-ontology layer, and an object ontology instance layer includes specific instance of a refrigerator obtained from input geometry instances. Figure 4 shows an example of an object layer.

Space Layer: it represents space and its properties. We deemed space as a set of functionally related objects. So space has several objects in it. Space Layer consists of meta-ontology, ontology and ontology instance layer, and they are similar to those in geometry and object layer. Figure 5 shows an example of a space layer.

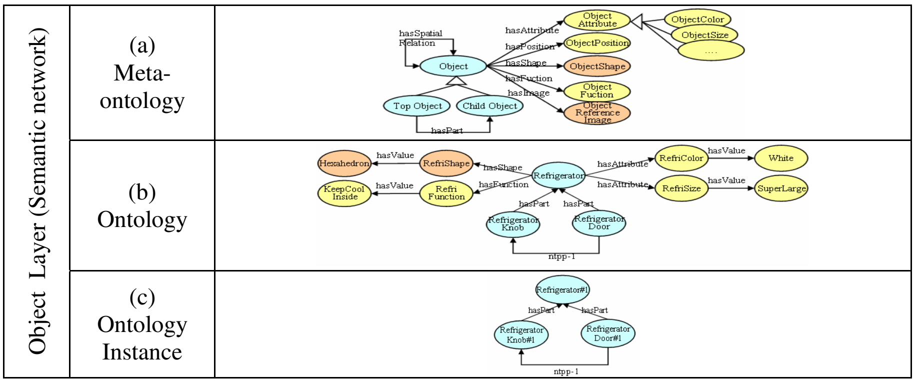

Fig. 4. Example of object layer

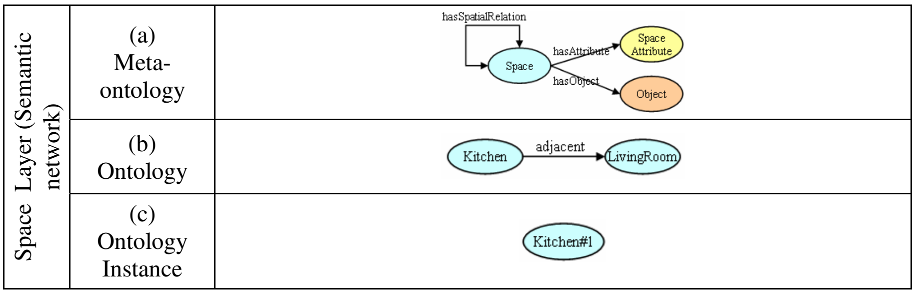

Fig. 5. Example of space layer

4.2 Axioms for Each KLayer

According to definition 4 and 5, we can define axioms. The axioms are categorized into two types. One is to explicitly specify the semantics of concepts and relations so that the ambiguity of context knowledge can be alleviated. The other is the relational constraints between the concepts and the relations. Relational constrains are for both spatial and hierarchical constraints. Table 1 shows the examples of axioms. The axioms are generally represented with logical language so that they can be a basis for additional reasoning.

4.3 Rules Between KLayers

According to definition 6 and 7, we can define rules. While axioms are defined in each KLayer, rules are defined between KLayers. The rules are used for reasoning the information which is necessary for recognizing instances in one layer from those in another layer. These rules are generally represented in the form of IF-THEN. And we semantically classify rules into two categories such as common rules and integrated rules which are discussed in the followings.

Unidirectional reasoning can be achieved by rules which are defined between two adjacent layers. Using these rules, image, geometry and object instances are inferred orderly. Table 3 describes rough examples which show how the rules are used to reason geometries from an image and object from those geometries. First, we derive lines from an image by an edge detection rule. Quadrangles are then inferred from the lines by a 'Rule_G1G2'. After that, we can reason a hexahedron based on quadrangles by a 'Rule_G2G3'. Lastly, we can recognize a refrigerator with the hexahedron and some additional information. In real situation, rules are defined more complicatedly. In these examples, we assume that a refrigerator has three types as shown in table 3.

Bidirectional reasoning is represented by integrated rules which are defined among several layers. Integrated rules are used to infer objects or some necessary information for object recognition when related information is given. A simple example of integrated rules in table 2 presents how we can recognize an unknown object as a refrigerator (or a computer) with object candidates and space information such as the kitchen (or the office).

Table 1. Examples of axioms

Layer	Type	Meaning	[Identifier] / FOL representation
I.L[1]	S[4]	Range of row value.	[AIS_Row] $\forall$x, c Xcord(x) $\wedge$ hasValue(x, c) $\wedge$ ($-2^{63} \le c \le +2^{64}$)
	S	Range of column value.	[AIS_Col] $\forall$y, c Xcord(y) $\wedge$ hasValue(y, c) $\wedge$ ($-2^{63} \le c \le +2^{64}$)
G.L[2]	S	Commutativity of 'Externally connected' relation	[AGS_EC] $\forall$x,y Geometry(x) $\wedge$ Geometry(y) $\wedge$ externallyConnected(x,y) $\Rightarrow$ externallyConnected(y,x)
	S	Inverse relation of 'TPP' and 'TPP-1'	[AGS_NTTP] $\forall$x,y Geometry(x) $\wedge$ Geometry(y) $\wedge$ TPP (x,y) $\Rightarrow$ TPP-1(y,x)
O. L[3]	S	Transitivity of 'hasPart' relation	[AOS_Transitivity] $\forall$x,y,z Object(x) $\wedge$ Object(y) $\wedge$ Object(z) $\wedge$ hasPart(x,y) $\wedge$ hasPart(y,z) $\Rightarrow$ hasPart(x,z)
	R[5]	Spatial relation between object	[AOR_DoorKnob] $\forall$x,y Knob(x) $\wedge$ Door(y) $\wedge$ hasPart(y,x) $\wedge$ $\neg$ inCenterOf(x, y)

Abbreviated terms:
(1: Image layer, 2 Geometry layer, 3: Object layer,
4: Specification of the semantics for concepts and relation, 5: Relational constraints)

Table 2. Example of rules for recognizing refrigerator (bidirectional reasoning with partial space information)

Rule Type	Meaning	[Identifier] / FOL representation
Integrated Rule	IF X's candidates are refrigerator and cabi net AND X is in space: kitchen THEN X is refrigerator	[IR_ref] $\forall$x,h,k (refrigerator(x) $\vee$ cabinet(x)) $\wedge$ kitchen(h) $\wedge$ hasObject(k,x) $\Rightarrow$ Refrigerator(x)
Integrated Rule	IF X's geometry is hexahedron AND X is in office AND There is keyboard and mouse ne ar X THEN X is computer	[IR_com] $\forall$x,h,k,m,o Object(x) $\wedge$ Hexahedron(h) $\wedge$ Keyboar d(k) $\wedge$ Mouse(m) $\wedge$ Office(o) $\wedge$ hasGeometry(x,h) $\wedge$ adjacent(x,k) $\wedge$ adjacent(x,m) $\wedge$ hasObject(o,x) $\Rightarrow$ Computer (x)

Table 3. Example of rules for recognizing refrigerator (unidirectional reasoning from image layer to object layer)

Rule Type	Meaning	[Identifier] / FOL representation	
Rule_IG	IF S is a set of edge pixel. AND orientations of all the elements in S are almost same. AND Y refers S THEN Y is line	[RIG_Line] S1: ∀p,x,y,r pixel(p)∧xcord(x)∧ycord(y)∧hasXcord(p,x)∧hasYcord(p,y) ∧R(r)∧hasR(p,r)⇒(fv(x,y)=multiply(differential(Gaussian(y)),Gaussian(x))∧(fh(x,y)=multiply(differential(Gaussian(x)),Gaussian(y))∧(Rv(x,y) =convolution(I(x,y),fv(x,y)))∧(Rh(x,y)=convolution(I(x,y),fg(x,y))) (R(x,y)=square(Rv(x,y))+square(Rh(x,y)))∧(r = R(x,y)) S2: ∀p,r,t pixel(p) ∧ R(r) ∧ hasR(p,r) ∧(r>T) ⇒ edgepixel(p) S3: ∀s,e2 ∃ e1, lineSet(s) ∧ edgepixel(e1) ∧ (e1∈s) ∧ edgepixel(e2) ∧neighbor(e1,e2)∧direction(d1)∧direction(d2)∧hasDirection(s,d1)∧hasDirection(e2,d2) ∧(abs(d1-d2) < D) ⇒ (e2 ∈ s) S4: ∀s,e1,e2,x1,y1,x2,y2,a,p1,p2 lineSet(s)∧refer(a,s)∧hasStart(s,e1)∧ hasEnd(s,e2)∧hasXcord(e1,x1)∧hasYcord(e1,y1)∧hasXcord(e2,x2)∧hasYcord(e2,y2) ⇒ line(a) ∧ hasStartPoint(p1) ∧ hasEndPoint(p2) ∧ hasXcord(p1,x1) ∧hasYcord(p1,y1)∧hasXcord(p2,x2)∧hasYcord(p2,y2) haslength(a, squreroot(square(x1-x2)+square(y1-y2))	
Rule_ G1G2	IF Y is consisted of 4 lines. AND Each line's end point is connected to other line's start point. AND All lines don't intersect each other. THEN Y is quadrangle.	[RGG_Quad] ∀a1,a2,a3,a4,sp1,ep1,sp2,ep2,sp3,ep3,sp4,ep4,y Line(a1) ∧ Line(a2) ∧ Line(a3) ∧ Line(a4) ∧ StartPoint(sp1) ∧ EndPoint(ep1) ∧ StartPoint(sp2) ∧ EndPoint(ep2) ∧ StartPoint(sp3) ∧ EndPoint(ep3) ∧ StartPoint(sp4) ∧ EndPoint(ep4) ∧ hasPoint(a1,sp1) ∧ hasPoint(a1,ep1) ∧ hasPoint(a2,sp1) ∧ hasPoint(a2,ep2) ∧ hasPoint(a3,sp3) ∧ hasPoint (a3,ep3) ∧ hasPoint(a4,sp4) ∧ hasPoint(a4,ep4) ∧ samePoint(sp1,ep4) ∧samePoint(sp2,ep1) ∧samePoint(sp3,ep2) ∧samePoint(sp4,ep3) ∧ hasPart(y,a1) ∧ hasPart(y,a2) ∧ hasPart(y,a3) ∧ hasPart(y,a4) ∧ ¬ intersect(a1,a2) ∧ ¬ intersect(a2,a3) ∧ ¬ intersect(a3,a4) ∧ ¬ intersect(a1,a4) ∧ ¬ intersect(a1,a3) ⇒ Quadrangle(y)	
Rule_ G2G3	IF Y is consisted of 3 quadrangles. AND Each quadrangle shares two lines with other two quadrangles. AND The number of shared line is two. THEN Y is Hexahedron	[RGG_Hexahedron] ∀a1,b1,a2,b2,a3,b3,a4,b4,r1,r2,r3,r4,h Line(a1)∧Line(b1)∧Line(a2)∧Line(b2)∧Line(a3)∧Line(b3)∧Line(a4)∧Line(b4)∧Quadrangle(r1)∧Quadrangle(r2)∧Quadragle(r3)∧Quadrangle(r4)∧hasPart(r1,a1)∧hasPart(r1,b1)∧hasPart(r2,a2)∧hasPart(r2,b2)∧hasPart (r3,a3) ∧hasPart(r3,b3) ∧hasPart(r4,a4) ∧hasPart(r4,b4) ∧hasPart(h,r1)∧ hasPart(h,r2)∧hasPart(h,r3)∧hasPart(h,r4)∧sameLine(a1,a2)∧sameLine (b2,b3)∧sameLine(b1,a3)∧differentLine(a1,b1)∧differentLine(a2,b2)∧differentLine(a3,b3) => Hexahedron(h)	
Rule_ GO	IF X 'shape is hexahedron. AND X's vertical side length is larger than its horizontal size length. AND X' color is white. AND it has one door. THEN X is refrigerator	[RGO_refrigerator1] ∀r,d,c,h,v,w Door(d) ∧ Hexahedron(h) ∧ ObjectColor(c) ∧ hasPart(r,d) ∧ hasValue(c,White) ∧ hasAttribute(r,c) ∧ hasShape(r,h) ∧ hasHeight(r,v) ∧ hasWidth(r,w) ∧ (w>v) ⇒ Refrigerator(r) ∧ hasType(r, T1)	Type1 
Rule_ GO	IF X has two doors which are placed above and below. AND Other conditions equal to 'Refrigerator (Type1)' THEN X is also refrigerator	[RGO_refrigerator2] ∀r,d1,d2,h,c,v,w Door(d1) ∧ Door(d2) ∧ Hexahedron (h) ∧ ObjectColor(c) ∧ hasPart(r, d1) ∧ hasPart(r, d1) ∧ ec(d1,d2) ∧ verticalWith(d1,d2) ∧ hasValue(c,White) ∧ hasAttribute(r,c) ∧ hasShape(r,h) ∧ hasHeight(r,v) ∧ hasWidth(r,w) ∧ (w>v) ⇒ Refrigerator(r) ∧ hasType(r, T2)	Type2 
Rule_ GO	IF X has two doors which are placed which are placed left and right. AND Other conditions equal to 'Refrigerator (Type1)' THEN X is also refrigerator	[RGO_refrigerator3] ∀r,d1,d2,h,c,v,w Door(d1)∧Door(d2) ∧ Hexahedron(h) ∧ ObjectColor(c) ∧ hasPart(r, d) ∧ ec(d1,d2) ∧ horizontalWith(d1,d2) ∧ hasValue(c, White) ∧ has Attribute (r, c) ∧ hasShape(r,h) ∧ hasHeight(r,v) ∧ hasWidth(r, w) ∧ (w>v)⇒ Refrigerator(r) ∧ hasType(r, T3)	Type3 

5 Conclusions and Further Study

This paper introduces Multi-layered Context Ontology Framework (MLCOF) for comprehensive, integrated object modeling and reasoning in a robot environment. MLCOF consists of five knowledge layers (KLayer) including axioms and rules. Five KLayers with axioms and rules enable to model integrated context information from sensor driven low level image to high level object semantics. With the integrated context information, a robot can understand objects through not only unidirectional reasoning between two adjacent layers but also bidirectional reasoning among several layers even with partial information. This research makes integrated robot-environment-recognition possible. However, further researches are necessary for practical application. The rich context vocabulary, axioms and rules need to be defined. In addition, going ahead of current object recognition level, the recognition of space and dynamic situation should be realized. An XML-based representation of the model is also necessary for sharing the environment knowledge with other agents.

Acknowledgement

This work is supported by the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Korea Ministry of Commerce, Industry and Energy.

References

1. Bozsak, E., Ehrig M., Handschuh S., Hotho A., Maedce A., Motik B., Oberle D., Schmitz C., Staab S. and Stojanovic L., (2002), “KAON – Towards a Large Scale Semantic Web,” Lecture notes in computer science, No. 2455, pp. 304-313.
2. Hanson**,** AR**, Riseman EM,** (1978), “VISIONS: a computer system for interpreting scenes”, Computer vision systems. Academic, New York**, pp. 303-333**.
3. Maillot, N, Thonnat M, Boucher A, (2004), “Towards ontology-based cognitive vision”, Machine Vision and Applications, pp. 33-40.
4. Matsuyama**,** T, Hwang V, (1990), “SIGMA – a knowledge-based aerial image understanding system”, Plenum, New York**.**
5. Song, L, Liang-Tien C, Syin C, (2004), “Ontology for Nature-Scene Image Retrieval”, CoopIS/ DOA/ ODBASE 2004, LNCS 3291, pp1050-1061.
6. Thonnat, M, Bijaoui, (1989), “A Knowledge-based galaxy classification systems. Knowledge-based systems in astronomy.”, Lecture Notes Phys , pp. 121-159.
7. Vompras, J., (2005), “Towards Adaptive Ontology-Based Image Retrieval”, 17th GI-Workshop on the Foundations of Databases, pp148-152.
8. Wang, X.H. , Zhang, D.Q. , Gu, T. and Pung, H.K., (2004), “Ontology based context modeling and reasoning using OWL”, Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on, pp.18-22.
9. Young Cheol Go, Joo-Chan Sohn, (2005), “Context Modeling for Intelligent Robot Services using Rule and Ontology”, The 7th international conference on, Vol. 2, pp. 813-816.

Appendix: Formal Definition of MLCOF

Definition 1
Multi-layered Context Ontology Framework

$MLCOF := (KLayer, R^0)$

Such that KLayers are knowledge layers and R^0 is a finite set of rules.

Definition 2
A set of Knowledge Layers of MLCOF consists of 17 Layers;

$KLayers := \{KLayer_{ij} \mid 1 \le i \le 6, 1 \le j \le 3\} \backslash \{KLayer_{11}\}$

We define a knowledge layer $KLayer_{ij}$ for $i,j \in N$ (set of natural numbers), $1 \le i \le 6$, $1 \le j \le 3$.

$KLayer_{1j}$ is a knowledge layer for the image layer,
$KLayer_{2j}$ is a knowledge layer for the 1D geometry layer,
$KLayer_{3j}$ is a knowledge layer for the 2D geometry layer,
$KLayer_{4j}$ is a knowledge layer for the 3D geometry layer,
$KLayer_{5j}$ is a knowledge layer for the object layer,
$KLayer_{6j}$ is a knowledge layer for the space layer,
$KLayer_{i1}$ is a knowledge layer for the meta-ontology layer,
$KLayer_{i2}$ is a knowledge layer for the ontology layer,
$KLayer_{i3}$ is a knowledge layer for the ontology instance layer.

Specially,
$KLayer_{31}$ and $KLayer_{41}$ are equal to $KLayer_{21}$.

Definition 3
A ij-th knowledge layer in MLCOF consists of 6-tuples;

$KLayer_{ij} := (C_{ij}, R_{ij}, Rel_{ij}, H_{ij}^C, H_{ij}^R, A_{ij}^0)$

Such that
C_{ij} is a set of concepts in $KLayer_{ij}$
R_{ij} is a set of relations in $KLayer_{ij}$
Rel_{ij} is a set of relation functions in $KLayer_{ij}$
H_{ij}^C is a set of concept hierarchies in $KLayer_{ij}$
H_{ij}^R is a set of relation hierarchies in $KLayer_{ij}$
A_{ij}^0 is a set of axioms

Definition 4
The set of axioms of each KLayer is a set of sentences Λwhich follows the representation of logical language. $\Lambda$is represent d by 3-tuples of KLayer (elements of C_{ij}, R_{ij} and Rel_{ij} of $KLayer_{ij}$), which are the elements in the same KLayer. Also, a sentence of Λ specifies the meaning of the elements by describing the relationship of the elements in a KLayer. Any sentence in Λ can not be entailed by other sentences in Λ.

Definition 5
Axioms are based on a structure of 3-tuples;

$A^0 = \{AI, \Lambda, \alpha\}$

(i) AI is a set of axiom identifiers
(ii) Λ is a set of logical sentences, and
(iii) α is a set of axiom mapping functions: α: AI $\Rightarrow \Lambda$

Definition 6
The set of rules is a set of sentences $\mathcal{D}$ which follows the representation of logical language. $\mathcal{D}$ is represented by 3-tuples of KLayer (elements of C_{ij}, R_{ij} and Rel_{ij} of $KLayer_{ij}$). A sentece of $\mathcal{D}$ represents the relationship between the three elements of KLayers (C_{ij}, R_{ij} and Rel_{ij}), and is used to entail other concept or relation. The rule should include at least two elements, one from a KLayer and the other from another KLayer.

Definition 7
Rules are a structure of 3 tuples:

$$R^0 = \{RI, \mathcal{D}, \beta\}$$

(i) RI is a set of rule identifiers
(ii) $\mathcal{D}$ is a set of logical sentences, and
(iii) β is a set of rule mapping functions: β: RI $\Rightarrow \mathcal{D}$