INTERNATIONAL DESIGN CONFERENCE - DESIGN 2006
Dubrovnik - Croatia, May 15 - 18, 2006.

DESIGN 2006

TRANSLATION BETWEEN LINGUISTIC STRUCTURES AND SHAPE STRUCTURES FOR BIDIRECTIONAL DESIGN

S. Miura, M. Utiyama, K. Takeuchi, H. Isahara, Y. Nagai and T. Taura

Keywords: machine translation, word translation, feature, CSG

1. Introduction

In upstream design, we generate a blueprint as a preparatory step in deciding on the concept of shape [Pahl and Beitz, 1988]. This process is important in forming the whole design object. When we construct a shape image, a negotiation occurs between the linguistic world and the shape world. Tomes et al. concluded, according to the results of an interview with an experienced designer, that such images involve progressive negotiation at verbal and nonverbal levels in the designer's mind [Tomes et al., 1998]. As we illustrate in Figure 1, our image generation process is based on such mutual negotiation.

We hope to support interactive design based on language and shapes, as shown in Figure 2. Here we evaluate the interaction as a kind of interface between a human and the design system. Thus we distinguish between interaction as an interface and negotiation as a phenomenon of the conceptual generation process. Such interactions are required for non-empirical design, that is, design free from geometric description and drawing for example, because the interactive system takes over the manual tasks in those interactions. An interactive design method that handles only shape was proposed [Ishida et al., 2001]. The method is based on Interactive Evolutionary Computation [Takagi, 2001]. Feature-based Modelling [Luby, 1986] supports both language and shapes.

Takagi's work is a biological approach. Idea is similar to crop improvement. In this method we elect a candidate shape during an evolution process with which we can obtain an outcome through repetition of that evolution. However, few applications support linguistic interaction. In particular, the method proposed by Ishida et al. supports the generation of shape images by the acquisition of an evaluation function that represents the designer's intention. However, as Tomes et al. mentioned [Tomes et al, 1998], it is difficult to generate a sufficient number of images without linguistic activity. Thus the design application requires negotiation as well as interactions between language and shapes.

In the feature-based method, feature is applied to describe the attributes of objects. It enables the expression of structures by combining relations. Luby developed Casper, a Computer-Integrated Manufacturing (CIM) system that is an application of feature-based design. He regarded features to be the presentations and dimensions required in CIM functions (such as graphics, analysis, process planning, and manufacturability evaluation), and availability as a primitive in these design processes. In the design process, however, what the required features are is unclear. In feature-based methods, clarified features have been described statically by top-down approaches. The features have recently been expanded, and they have been used to describe product information. One application is the feature-based Modelling, which is Dimensioning and Tolerance module (FbMDT) [Bley et al., 1999]. FbMDT is a method to compute dimension and tolerance by features in terms of "function view" and "manufacturing view," a new method enables us to design flexibly. However, it does not enable a bridge between natural language and shapes. Understandably, we consider language to be dynamic; thus we must bridge between linguistic description and shapes by bottom-up approaches. For these reasons, this approach is considered to be insufficiently flexible. We show a comparison of the feature-based method and our method in Figure 3.

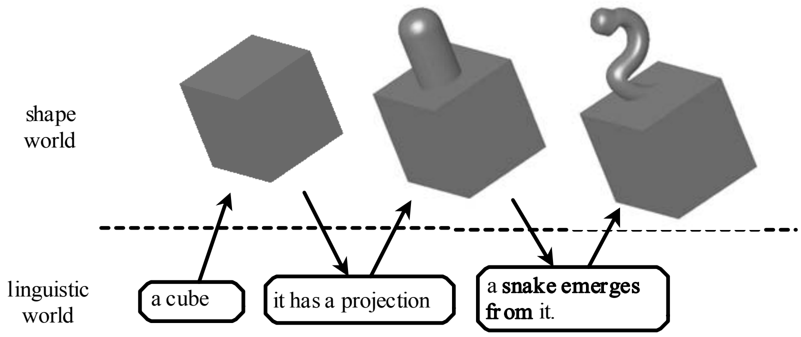


Figure 1. Negotiation between linguistic world and shape world. Our purpose is to support conceptual design by bridging natural language and shapes

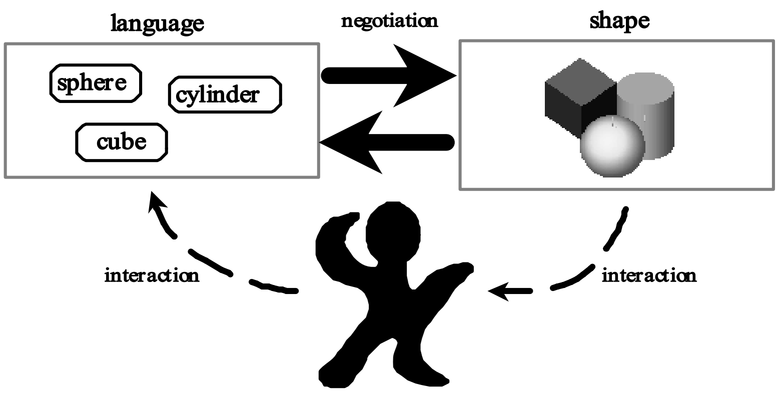


Figure 2. Interactive design method with language and shapes

To negotiate between language and shapes, we extract knowledge from a linguistic description of shapes, then match this knowledge to knowledge of shapes. The feature-based method does not support metaphorical expression. We feel that our method enables such expression (Figure 4).

We focused on an English-Japanese matching method to match linguistic knowledge and shape knowledge [Utsuro et al., 1994; Utiyama and Isahara, 2003; Dunning, 1993]. This method is based on statistics from sentences and words in texts. It is distinct from top-down approaches (i.e., rule-based matching) and enables us to express subjective characteristics for each linguistic datum. Therefore, statistical approaches enable bottom-up matching. For these reasons, we use such a statistical method.

This study was progressed with the assistance of Dr.Utiyama, Dr.Takeuchi, and the group leader Isahara, at the National Institute of Information and Communication Technology, and Associate Professor Nagai the School of Knowledge Science, Japan Advanced Institute of Science and Technology.

In this paper, we describe transcription data to extract linguistic knowledge. We then describe a linguistic model and a shape model. Next we show two methods for matching linguistic presentation and shape presentation, and we evaluate them. We draw from the results to describe our conclusions and future vision.

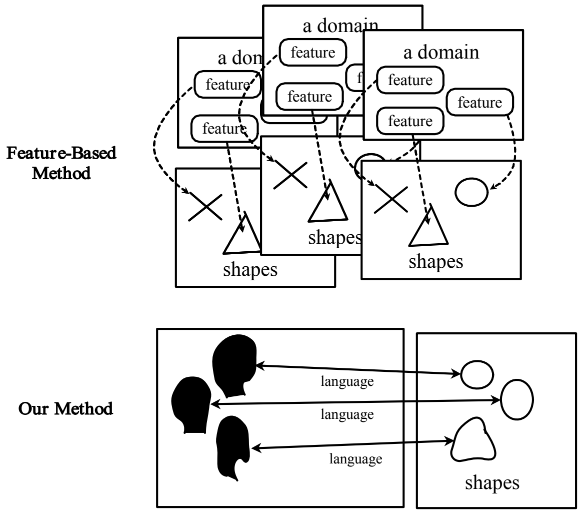


Figure 3. A feature-based method and our method: Our method enables a person to extract shapes by individual linguistic expression

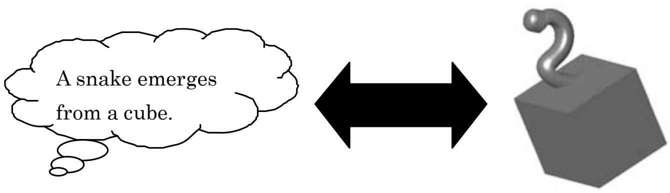


Figure 4. Matching between language and shapes by the bottom-up method. It can support metaphors, such as "snake" and "look out"

2. Models of Knowledge

In this section, we describe linguistic knowledge and shape knowledge. In psychology, we can regard shapes to be an assembly of components [Biederman, 1987]. Primitives are the basic components in the assembly. Therefore we propose the Linguistic Relational Model (LRM), which is able to describe the hierarchical structures of components and in which we use Constructive Solid Geometry (CSG). In particular, "transcription data" describes shapes. We extract knowledge from it by applying LRM.

2.1 Transcription Data

To extract knowledge by LRM, we gathered transcription data that describes the details of shape. In the experiment for compiling the data, subjects were given the following task.
"Please describe these shapes (Figure 5) in words. For example, your description should enable a person you are talking to by telephone to imagine these shapes."
We have already acquired data on 5 types of shapes from each of 9 subjects. We show a sample of the transcription data in Table 1.
In this research, we asked students at School of Knowledge Science, Japan Advanced Institute of Science and Technology, to create the transcription data. Here it is possible that the expressions are different within each domain. For example, a milling engineer uses more appropriate expressions in terms of manufacturing process. However, we believe our bottom-up approach decreases such differences of expression.

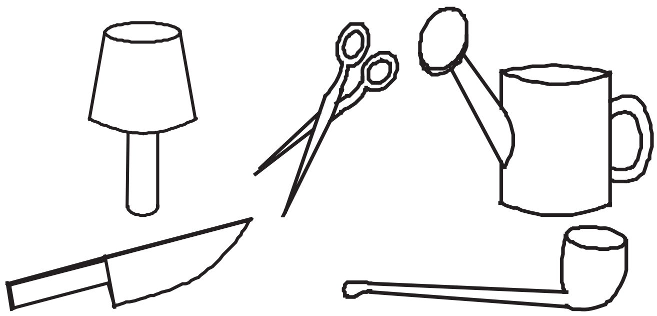


Figure 5. Shapes for transcription: We showed the above shapes to subjects to obtain transcriptions

Table 1. Sample of transcription data

A cone cut off at 1/3 the length from the vertex is put on a slender cylinder.
It looks like a watering can. An oval-shaped handle is attached at the side of the cylindrical base. A long and slender bar like a nozzle is attached to the opposite side. An oval disk is stuck onto the slender tip of the bar.
There is a very thin rectangular parallelepiped. A right triangle is attached to the end.

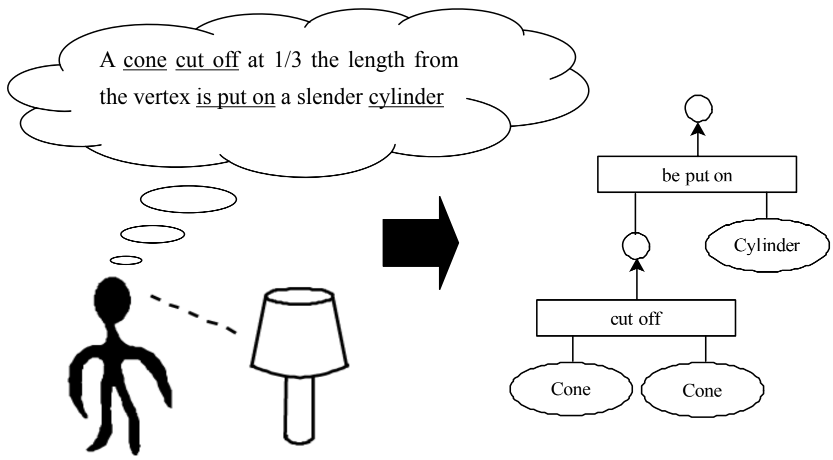


Figure 6. Example of extraction by our bottom-up approach. The right chart shows knowledge about the lamplike shape on the left

(a)

2.2 Linguistic Relational Model (LRM)

Our final purpose is to realize design on a language basis. For this purpose, we consider it indispensable to label each shape. Therefore, based on Biederman's work [Biederman, 1987], we extracted the describable labels of the shapes from the transcription data. We first paid attention to "the name of the component," such as "cylinder," and "what kind of relations does each component have?" We believe that the "relation" label appears superficially as a linguistic expression. Thus we assume that labels of components and relations in knowledge can be extracted from the transcription data by using the LRM.

Table 2. Definition of LRM: (a) symbols in LRM, (b) definition of category index in LRM, (c) syntactic rule set

(a)

A	basic expressions in LRM
F_γ	a syntactic operation
X_δ	expressions set in category δ
S	syntactic rule set
Γ	indexes for syntactic operations
Δ	category index set

(b)

category index	definition
REL	relation
COMP	component

(c)

S_{REL}	$< F_{REL}, < COMP, COMP >, REL >$
S_{COMP}	$< F_{COMP}, < REL >, COMP >$

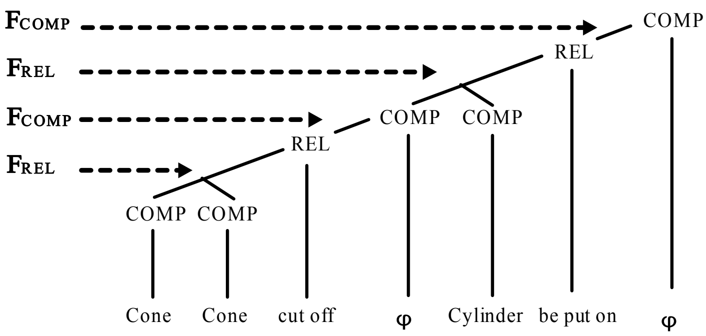


Figure 7. Generating process for LRM tree

Then the relation leads to a new structure from these components. The other relations are able to include the new structure.

In this study, “components” are nouns to describe shape. However, if we cannot imagine the shape, some adjectives are attachable (i.e., “planar-object”). The term “relation” refers to verbs to combine between each component, for example “join” and “cut off.” In the same way, extensional verbs or nouns are attachable to “relation.” We illustrate an example of description for using the LRM in Figure 6.

We define LRM as below.

$$LRM = < A, F_\gamma, X_\delta, S >_{\gamma \in \Gamma, \delta \in \Delta} \qquad (1)$$

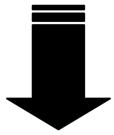
The definition of these symbols is shown in Table 2-(a). Here, “basic expression” means expression from the transcription data, and we need to translate such expressions into basic form, which is mentioned in the dictionary. “Category” means the classification of an expression on grammar and is used in syntactic operations. “Category index” means a name of such category. In LRM, Δ contains those category indexes (in Table 2-[b]). “Syntactic operation” is a function to generate a “category index” from other category indexes. Syntactic operations of this kind accept all category indexes. Thus we must define some rules for syntactic operations to filter input/output category indexes. This is “syntactic rule set.” We show description n-terms syntactic rules as below.

$$< F_\gamma, < \delta_0, \delta_1, ..., \delta_{n-1} >, \varepsilon > \qquad (2)$$

F_γ is a syntactic operation. $< \delta_0, \delta_1, ..., \delta_{n-1} >$ is an input sequence for the syntactic operation. ε is an output by the syntactic operation. Here, in LRM, we show syntactic rule set S in Table 2-(c).

Table 3. Example of data obtained using LRM: process for generating a lamplike structure

Relation	Child1	Child2	Parent
R1: cut off	C1: Cone	C2: Cone	C3: ϕ
R2: put on	C3: ϕ (cut off [cone cone])	Cylinder	C5: ϕ



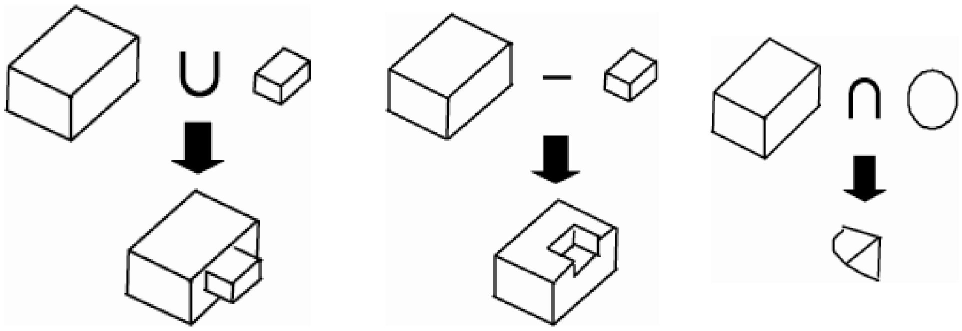


Figure 8. Operations on CSG: (a) union, (b) difference and (c) intersection

LRM consists of only 2 syntactic rules. S_{REL} is a rule to generate REL from 2 COMPs. S_{COMP} is a rule to generate COMP from a REL. In Figure 7, we show the generating process. The components labeled "ϕ" are nameless components whose name did not appear in the transcription data.

An extraction of expressions is done by hand. First we extract "components" and "relations" from transcription data. Next we describe the relationships of parent and children between "components" and "relations." In our study, we assigned ID to describe such relations. We show the actual LRM data in Table 3. The example contains two components (i.e., a cone and a cylinder) and two relations (i.e., cut off, put on).

2.3 Constructive Solid Geometry (CSG)

In section 2.2, we presented the linguistic model LRM, which is able to describe the hierarchical structures of components. On the other hand, we chose Constructive Solid Geometry (CSG) as a geometric model. In CSG, we can describe an assembly with 3 types of operations (Figure 8): union ($\bigcup$), difference (-), and intersection ($\bigcap$). We show an example of CSG in Figure 9. CSG is capable of expressing hierarchical structures as well as LRM, thus it is appropriate to match LRM.

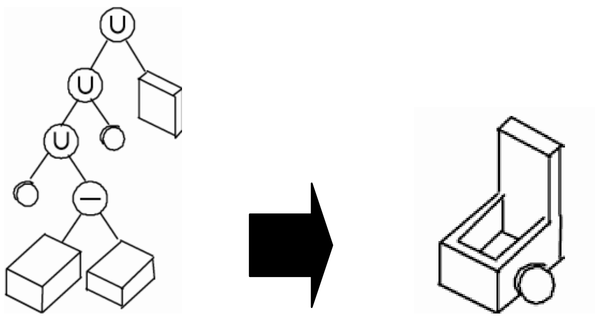


Figure 9. Example of presentation using CSG. It describes a toy cart

2.4 Relationship between LRM and CSG

We assumed that "relations" in LRM (i.e., "be attached" and "put on") are able to match "operators" in CSG (i.e., "+"), and that LRM "components" are able to match "primitives" or subtrees in CSG. Therefore, when we extracted components or relations from transcription data, we believe that components match with primitives and primitives match with operators. Thus we describe the following chapters under this assumption. However, we need to inspect what amount of match between Natural Language by LRM and expression by CSG.

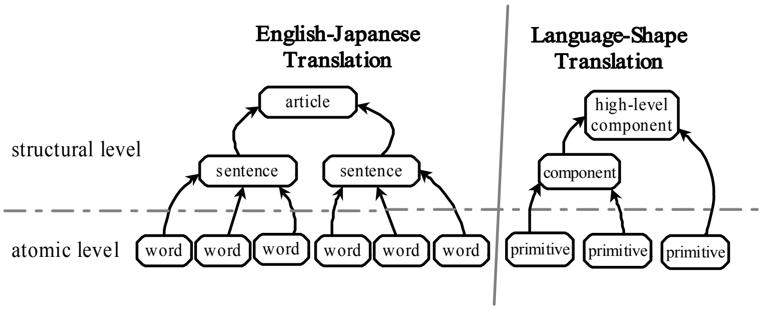


Figure 10. Atomic level and structural level

3. Method of Matching

In Natural Language Processing, the problem of matching bilingual text is known as the Bilingual Sentence Alignment Problem [Utsuro et al., 1994]. As we described in section 1, we use the approach of an English-Japanese translator. We analogize linguistic knowledge and geometric knowledge as being bilingual. We adopt statistical approaches, which are used in machine translation as matching methods. We attempt 2 levels of matching – "atomic level" and "structural level." The atomic level is each word in the Japanese-English translation [Utsuro et al., 1994; Dunning, 1993]. The structural level is each sentence or article in the translation [Utsuro et al., 1994; Utiyama and Isahara, 2003]. In our method, the atomic level is each primitive. The structural level is each component that is not primitive. In these methods, we must first perform matching at the atomic level to match each structure, and subsequently the same at the structural level. We illustrate the atomic level and the structural level in both the English-Japanese Translation system and the Language-Shape Translation system (in Figure 10).

We define C_{LRM} as knowledge described by LRM. We define C_{CSG} as knowledge described by CSG. C_{LRM} and C_{CSG} are shown in equations (3) and (4). In these equations, U and V express trees consisting of C_{LRM} and C_{CSG}. Next we define u as a subtree of U and v as a subtree of V. We then define a function child (z) as a set of nodes under node z.

$$C_{LRM} = U_1, U_2, ..., U_n \tag{3}$$

$$C_{CSG} = V_1, V_2, ..., V_n \tag{4}$$

In the following, we describe the Log-Likelihood Ratio, which is a statistical measure for atomic-level matching. Next we describe the Recursive Score for matching structural data on the basis of the definition.

3.1 Atomic-Level Matching

First we matched LRM primitives and CSG primitives. Then we matched LRM relations and CSG operators. We used the Log-Likelihood Ratio (LLR) [Dunning, 1993], which is a measure for analyzing co-occurrence words. We can apply the method to bilingual texts. Therefore this method is generally used to extract translation words from bilingual text pairs. The idea of translation by LLR is as follows. Let us evaluate the correlation between word α and another word β. Thus when we evaluate the correlation between the LRM node and the CSG node, we define random variables A1 and A2 as

$$A1 = \begin{cases} 1 & (u \in C_{LRM}) \\ 0 & (otherwise) \end{cases} \tag{5}$$

$$A2 = \begin{cases} 1 & (v \in C_{CSG}) \\ 0 & (otherwise). \end{cases} \tag{6}$$

Then by defining a1 and a2 as values of random variables A1 and A2 on knowledge U and V, we can define a random variable sequence D.

$$\mathbf{D} = (a1_1, a2_1), (a1_2, a2_2), \ldots, (a1_n, a2_n) \tag{7}$$

We hypothesize as below.

H_{indep}: A1 and A2 are independent.

H_{dep}: A1 and A2 are dependent.

We apply equation (8) to compute the possibility of generating D on hypothesis H.

$$P(D \mid H) = \prod_{i=1}^{n} P(A1 = a1_i, A2 = a2_i \mid H) \tag{8}$$

LLR is defined as equation (9) by the above definitions.

$$LLR = \log \frac{P(D \mid H_{dep})}{P(D \mid H_{indep})} \tag{9}$$

Table 4. Evaluations of matching methods: (a) atomic-level matching, (b) structural- level matching

(a)

Kind of object	Number of datums	Correct matching	Accuracy
Primitives	53	35	74%
Relations	42	22	52%

(b)

Kind of object	Number of datums	Correct matching	Accuracy
Trees of knowledge	42	29	69%
Subtrees whose a component is a root	184	8	4%
Subtrees whose a relation is a root	91	32	35%

Table 5. Examples of matching at the atomic level between LRM and CSG. (a) The primitives of CSG are labelled correctly. (b) The primitives of CSG are labelled incorrectly

(a)

Label of LRM	Primitive of CSG
cylinder	Cylinder
umbrella	Cone
parallelepiped	Brick
bean	Sphere
lamp	Sphere

(b)

Label of LRM	Primitive of CSG
disk	Arc
planar object	Arc
arc	Sphere
bowl	Arc
pipe	Sphere

3.2 Structural Level Matching

In section 3.1, we described the method for atomic level matching. However, the shapes consisting of components are structural. Therefore, structural level matching is required to label abstract components.

Utsuro et al. mentioned the generic method for matching bilingual sentences [Utsuro et al, 1994]. It enables us to match 2nd level structural data (a sentence containing words). Utiyama and Isahara proposed SntScore [Utiyama et al, 2003]. It enables us to match 3rd level structural data (an article including sentences that contain words). On the other hand, in this study, the LRM knowledge or CSG has an n level hierarchy. Thus we expanded those ideas, and we proposed Recursive Score, which is a statistical measure for matching n level structural data. We define the Recursive Score in the next equations (10) (11).

$$\mathrm{RecScore}_{v}(\mathrm{u},\mathrm{v}) = \frac{\mathrm{V}_{\mathrm{avg}}(\mathrm{u}, \mathit{child}[\mathrm{v}]) + h(u,v)}{2} \tag{10}$$

$$\mathrm{RecScore}_{u}(\mathrm{u},\mathrm{v}) = \frac{\mathrm{RecScore}(\mathrm{u},\mathrm{v})_{\mathrm{v}} + \mathrm{U}_{\mathrm{avg}}(\mathrm{child}[\mathrm{u}],\mathrm{v})}{2} \tag{11}$$

Then let h (u, v), U_{avg}, and V_{avg} be expressed as equations (12), (13), and (14).

$$h(u,v) = \frac{\mathrm{co}(\mathrm{u},\mathrm{v})}{n_u + n_v} \tag{12}$$

$$\mathrm{V}_{\mathrm{avg}}(\mathrm{u},\mathrm{v}) = \frac{1}{|\mathrm{V}|}\sum_{\mathrm{v}=\mathrm{V}} \mathrm{RecScore}_{v}(\mathrm{u},\mathrm{v}) \tag{13}$$

$$\mathrm{U}_{\mathrm{avg}}(\mathrm{u},\mathrm{v}) = \frac{1}{|\mathrm{U}|}\sum_{\mathrm{u}=\mathrm{U}} \mathrm{RecScore}_{u}(\mathrm{u},\mathrm{v}) \tag{14}$$

co(u,v) represents the correlation between u and v. We use LLR to compute the correlation. If u is a leaf node, let U_{avg} be 0. If v is a leaf node, let V_{avg} be 0. We consider the problem to be the optimization of u and v in equation (15).

$$\mathrm{RecScore}_{u}(\mathrm{u},\mathrm{v}) = \max_{u\in U, v\in V} \frac{\mathrm{RecScore}_{v}(\mathrm{u},\mathrm{v}) + \mathrm{U}_{\mathrm{avg}}(\mathrm{child}[\mathrm{u}],\mathrm{v})}{2} \tag{15}$$

4. Evaluation

We extracted knowledge from the transcription data presented in section 2.1 by LRM. Then we tried matching it to the CSG presentation in section 2.3. We evaluated atomic-level matching by LLR in section 3.2 and structural-level matching by the Recursive Score in section 3.2.
In this study, we assume there are 7 kinds of primitives: Cone, Brick, Cylinder, Ring, Pyramid, Arc, and Sphere. We consider that there are 3 types of operators--union, difference, and intersection--as described in section 2.3.
We performed the evaluation with the following equation (16), which gives the accuracy. Here let n be the total number of evaluated objects, and c the number of correctly matched objects. We must let n and c for each evaluated object.

$$Accuracy = \frac{c}{n} \tag{16}$$

4.1 Evaluation of Atomic-Level Matching

We tried matching phrases concerning components in knowledge extracted by LRM and components in CSG presentation. Next, we also tried to match phrases concerning relations in LRM and operators in CSG. In our study we define whether matching is correct or incorrect, in terms of the candidate with the highest score. If the candidate is the correct object, we regard the matching to be correct. For example, if "rounded object" in LRM is matched to "brick" in CSG, it is an incorrect match. We performed the evaluation by using equation (16); let n be 53 and c be 35 in the case of matching primitives; then let n be 42 and c be 22 in the case of matching relations. We show the results of the evaluation in Table 4-(a). We also show examples of correctly matched atoms and incorrectly matched atoms in Table 5.

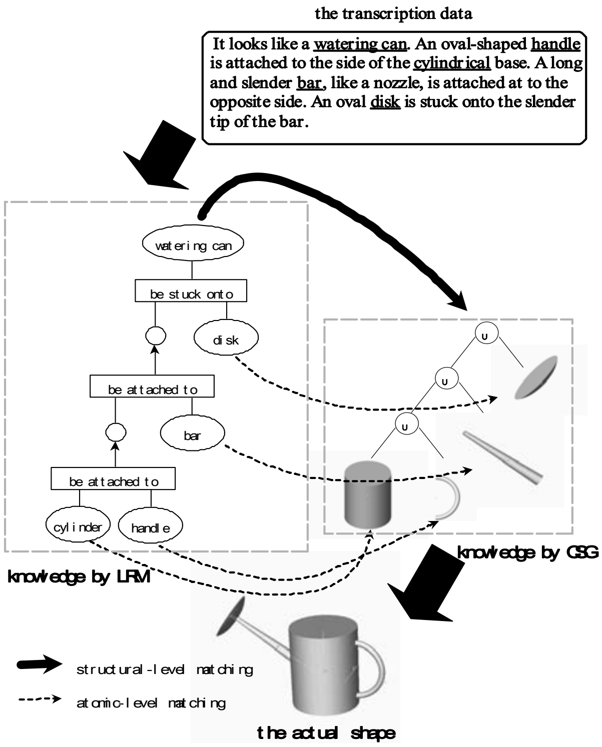


Figure 11. Concept of the matching between linguistic knowledge and shape knowledge. This is an example of the shape resembling a watering can

4.2 Evaluation of Structural-Level Matching

We matched each tree of knowledge on the basis of the Recursive Score. There are 42 kinds of trees of knowledge obtained by LRM, and 5 kinds obtained by CSG. The idea of matching is as follows. We evaluated the match between a tree of knowledge by LRM and one by CSG. Twenty-nine kinds of LRM trees were matched correctly. We used equation (14) for the evaluation; let n be the total number of LRM trees and c the number of correctly matched trees.

Next, we carried out matching between subtrees in already matched trees. We could correctly match 8 nodes in 184 subtrees whose roots are component nodes. Then we could correctly match 32 nodes in 91 subtrees whose roots are relational nodes. We used equation (14) for the evaluation. Let n be the total number of LRM subtrees and c the number of correctly matched subtrees. We show the results of evaluations in Table 5-b.

5. Conclusion

In summary, we proposed LRM, a language-based model. Next, we extracted structural knowledge from the transcription data. Then we matched linguistic knowledge and CSG expressions.

In this study, we were able to accomplish atomic-level matching with few samples. However, as we showed in Table 5, the accuracy was not very good when metaphorical expression was included.

On the other hand, as we showed in Table 4-b, we also considered structural-level matching. The accuracy in the matching-trees problem was fair. The accuracy in the matching-subtrees problem was poor. The accuracy of structural matching depends on the result of atomic-level matching. Thus the accuracy of matching grows worse for each subtree that includes relational nodes and operator nodes. We believe the solutions to be noise proof because of increasing transcription data, the use of the Threshold Function [Utsuro, 1994] and use of dictionary in the statistic measure. However, we do not believe the dictionary should be not standardized, and we think it should be constructed from the transcription data by bottom-up approach. In addition, our experiment was done on small number of datum. Thus the data sparseness problem was significant. Therefore we desire to increase number of datum.

In conclusion, the results described above indicated that our method enabled the following matchings:

1. of each primitive component,
2. of each structural tree.

The concept of our method is shown in Figure 11. In this paper, the main viewpoint was the structure. However, it is also necessary to consider adjectives such as “large,” “round” and “strange” to express the component properties. Then we must also describe phrases such as “at the shorter edge” and “in series” to express locations. Our model on expressiveness should be improved by including those properties. In this study, we used a pre-determined uni-gram phrase. We hope to adopt bi-gram or n-gram to include such adjectives, because these approaches make use of main words and adjusts attached words simultaneously.

For the final purpose, we wish to realize a framework for creativity support based on mutual language -shape by resources in this study.

Acknowledgements

We thank Dr. Nagai’s students for providing the transcription data. We would also like to thank Min Ann and Tsuyoshi Maeda for their advice and data, and Keita Hashimoto for transcribing the data.

References

Biederman, I., “Recognition-by-Components: A Theory of Human Image Understanding,” the American Psychological Association, 1987, Vol. 94, No. 1, pp. 115-147.

Bley, H., Oltermann, R., Thome, O., Weber, C.: “A tolerance system to interface design and manufacturing,” Proceedings of the 6th CIRP International Seminar on ComputerAided Tolerancing, pp.149-156.

Dunning, T., “Accurate Methods for the Statistics of Surprise and Coincidence,” Computational Linguistics, 1993, Vol. 19, No. 1, pp. 61-74.

Ishida, R., Shiose, T., and Taura, T., “A study on interactive shape representation via strategic biasing,” In 28th SICE Symposium on Intelligent Systems, 2001, pp. 147-152.

Luby, S. C., "Creating And Using A Features Data Base," Computer in Mechanical Engineering, 1986, Vol. 5, No. 3, pp 25-33.

Pahl, G., Beitz, W., Engineering Design: a systematic approach, Springer-Verlag, 1988, Chapter 5.

Takagi, H., "Interactive Evolutionary Computation: Fusion of the Capacities of EC Optimization and Human Evaluation," Proceedings of the IEEE, 2001, Vol. 89, No. 9, pp. 1275-1296.

Tomes, A., Oates, C., Peter. A., "Talking design: negotiating the verbal-visual translation," Design Studies 19, 1998, pp. 127-142

Utiyama, M., Isahara, H., "Reliable Measures for Aligning Japanese-English News Articles and Sentences," ACL-2003, 2003, pp. 72-79.

Utsuro, T., Ikeda, H., Yamane, M., Matsumoto, Y., Nagao, M., "Bilingual Text Matching using Bilingual Dictionary and Statistics," In COLING'94, 1994, pp. 232-241.

Shinji Miura
Taura Laboratory
Graduate School of Science and Technology, Kobe University
1-1 Rokkodai, Nada-ku, Kobe, Japan
Tel.: 078-803-6503
Fax.: 078-803-6503
Email: miura@ma-5.scitec.kobe-u.ac.jp