Text Parsing for Sign Language Generation with Combinatory Categorial Grammar

Jin-Woo Chung
Computer Science Department
KAIST
Daejeon, South Korea
jwchung@nlp.kaist.ac.kr

Jong C. Park
Computer Science Department
KAIST
Daejeon, South Korea
park@nlp.kaist.ac.kr

ABSTRACT
In this paper, we propose a method to convert a written sentence in spoken language into a suitable representation in sign language within the framework of Combinatory Categorial Grammar (CCG). The representation reflects the multi-channel nature of sign language performance, including manual and non-manual linguistic signals of multiple channels and information about their coordination. We show that most information needed to address linguistic phenomena in sign language such as word order, spatial references, classifier construction, and verb inflection can be encoded in the CCG sign lexicon. During the CCG derivation process, a semantic representation for sign language expressions is created so that the resulting output can be directly interpreted as a sequence of signs, each containing manual and non-manual components and representing their coordination and spatial relationship. The derivation process with the constructed lexicon is presented with several examples for Korean Sign Language. We discuss implications of our proposal and future directions.

Categories and Subject Descriptors
I.2.7 [**Artificial Intelligence**]: Natural Language Processing - *Language generation, Machine translation, Text analysis*

General Terms
Algorithms, Languages

Keywords
Korean Sign Language, Natural Language Generation, Combinatory Categorial Grammar

1. INTRODUCTION
In recent years, many attempts have been made to automatically generate sign language utterances from written text in spoken language to overcome the communication gap between hearing and deaf users [1]. They focus on the fact that textual information is not much accessible to deaf users and that sign languages are their native languages whose linguistic characteristics are distinct from those of spoken languages, to the effect that there should be some linguistic processing for appropriately mapping the two types of languages.

Like other machine translation researches, most of the current studies on generating sign language representations or animation from written text are based on two paradigms: rule-based (or grammar-based) and statistics-based. While recent studies on machine translation between pairs of spoken languages mostly adopt the statistics-based paradigm with an annotated parallel corpus, many researches in the sign language domain have exploited grammar formalisms originally designed for analyzing text in spoken languages [2-4], since no such annotated corpus is yet available to provide reliable statistics for translation. The common problem of the rule-based paradigm is that the coverage and applicability of the manually constructed lexicon and rules are fairly limited, so that most funded projects have focused on a restricted domain such as weather forecast and eGovernment services for practical results. Recently, corpus-based researches on sign language translation appeared [5-8], but they are still at an early stage.

In this paper, we present a method to generate a semantic representation for sign language expressions from a written sentence based on a grammar formalism. The resulting representation consists of a sequence of signs with appropriate parameters to account for various linguistic phenomena that occur in sign language. Since this sequence corresponds to the actual order in which each sign in the translated sentence is expressed, it can be easily interpreted as a sign language expression. In this paper, we restrict out attention to the task of analyzing original text in spoken languages and generating a semantic representation for corresponding sign language animation, but exclude animation synthesis itself. In particular, the output of our system does not include any information about detailed parameters for animation synthesis, which is left for future work. We will however briefly present the approach of converting the final output into another transcription system – HamNoSys [9].

In order to account for various phenomena of sign language and make such an account explicit in the representation obtained by the derivation process, it is important to design the lexicon carefully since important information for the derivation that is based on lexicalized grammars such as CCG is all encoded in the lexicon. We will hence focus on explaining what kind of information must be encoded and how it is represented in the CCG sign lexicon.

The rest of this paper is organized as follows: Section 2 gives an overview of Combinatory Categorial Grammar (CCG) as used in our system. Section 3 explains how to construct a CCG sign lexicon, which is a key part of CCG parsing and derivation. Section 4 presents the process of generating a semantic representation for sign language expressions with the constructed

SLTAT 2011, 23 October 2011, Dundee, UK

lexicon. Section 5 discusses implications of our study and Section 6 concludes the paper.

2. COMBINATORY CATEGORIAL GRAMMAR

Combinatory Categorial Grammar (CCG) is a highly lexicalized, efficiently parseable, yet linguistically expressive grammar formalism. It has a transparent interface between surface syntax and underlying semantic representation and thus can model difficult linguistic phenomena, such as coordination, long distance extraction and topicalization, without further stipulation [10, 11]. CCG has been widely used for computational modeling of natural language sentences in many languages, including Korean [12-16]. The process of deriving syntax and semantics from a written sentence using CCG is exemplified in Figure 1. Each lexical item is assigned one of the possible categories that consist of the basic categories such as S (sentence) and NP (noun phrase) and left or right slashes indicating the directionality of arguments. For example, the following category for the transitive verb *아부하다*(*flatter*) in Korean takes two NP arguments on its left indicated by each backslash '\' (since Korean is an SOV language) and produces the result as a sentence of the category S.

flatter `:= (S\NP)\NP`

During the derivation process, two adjacent categories are combined into one by function application and combinatory rules (Table 1) in a bottom-up fashion. After this process is completed, one semantic form of the category S is derived, which corresponds to the semantic interpretation of a whole sentence. For further explanation and computational motivation for this theory of grammar, the reader is directed to [10, 11], among others.

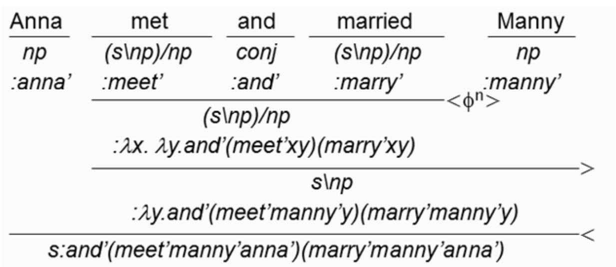

Figure 1. CCG derivation

Table 1. CCG function application and combinatory rules

Rule	Name	Symbol
X/Y Y → X	Forward Application	>
Y X\Y → X	Backward Application	<
X/Y Y/Z → X/Z	Forward Composition	>B
Y\Z X\Y → X\Z	Backward Composition	<B
X → T/(T\X)	Forward Raising	>T
X → T\(T/X)	Backward Raising	<T
X conj X → X	Coordination	<§>^{n}

3. CCG SIGN LEXICON CONSTRUCTION

We follow previous studies of CCG in assigning two categories to each lexical item in our method, syntactic and semantic categories, but we introduce slightly different representations for both categories.

3.1 Syntactic Category

We represent syntactic categories for clauses and noun phrases as `s` and `np`, respectively. We also distinguish categories for three kinds of clauses: `s` for a clause that has a sentence-final ending, s_o for a clause that has embedded clauses and does not have a final ending, and s_u for a clause that has neither embedded clauses nor a final ending.

In addition, we introduce some new features for the basic categories to distinguish them by their role in the sentence and to impose constraints on their unification. The basic category for the clause with a final ending (`s`) has a feature indicating its sentence type: `dec` for declarative, `imp` for imperative, and `int` for interrogative. The feature for the category s_o indicates the type of its embedded clause: `cond` for conditional, `temp` for temporal, and `caus` for causal. The features for the basic category of noun phrases (`np`) are of three kinds: 1) the case of a head noun (`nom` for nominative, `acc` for accusative, and `dat` for dative), 2) an indication whether a head noun is a pronoun (`+pro`) or not (`-pro`), i.e. whether or not it uses a particular location of the signing space for spatial references, and 3) an indication whether both hands are used to perform a head noun (`+both`) or not (`-both`). An example of adding feature sets to the category `s\np/np` is shown below:

```
s(dec)\np(nom,-pro,-both)/np(dat,+pro,-both)
```

Obviously, more feature values or another type of feature would be needed to explain other possible phenomena. Although it is not unreasonable to extend a feature set for each basic category, we use the small set described above just to explain the key phenomena.

3.2 Semantic Form

The semantic forms as used in our method are different from those of typical lambda calculus. We no longer use the conventional predicate-argument structure for the semantic representation as in Figure 1, but rather represent it as a sequence of signs consisting of multiple linguistic channels that are necessary for distinguishing the meaning of words in sign language, such as hand shape, hand position, eye gaze, and head nodding. This representation is intended to incorporate these channels and their coordination. We can interpret it as a sign language expression by reading off a sequence of glosses and their parameters one-by-one.

The semantic forms are attached to the right of the syntactic category via the colon ':' operator, but what is different from the conventional form is that they are attached to each of the basic syntactic categories `s` and `np`, as shown in bold face in the following example.

```
s(dec):sem1
\np(nom,-pro,-both):sem2
/np(dat,+pro,-both):sem3
```

Each semantic form for `s` and `np` (**`sem1`**, **`sem2`**, and **`sem3`** above) consists of a single sign gloss or a sequence of glosses. Each gloss consists again of four parts: gloss name, signal for the dominant hand, signal for the non-dominant hand, and signal for non-manual components. More specifically, they are represented as follows.

```
gloss_name(dominant hand signal,
           non-dominant hand signal,
           non-manual signal)
```

The gloss name is used to identify a single sign by its name. The other three signals are parameters for the sign. They consist of multiple channels each corresponding to a ‘phoneme’ of sign language. We use 3 channels each for the dominant hand and non-dominant hand: hand shape, hand position, and hand movement. We also use 4 non-manual channels: eye gaze, head movement, eyebrows, and mouthing. Each gloss thus has 10 channels in total. The reason why we introduce separate channels for dominant and non-dominant hands is that there are cases where each hand plays a different linguistic role and that they should thus be treated separately to explain linguistic phenomena appropriately. Note, however, that there are definitely more channels (or phonemes) in reality such as palm orientation, shoulder movements, and eye lids, but we use only these 10 channels for the convenience of exposition because they are sufficient enough to explain various phenomena dealt with in this paper. We also believe that the proposed method can accommodate the additional channels in a straightforward manner.

Each channel may have a specific range of values as shown in Table 2. We assume that for every gloss each channel has a default value even when no value is explicitly specified for the channel during the parsing process. In this case, its value is denoted as the hyphen ‘-’.

Table 2. Linguistic channels for sign language expressions

Channel		Type of value	Example
Manual channels	Hand shape (hs)	Name of a hand classifier	male, female
	Hand position (hp)	Position number	0, 4, 9
	Hand movement (hm)	Position number	0, 4, 9
Non-manual channels	Eye gaze (eg)	Position number	0, 4, 9
	Head movement (h)	Name of movement	shaking, bowed
	Eyebrows (eb)	Name of movement	raised, furrowed
	Mouthing (m)	Name of Mouth shape	oh, ah

As for hand position, hand movement, and eye gaze, the number indicating a position in signing space can be specified. We use 11 distinct positions, as shown in Figure 2, to place entities in signing space for spatial references. Although the division of signing space in Figure 2 is rather simplified, and the coordinate system deaf users actually have in mind would be much more complex, it is sufficient to use such 11 discrete positions to show that our approach can account for verb agreements and spatial references.[1]

Figure 2. Signing space

In Table 2, the position number for hand position indicates the location of the dominant (or non-dominant) hand in signing space while a sign is performed. This value is specified only when the position can be spatially referenced later. Otherwise, it is left unspecified (‘-’). As for eye gaze, the position number indicates the position on which eye gaze is fixed. The position number for hand movement is specified only when a directional verb is used, i.e. when initial and final locations of the movement are syntactically (agreement verbs) or topographically (spatial verbs) meaningful. For these verbs, the position numbers for hand position and hand movement are used as the initial and final locations, respectively. For example, if the dominant hand of the KSL agreement verb ‘GIVE’[2] has the value 3 for hand position and 4 for hand movement, entities in positions 3 and 4 are regarded as its subject and object, respectively, and the movement is made from position 3 to position 4.

The representation for each gloss can be rewritten as follows, incorporating channels for manual and non-manual signals:

```
gloss_name(d(hs,hp,hm), n(hs,hp,hm), nms(eg,h,eb,m))
```

d(…) and n(…) are ordered sets of channels for dominant and non-dominant hands, respectively. nms(…) is an ordered set of non-manual channels. Each channel in each set is denoted as an abbreviation of its name such as hs (hand shape), hp (hand position), and hm (hand movement), also as shown in Table 2. An example of assigning values to its channel in the representation above is shown below:

```
give(d(money,3,4), n(male,4,-), nms(-,-,raised,oh))
```

This gloss represents ‘GIVE’ in KSL. The dominant hand is moved from position 3 to position 4 with the ‘money’ classifier shape. The non-dominant hand is fixed in position 4 with the ‘male’ classifier shape. Eyebrows are raised and the mouth is in the ‘oh’ shape. Considering that the verb ‘GIVE’ moves its dominant hand in the subject position towards its non-dominant hand in the object position, the gloss above is interpreted as “Someone (position 3) gives money to him (position 4)”. In fact, such verbs play an important role in determining the entire structure of a sentence because they are mostly functors in CCG and their category thus indicates how many and what kind of arguments they take.

---

[1] We do not intend in this paper to deal with all aspects of the use of signing space, which are actually quite challenging to cover in a single paper. Our current approach thus models only the syntactic use of the signing space in a simplified way, as presented in Section 4.

[2] In this paper, we denote KSL and Korean words differently to make a distinction between them. We denote KSL words in upper case enclosed by single quotation marks such as ‘GIVE’ and ‘TAKE’. As for Korean words, we denote them in lower case and italic type such as *give* and *take*.

Most channels of the semantic form for functors (like that of *give*) are initially left unspecified, and their values are assigned, or realized through unification with other arguments during the derivation process. In this case, we represent these channels as a variable to indicate that its value will be specified later. Variables start with a letter in upper case followed by the abbreviation of the corresponding channel, such as `Xhs`, `Xhm`, and `Xeb`. Variables are also used for the gloss name (`Xi`) and manual (`Xd`, `Xn`) or non-manual signals (`Xnms`). The following shows an example of the representation with variables.

```
Xi(d(Xhs,Xhp,Xhm), n(Xhs,Xhp,-), Xnms)
```

Here the hyphen '`-`' is used for the hand movement channel of the non-dominant hand because, as mentioned earlier, this channel has a default value and thus its value does not need to be explicitly specified in the representation. We also replace the manual or non-manual signal with a single hyphen when all of its channels have a default value, i.e. we use '`-`' instead of `d(-,-,-)`, `n(-,-,-)`, and `nms(-,-,-,-)`, as shown below.

```
Xi(d(Xhs,Xhp,Xhm), n(Xhs,Xhp,-), -)
```

In addition, we use the asterisk '`*`' for particular channels in the representation when we want to indicate that the channel can be unified with any values and variables. Again, this symbol can also replace the manual or non-manual signal as shown below when all of its channels have a default value.

```
Xi(d(Xhs,Xhp,*), *, -)
```

We often see that when two signs are performed one after another, one of the dominant and non-dominant hands holds during the second sign, and only the other hand performs differently. In this case, we denote such a fixed hand by using the subscript 'c' (meaning 'continuous') such as d_c, n_c, and nms_c, as shown below. Obviously, this notation works only when the previous sign exists.

```
Xi(d(Xhs,Xhp,Xhm), n_c, -)
```

We also introduce an additional marker for each gloss to indicate its semantic role in a sentence such as agent, goal, and theme. It is attached to the left of a gloss and connected by the equal sign '`=`', as follows.

```
agent=Xi(d(Xhs,Xhp,Xhm), n_c, -)
```

The semantic form for clauses (s, s_o, s_u) contains a sequence of such glosses enclosed by matching square brackets. The following shows the category of the verb *give*.

```
s(dec): [agent=Xi(d(-,Xhp,-), -, -),
         goal=Yi(d_c, n(-,Yhp,-), -),
         theme=Z,
         action=give(d(-,Xhp,Yhp), -, -)]
\np(nom,+pro,-both): Xi(d(-,Xhp,-), -, -)
\np(dat,+pro,-both): Yi(-, n(-,Yhp,-), -)
\np(acc,-pro,+both): Z
```

The assignment of the category above is done prior to the derivation process. Each variable is assigned an appropriate value by unification with another category. It is shown that the position of the dominant hand of the nominative NP (`Xhp`) and the position of the non-dominant hand of the dative NP (`Yhp`) are also used for the initial and final locations of the verb 'GIVE' as follows, which reflects the directionality of agreement verbs.

```
action=give(d(-,Xhp,Yhp), -, -)
```

The following example shows one of the possible results of the derivation, which is the representation for the sentence *he gives me a key*. In this case, the verb takes three arguments *he*, *I*, and *key*.

```
s(dec):[agent=he(d(-, 3, -), -, -),
        goal=i(d_c, n(-, 1, -), -),
        theme=key(-, -, -),
        action=give(d(-, 3, 1), -, -)]
```

It can be represented as shown in the diagram in Figure 3 which enumerates the four glosses from top to bottom (the non-manual signal is omitted). It is shown in the figure that the dominant hand for the first sign 'HE' holds while the non-dominant hand for the following sign 'I' performs. As mentioned earlier, the hyphen '`-`' indicates that the corresponding channel has a default value. The KSL expression for this diagram is illustrated in Figure 4.

Gloss name	(dominant hand) hs	hp	hm	(non-dominant hand) hs	hp	hm
'HE'	-	3	-	-	-	-
'I'				-	1	-
'KEY'	-	-	-	-	-	-
'GIVE'	-	3	1	-	-	-

time ↓

Figure 3. A gloss diagram for *he gives me a key*

'HE' 'I' 'KEY' 'GIVE'

Figure 4. A KSL expression for *he gives me a key*

3.3 Linguistic Phenomena of KSL

Since CCG is a highly lexicalized grammar, it is important to assign suitable categories to each lexical item so that the phenomena of sign language are well reflected in the resulting representation through the derivation process. In order to accomplish this, lexicon developers must be well aware of such phenomena and be able to turn them into statements. We focus on several phenomena that are manifested in the following example sentence, and show how to construct a CCG sign lexicon to derive an appropriate semantic representation.[3]

"존이 그녀에게 아부하고 돈을 준다면, 나는 그를 돕지 않겠다."

$John_{nom}$ her_{dat} flatter (and) $money_{acc}$ give (if) I_{nom} him_{dat} help will not

If John flatters her and gives money to her, I will not help him.

Figure 5. An example sentence

[3] Note that anaphora resolution, for instance that of identifying the referent of the pronoun *him* with that of the proper noun *John*, is not addressed in the present paper.

The example in Figure 5 has some linguistic issues of KSL to be resolved for appropriate translation as follows.[4]

(1) The word order of KSL (subject, object, and verb) is stricter than that of Korean because, unlike Korean, KSL does not use case markers.

(2) Directional verbs use the dominant hand for the subject (or the initial location) and the non-dominant hand for the object (or the final location).

(3) For directional verbs, the dominant hand holds when the non-dominant hand performs.

(4) Eye gaze is on the position of the object while signing directional verbs.

(5) The two non-manual signals 'eyebrows raised' and 'head bowed' appear while signing verbs in a conditional clause (if-clause). This is also applied to verbs in a coordinate structure; in the example above, these non-manual signals appear in both 'FLATTER' and 'GIVE'.

(6) While signing the agreement verb 'GIVE', its dominant hand has a classifier shape of its object; in the example above, the dominant hand of 'GIVE' takes the 'money' classifier shape.

(7) When the negative 'NOT' is used, headshake lasts while signing the preceding negated verb ('LIKE' in Figure 6) and following the negative 'NOT'. Eyebrows are kept furrowed during the sequence of three signs: the object of the negated verb, negated verb, and negative 'NOT', as shown in Figure 6.

I_{nom} him_{acc} like not
나는 그를 좋아하지 않는다.
headshake
eye brows furrowed

Figure 6. A negative sentence and non-manual signals for *I don't like him*

The category of *give* for the sentence *he gives me a key* we described above is not appropriate for explaining some of these phenomena (especially the sixth one) because the usages of the agreement verb 'GIVE' in the two sentences *he gives me a key* and *he gives me money* are slightly different in KSL; more specifically, it is because of the different type of their direct object. The category of *give* that takes the direct object *money* is encoded as follows:

```
s_u: [agent=Xi(d(-,Xhp,-), -, -),
      goal=Yi(d_c, n(-,Yhp,-), -),
      theme=Zi(d(Zhs,Xhp,-), n_c, nms(Xhp,-,-,Zm)),
      action=give(d(Zhs,Xhp,Yhp), n_c, nms(Yhp,-,-,Zm))]
\np(nom,+pro,-both): Xi(d(-,Xhp,-), -, -)
\np(dat,+pro,-both): Yi(d(-,Yhp,-), -, -)
\np(acc,+pro,-both): Zi(d(Zhs,*,-), -, nms(-,-,-,Zm))
```

While the hand shape and position of 'KEY' are not significant, those of 'MONEY' have some importance. The hand position of 'MONEY' indicates the person who has the money. We can see that this is reflected in the category above; the hand position (`Xhp`) of the direct object (or `theme`) is the same as that of the subject (or `agent`), meaning that the money is owned by the subject. It is also shown that the hand shape of 'MONEY' (Figure 7) is the same as that of 'GIVE', which explains the sixth phenomenon. Figure 8 illustrates the performance of two kinds of verbs, one with no classifier shape (used for *give a key*) and the other with the classifier shape for 'MONEY' (used for *give money*).

Figure 7. The classifier shape for 'MONEY' in KSL

Figure 8. Two kinds of 'GIVE' – the one on the left without a classifier shape and the one on the right with the classifier shape for 'MONEY'

Thus, different categories must be assigned to *give* according to the type of its direct object. This can be explained by the different syntactic category assigned to its NP argument; the categories of *key* and *money* are `np(acc,-pro,+both)` and `np(acc,+pro,-both)`, respectively.

We can see that the value of channels of a particular word can be propagated to those of another word by unification. We show in the next section that the appropriate assignment of categories to each lexical item leads eventually to an appropriate semantic representation for the entire sentence through the derivation process.

4. CCG DERIVATION

4.1 Signing Space

In sign language generation, mapping discourse entities to signing space is another challenging task. We only deal with the syntactic use of signing space in a rather simple way. While assigning categories to lexical items, one of the positions in Figure 2 is also assigned to each entity, but some special entities are associated with a particular position. For example, we assign position 1 to 'I' and positions 3 and 4 to the subject and object of agreement verbs, respectively. As for spatial verbs, the initial and final locations of their movement are also assigned to positions 3 and 4 for convenience. We assume that once a particular position is assigned, the assignment lasts until that position is assigned to another entity. The entity can be referenced across sentences by its associated position.

Several issues on signing space allocation still remain to be resolved. Modeling the topographical use of signing space for spatial verbs is one of them. A more challenging issue is to

[4] Although we address the translation of Korean into Korean Sign Language (KSL), the distinction between KSL and Signed Korean (or any other Korean-like sign languages) is not yet established clearly and without controversy. In this paper we focus mainly on core properties of KSL which are markedly different from Korean.

determine how to assign locations effectively when a number of entities are mentioned alternately in a discourse. Although they are beyond the scope of the present work, they need to be studied further for meaningful translation.

4.2 Derivation Process

As mentioned in Section 3.3, the most important part of our proposal is the construction of a CCG sign lexicon, i.e. the assignment of categories to each lexical item. Once appropriate categories are assigned, the remaining process is just to unify adjacent items by function application and combinatory rules in a bottom-up fashion. In this section, we show with several examples that the linguistic phenomena discussed in Section 3.3 are adequately resolved by the derivation process.

Word order. One of the important differences between Korean and KSL is that Korean has relatively free word order but KSL does not, so it is necessary to appropriately map their different word orders during the translation. In order to address this problem, we make use of another category that just takes arguments in a different order. For example, each category for *like* in Figures 9 and 10 takes NP arguments in a different order. The derivation process in Figure 9 combines *like* and *him* first, filling a slot for an object in the category of *like*, but the process in Figure 10 combines *like* and *I* first, filling a slot for an object later. In this way, two Korean sentences in different word orders are converted into KSL sentences in the same word order. This is also applied to more complex categories such as agreement verbs 'GIVE' and 'FLATTER'.

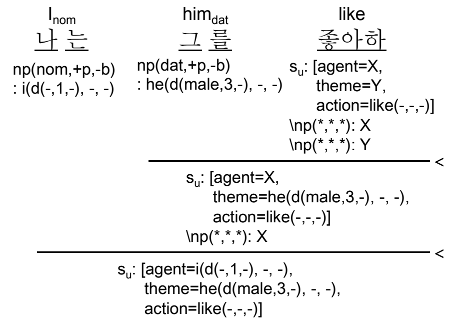

Figure 9. The derivation process for *I like him* in the word order SOV

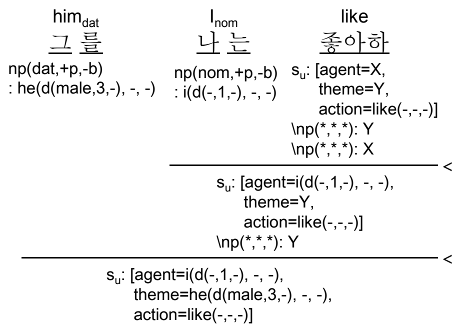

Figure 10. The derivation process for *I like him* in the word order OSV

Verb inflection. The verb inflection in sign language, especially for agreement and spatial verbs, is a typical phenomenon that distinguishes it from spoken language. Figure 11 illustrates how the KSL agreement verb 'FLATTER' is inflected after the derivation process. The representation for the gloss `flatter` in the derivation result indicates that it is inflected by its object *her*. More specifically, both hands perform in position 4 which is assigned to the object *her* with the 'female' classifier shape of its non-dominant hand. It is also shown that eye gaze is also on position 4 while signing 'SHE' and 'FLATTER'.

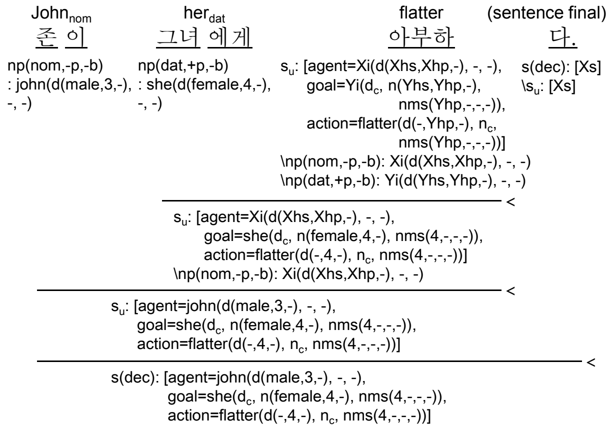

Figure 11. The derivation process for *John flatters her*

Negative sentence. As mentioned in Section 3.3, some non-manual signals appear in a negative sentence. The following category for *not* can account for such phenomena.

$$
\begin{aligned}
&\texttt{s}_{\texttt{u}}\texttt{: [agent=X,}\\
&\quad\texttt{theme=Y}_{\texttt{nms(*,*,furrowed,*)}}\texttt{,}\\
&\quad\texttt{action=Z}_{\texttt{nms(*,shaking,furrowed}_{\texttt{c}}\texttt{,*)}}\texttt{,}\\
&\quad\texttt{not=not}_1\texttt{(-, -, nms(-,shaking}_{\texttt{c}}\texttt{,furrowed}_{\texttt{c}}\texttt{,-))]}\\
&\backslash\\
&\texttt{s}_{\texttt{u}}\texttt{: [agent=X, theme=Y, action=Z]}
\end{aligned}
$$

This takes the category $\texttt{s}_\texttt{u}$ on its left and changes its semantic form. More specifically, it inserts the gloss $\texttt{not}_1$ at the end and adds the non-manual signal 'eyebrows furrowed' to the three glosses: `theme`, `action`, and `not`. Here the subscript 'c' ($\texttt{furrowed}_\texttt{c}$) indicates that this signal continues over the three glosses. Headshake also appears and continues over the last two glosses in a similar way. Note that in this category we use the asterisk in a different manner; it appears in the subscript of a variable. In this case, it means "keep its value unchanged". For example, $\texttt{Y}_{\texttt{nms(*,*,furrowed,*)}}$ indicates that, regardless of the value that is unified with the variable `Y`, it changes the non-manual signal part such that its channel for eyebrows has the value 'furrowed' with other channels unchanged, which is denoted by '`*`'.

Figure 12 shows the derivation process for *I don't like him*, which involves the category of *not*. It is shown that the resulting representation reflects well the coordination of non-manual signals illustrated in Figure 6.

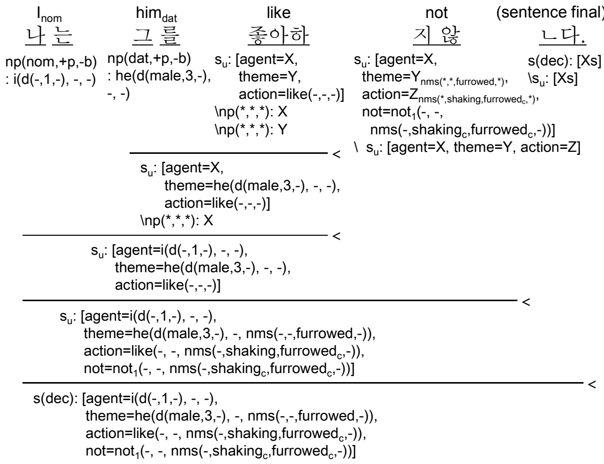

Figure 12. The derivation process for *I don't like him*

Conditional sentence. The non-manual signals 'eyebrows raised' and 'head bowed' appear while signing the verb in a conditional sentence. This phenomenon is explained by the following category for *if*.

```
so(cond): [Xs1action=*(*, *, nms(*,bowed,raised,*)), Xs2]
/su: [Xs2]
\su: [Xs1]
```

This category works in a similar way to that of *not*. It takes a clause and adds the two non-manual signals to the sign semantics of its verb (action) with the other channels unchanged. Figure 13 illustrates how the representation of the if-clause *if John flatters her* is derived.

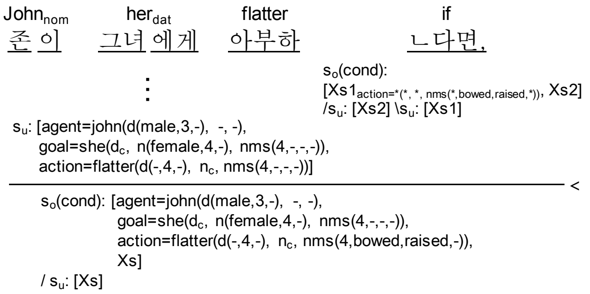

Figure 13. The derivation process for *if John flatters her*

Coordination. Two lexical items in a coordinate structure can be combined by *and* if their syntactic categories are of a similar form, i.e. if they are either identical or at least unifiable. If their semantic forms have common elements, these elements are unified together into a single element. The other elements that are not common are listed in order in the resulting semantic form. Figure 14 shows the coordination of the simple verb 'LIKE' and the agreement verb 'HELP'. It is shown in the result that their agent and theme are unified but their action appears as it is, i.e. as two consecutive verbs.

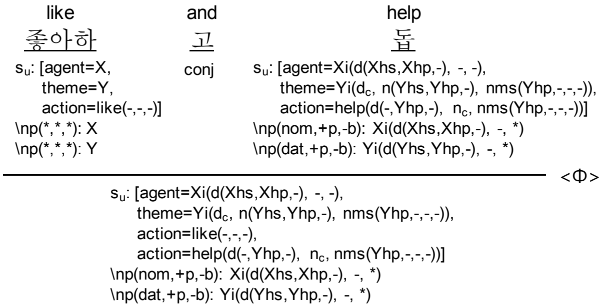

Figure 14. The derivation process for *like and help*

All together. The sentence *if John flatters her and gives money to her, I will not help him* (Figure 5) includes all of the issues so far. The syntactic and full derivations for this sentence are shown in Figures 19 and 20, respectively. All categories shown in this example are adopted from those of the small examples above. The final representation (Figure 15) is also equivalently illustrated as a diagram like Figure 21. It is easier to see in the diagram how values on each channel are changed and coordinated during derivation.

```
s(dec): [
  agent=john(d(male,3,-), -, -)
  goal=she(dc, n(male,4,-), nms(4,-,-,-))
  action=flatter(d(-,4,-), nc, nms(4c,bowed,raised,-))
  theme=money(d(money,3,-), nc, nms(3,-,-,oh))
  action=give(d(money,3,4), nc, nms(4,bowed,raised,oh))
  agent=i(d(-,1,-), -, -)
  theme=he(dc, n(male,3,-), nms(3,-,furrowed,-))
  action=help(d(-,3,-), nc, nms(3c,shaking,furrowedc,-))
  not=not1(-, -, nms(-,shakingc,furrowedc,-))
]
```

Figure 15. The final representation for *if John flatters her and gives money to her, I will not help him*

Verb valency. Although not illustrated above, another important issue is that there are cases where Korean and KSL words with the same meaning differ in the valency of verbs such that the number of arguments taken by these verbs may not be the same. For example, the intransitive verb *blow* in Korean (as well as in English) takes one argument (subject) in the sentence such as *a wind blows.* However, in KSL this sentence is expressed by a single word 'BLOW' alone since it already incorporates the performance of the subject 'WIND' (in fact, 'BLOW' is equivalent to 'WIND' in KSL). Moreover, while adjectives and adverbs are explicitly used to qualify nouns and verbs, respectively, KSL does this by just changing the speed and intensity of their performance instead of introducing additional words. So both of the sentences *a strong wind blows* and *a wind blows hard* are expressed equivalently as the single KSL word 'BLOW' with its 'strong' performance. There should thus be categories to reflect these differences properly. The following categories for the verb *blow*, adjective *strong*, and adverb *hard* can be used for the correct derivation.

blow := `su:[action=blowU(-, -, -)]`
`\np(nom,-p,+b):windU(-, -, -)`

strong := `np(Feat):X#/np(Feat):X`

hard := `(s:[Xsaction=*#]\np(Feat):Y)`
`\(s:[Xs]\np(Feat):Y)`

It is shown that the semantic form of s_u in the category of *blow* contains the gloss for *blow* only, but not the gloss for *wind*. This means that this category takes the subject argument *wind* on the left, but does not produce its gloss in the resulting representation. So in this case, the argument *wind* is actually considered redundant for generating the output representation.

The categories of *strong* and *hard* work in a similar way. Their role is to take a category for a noun or a verb as an argument and to qualify it by adding intensity information. We use the subscript '#' on the right of a gloss name in the semantic form to indicate its intensity information. As shown above, the category of *strong* takes a noun phrase on its right and attaches '#' to the gloss in the semantic form ($X_\#$), with other information unchanged. In a similar way, the category of *hard* takes the functor category `s\np` as argument and attaches '#' to the gloss for action in the semantic form of `s` so that the resulting category can generate a signed representation with an 'intensified' verb (or action).

Figures 16 and 17 illustrate the derivation processes for *a strong wind blows* and *a wind blows hard*. Figure 16 shows that *strong* and *wind* combine first into the KSL noun 'WIND' with strong performance (denoted by the subscript '#'). It then combines with *blow* on its right and forms 'BLOW'. During this process, the subscript '#' of 'WIND' is passed to that of 'BLOW' by unification (using common subscript variable `U`). It means that the intensity information of 'BLOW' comes from the category of *strong*. The derivation process eventually generates the semantic representation for the single KSL word 'BLOW' with its intensity information.

The derivation process for *a wind blows hard* in Figure 17 is slightly different. The category of *hard* combines with that of *blow* first and turns it into the category with intensity information added to its gloss for 'BLOW'. This combined category then combines with that of *wind* again, resulting in 'BLOW' with intensity information, which is the same as the one obtained by the process in Figure 16.

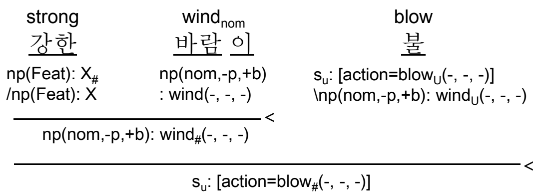

Figure 16. The derivation process for *a strong wind blows*

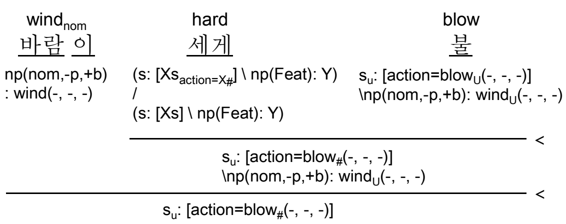

Figure 17. The derivation process for *a wind blows hard*

5. DISCUSSION

5.1 Advantages and Limitations

The most noticeable difference between our approach and others would be that while most other approaches go through several separate steps for translation and use different formalisms or techniques for each step, our proposed method generates directly a semantic representation for sign language expressions by a single step derivation, which is one of the strengths of the CCG framework. Thus we do not need to manage separate lexicons and rules for several different stages. Moreover, since most information needed for parsing a sentence and resolving linguistic phenomena is encoded in the CCG sign lexicon, it is easier to check for the role of each lexical item for derivation and to see how signals in a single sign are propagated to and coordinated with other signs. It is thus possible to manage the derivation process and to deal with additional phenomena, by just changing items in the CCG sign lexicon. For example, if the non-manual signal 'eyebrows furrowed' turns out not to appear in the negative 'NOT' and should thus be changed immediately, all we need to do is just to make a small change to its semantic form, as shown in Figure 18.

Our method can also represent coordination among multi-channel signals, especially for the signals produced continuously over multiple signs, such as non-manual signals in Figure 6. We show that the output representation can be illustrated as a diagram as in Figures 3 and 21, which makes it much easier to see the coordination among channels over time.

```
su: [agent=X,
     theme=Y_nms(*,*,furrowed,*),
     action=Z_nms(*,shaking,furrowed_c,*),
     not=not_1(-, -, nms(-,shaking_c,furrowed_c,-))]
\ su: [agent=X, theme=Y, action=Z]

                    ⇩

su: [agent=X,
     theme=Y_nms(*,*,furrowed,*),
     action=Z_nms(*,shaking,furrowed_c,*),
     not=not_1(-, -, nms(-,shaking_c,-,-))]
\ su: [agent=X, theme=Y, action=Z]
```

Figure 18. Modification to the semantic form

Our representation still has some limitations. First, the synchronization of signals is done at a level of individual signs only. So the present model cannot explain well how signals change during a single sign; for example, eye gaze moves from the initial position to the final position during a single spatial verb. One solution would be to denote signals in a finer grained way, for example "eye gaze on the dominant hand", but this is still a challenging issue to be studied further in future work.

It is also necessary to include information about timing, speed, and pause of signals that are linguistically meaningful because it plays a significant role in communication among deaf users [17]. Similarly, the change of signal intensity over time is another important issue to be resolved [18]. We leave these issues for future work, together with other remaining questions.

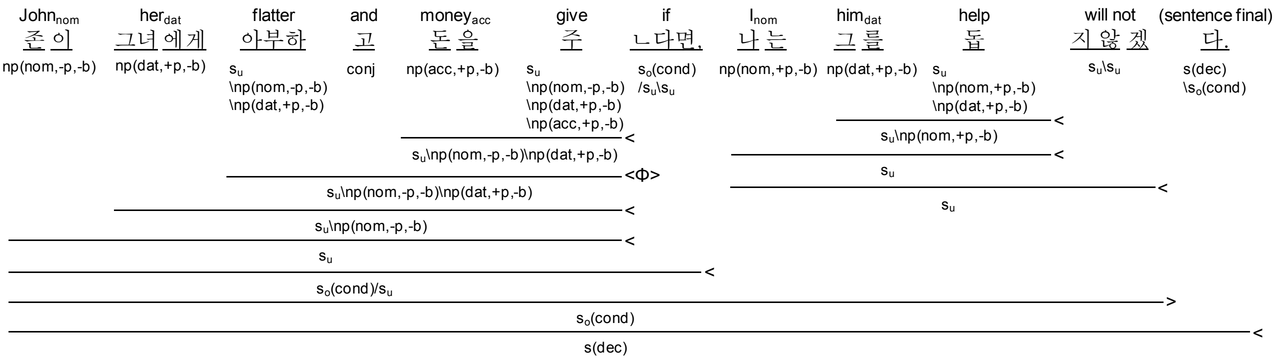

Figure 19. The process of syntactic derivation for *if John flatters her and gives money to her, I will not help him*

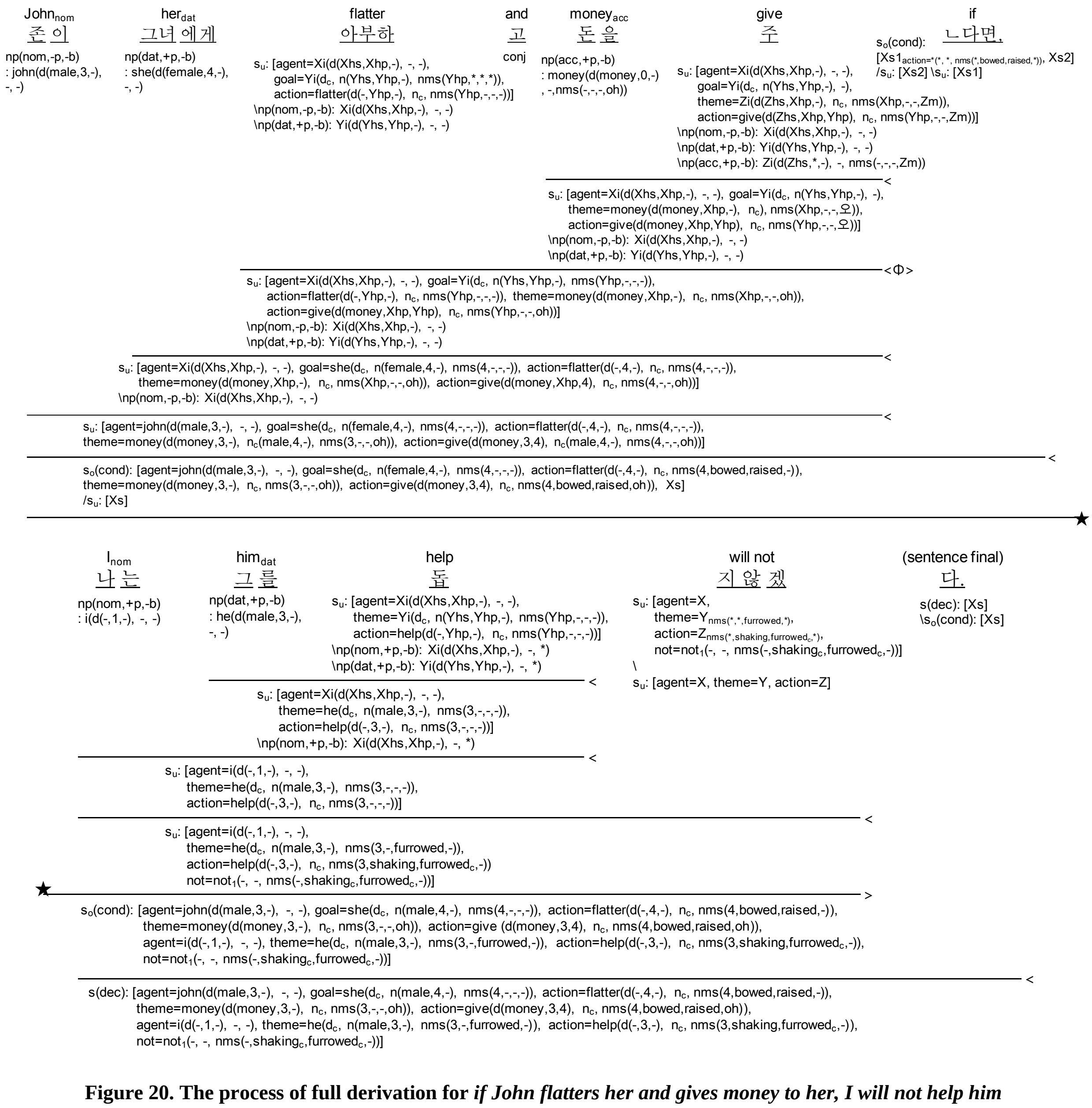

Figure 20. The process of full derivation for *if John flatters her and gives money to her, I will not help him*

	(dominant hand)			(non-dominant hand)			(non-manual component)			
Gloss name	hs	hp	hm	hs	hp	hm	eg	h	eb	m
John	male	3	-	-	-	-	-	-	-	-
she				female	4	-	4	-	-	-
flatter	-	4	-					bowed	raised	-
money	male	3	-				3	-	-	oh
give	money	3	4				4	bowed	raised	
I	-	1	-	-	-	-	-	-	-	-
he				male	3	-	3	-	furrowed	-
help	-	3	-					shaking		-
not_1	-	-	-	-	-	-	-			-

time

Figure 21. A gloss diagram for *if John flatters her and gives money to her, I will not help him*

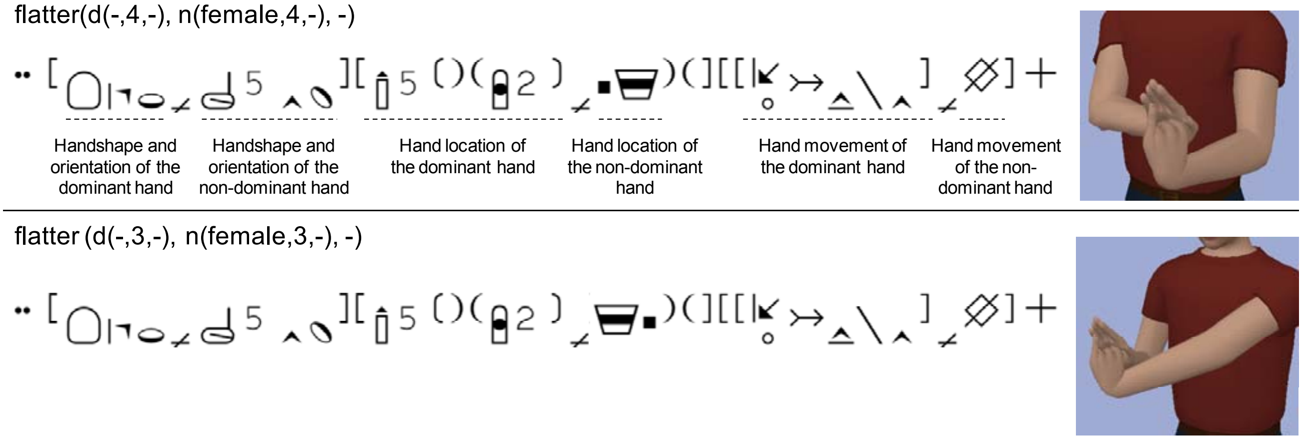

Figure 22. The HamNoSys representations for 'FLATTER' with different hand positions of the non-dominant hand

5.2 Mapping to Notation

The focus of this paper has not been on translating the source text into a particular notation system, but there is a good possibility of converting our parsing result into the HamNoSys representation [9]. Since our representation consists of signs parameterized by multiple phonemes of sign language, and a HamNoSys string also describes individual signing gestures at a phonetic level, it would be fairly straightforward to find some mapping between them.

We use a method similar to the one proposed in the ViSiCAST project [19, 20]. A HamNoSys string is prepared for each gloss in the CCG sign lexicon, which is parameterized and instantiated by default. Whenever the representation for a lexical item specifies an explicit value on a particular channel, the corresponding part in the HamNoSys string is instantiated by an appropriate value accordingly. Figure 22 shows two HamNoSys strings for the KSL verb 'FLATTER' with only their hand positions instantiated differently. They are indicated by ▪⊟ and ⊟▪, respectively. The pictures on the right are the result of synthesizing animation using the SiGMLSigning software [19] with the two HamNoSys strings on the left. The semantic representation can thus be converted into a sequence of the HamNoSys strings for each sign in this way. However, the major problem in such approach is the difficulty of representing the coordination and continuity of signals. In particular, since the expressive power of the current version of HamNoSys with respect to non-manual signals is still limited, it is not easy to represent the signals produced over multiple signs simultaneously, such as the ones in Figure 6.

6. CONCLUSION

In this paper, we described a grammar-based method to translate written text into a semantic representation for sign language expressions. A CCG framework is employed to analyze sentences and address linguistic phenomena. We showed that the lexicon creation is an important part for correct translation. The resulting representation can be directly interpreted as a sequence of signs with linguistic channels coordinated over multiple signs. An approach to convert the result into the HamNoSys strings is also presented, but there are some limitations to overcome so that the mapping between them can be made adequately.

In order to enable high quality translation, we believe that more linguistic phenomena must be reflected correctly on the CCG sign lexicon. The problem, however, is that the linguistic characteristics of KSL are not yet much studied enough to provide reliable references, compared to other sign languages such as ASL and BSL. We have thus provided only a limited number of examples in this paper, waiting for further advances in the study of KSL.

Although we have only addressed the translation of Korean into KSL, our proposed method should essentially be languageindependent, since the computational modeling of written sentences with CCG has been widely studied for many languages, and different sign languages can be explained by similar phonology and phonetics. It is thus expected that CCG-based generation of sign language representations of the kind advanced in this paper would be also applicable to other pairs of spoken and sign languages.

7. ACKNOWLEDGMENTS

We thank the anonymous reviewers for valuable feedback. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (MEST) (No. 2011-0004422). This work was also supported by the NRF grant funded by MEST (No. 2011-0018262).

8. REFERENCES

[1] Huenerfauth, M. and Hanson, V. L. 2009. Sign language in the interface: access for deaf signers. *The Universal Access Handbook*. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

[2] Safar, E. and Marshall, I. 2002. Sign language translation via DRT and HPSG. 3rd International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2002).

[3] Marshall, I. and Safar, E. 2005. Grammar development for sign language avatar-based synthesis. 3rd International Conference on Universal Access in Human-Computer Interaction (UAHCI 2005).

[4] Efthimiou, E., Fotinea, S. -E., and Sapountzaki, G. 2007. Feature-based natural language processing for GSL synthesis. *Sign Language & Linguistics*, Vol. 10, No. 1, pages 3-23.

[5] Chiu, Y. -H., Wu, C. -H., Su, H. -Y., and Cheng, C. -J. 2007. Joint optimization of word alignment and epenthesis generation for Chinese to Taiwanese sign synthesis. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 1 (January 2007), 28-39. DOI=http://dx.doi.org/10.1109/TPAMI.2007.15

[6] Morrissey, S., Way, A., Stein, D., Bungeroth, J., and Ney, H. 2007. Combining data-driven MT systems for improved sign language translation. *Machine Translation* Summit XI, 10-14 September, 2007.

[7] Lu, P. and Huenerfauth, M. 2010. Collecting a motion-capture corpus of American Sign Language for data-driven generation research. NAACL HLT 2010 Workshop on Speech and Language Processing for Assistive Technologies (SLPAT'10).

[8] Almohimeed, A., Wald, M., and Damper, R. I. 2011. Arabic text to Arabic Sign Language translation system for the deaf and hearing-impaired community, 2nd Workshop on Speech and Language Processing for Assistive Technologies (SLPAT'11).

[9] Hanke, T. 2004. HamNoSys – Representing sign Language data in language resources and language processing contexts. Workshop on the occasion of the 4th International Conference on Language Resources and Evaluation (LREC 2004).

[10] Steedman, M. 1996. *Surface structure and interpretation*. The MIT Press.

[11] Steedman, M. 2000. *The syntactic process*. The MIT Press.

[12] Park, J. C. and Cho, H. 2000. Informed parsing for coordination with Combinatory Categorial Grammar. 18th International Conference on Computational Linguistics (COLING 2000).

[13] Cha, J., Lee, G. and Lee, J. 2002. Korean Combinatory Categorial Grammar and statistical parsing. *Computers and the Humanities.* Vol. 36, pages 431-453.

[14] Bos, J., Clark, S., Steedman, M., Curran, J. R., and Hockenmaier, J. 2004. Wide-coverage semantic representations from a CCG parser. 20th International Conference on Computational Linguistics (COLING 2004).

[15] Çakici, R. 2005. Automatic induction of a CCG grammar for Turkish. ACL Student Research Workshop.

[16] Hockenmaier, J. 2006. Creating a CCGbank and a wide-coverage CCG lexicon for German. Joint Conference of the International Committee on Computational Linguistics and the Association for Computational Linguistics (COLING/ACL 2006).

[17] Huenerfauth, M. 2009. A linguistically motivated model for speed and pausing in animations of American Sign Language. *ACM Trans. Access. Comput.* 2, 2, Article 9 (June 2009). DOI= http://doi.acm.org/10.1145/1530064.1530067

[18] Huenerfauth, M. 2005. Representing coordination and non-coordination in American Sign Language animations. 7th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS'05).

[19] Kennaway, J. R., Glauert, J. R. W., and Zwitserlood, I. 2007. Providing signed content on the Internet by synthesized animation. *ACM Trans. Comput.-Hum. Interact.* 14, 3, Article 15 (September 2007). DOI= http://doi.acm.org/10.1145/1279700.1279705

[20] Elliott, R., Bueno, J., Kennaway, R., and Glauert, J. 2010. Towards the integration of synthetic SL animation with avatars into corpus annotation tools. 7th International Conference on Language Resources and Evaluation (LREC 2010).