Czech Text-to-Sign Speech Synthesizer*

Zdeněk Krňoul, Jakub Kanis, Miloš Železný, and Luděk Müller

Univ. of West Bohemia, Faculty of Applied Sciences, Dept. of Cybernetics
Univerzitní 8, 306 14 Pilsen, Czech Republic
{zdkrnoul,jkanis,zelezny,muller}@kky.zcu.cz

Abstract. Recent research progress in developing of the Czech – Sign Speech synthesizer is presented. The current goal is to improve the system for automatic synthesis to produce accurate synthesis of the Sign Speech. The synthesis system converts written text to an animation of an artificial human model (avatar). This includes translation of text to sign phrases and their conversion to the animation of the avatar. The animation is composed of movements and deformations of segments of hands, a head and also a face. The system has been evaluated by two initial perceptual tests. The perceptual tests indicate that the designed synthesis system is capable to produce the intelligible Sign Speech.

Keywords: Sign speech, automatic synthesis, machine translation.

1 Introduction

In the scope of this paper, we use the term Sign Speech (SS) for both the Czech Sign Language (CSE) and Signed Czech (SC). The CSE is a natural and adequate communication form and a primary communication tool of the hearing-impaired people in the Czech Republic. It is composed of the specific visual-spatial resources, i.e. hand shapes (manual signals), movements, facial expressions, head and upper part of the body positions (non-manual signals). It is not derived from or based on any spoken language. CSE has basic language attributes, i.e. system of signs, double articulation, peculiarity and historical dimension, and has its own lexical and grammatical structure. On the other hand, the SC was introduced as an artificial language system derived from the spoken Czech language to facilitate communication between deaf and hearing people. SC uses grammatical and lexical resources of the Czech language. The Czech sentence is audibly or inaudibly articulated during the SC production and the CSE signs of all individual words of the sentence are simultaneously signed with the articulation.

The use of written language instead of spoken one is a wrong idea in the case of the Deaf. Hence, the Deaf have problems with understanding the majority language (the language used by hearing people) when they are reading a written text. The majority language is the second language of the Deaf and its use by

* Support for this work was provided by the Grant Agency of Academy of Sciences of the Czech Republic, project No. 1ET101470416 and MŠMT LC536.

A. Popescu-Belis, S. Renals, and H. Bourlard (Eds.): MLMI 2007, LNCS 4892, pp. 180–191, 2007.

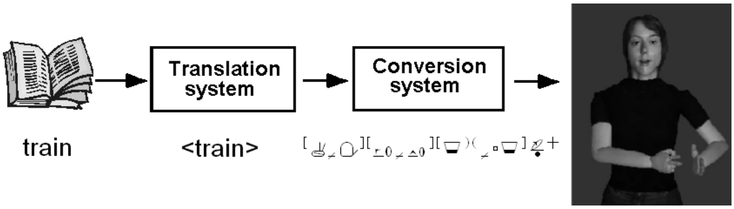


Fig. 1. The schema of the Sign Speech synthesis system

the deaf community is only particular. Thus, the majority language translation into the Sign Speech is highly important for better orientation of the Deaf in the majority language-speaking world. Currently, human interpreters provide this translation, but their service is expensive and not always available. A full dialog system (with ASR and Text-to-Sign-Speech (TTSS) systems on one side (from spoken to sign language) and Automatic-Sign-Speech-Recognition (ASSR) and TTS systems on the other side (from sign to spoken language)) represents a solution which does not intend to fully replace the interpreters, but its aim is to help in everyday communication in selected constrained domains such as the post office, health care, traveling, etc.

Our synthesis system consists of two parts: the translation and the conversion subsystem (see Figure 1). The translation system transfers Czech written text to its textual representation in the Sign Speech (textual sign representation). The conversion system then converts this textual sign representation to an animation of an artificial human model (avatar). The resulting animation then represents the corresponding utterance in the Sign Speech.

The translation system is an automatic phrase-based translation system. A Czech sentence is divided into phrases and these are then translated into corresponding Sign Speech phrases. The translated words are reordered and rescored using a language model at the end of the translation process. In our synthesizer we use own implementation of simple monotone phrase-based decoder - SiM-PaD. This decoder and its performance will be described in more details in next section.

The problem of translated phrase conversion is composed of the isolated sign animation and their concatenation. Each sign is expressed by the manual and non-manual component. The manual component represents necessary movements, orientations and shapes of hands. The non-manual component is composed of complemented movements of the upper half-body, face gestures or face articulation (lips and inner mouth organs). The symbolic notation HamNoSys[1] (Hamburg Notation System for Sign Languages) was applied for controlling of the manual component and the no-facial upper half-body movements. The synthesis of a mouth articulation is separately supplemented by a talking head system.

[1] Available at www.sign-lang.uni-hamburg.de/projects/HamNoSys.html.

2 Translation System

The machine translation model is based on the noisy channel model. When we apply the Bayes rule on the translation probability $p(\mathbf{t}|\mathbf{s})$ for translating a sentence $\mathbf{s}$ in a source language into a sentence $\mathbf{t}$ in a target language we obtain:

$$\mathrm{argmax}_{\mathrm{t}} p(\mathbf{t}|\mathbf{s}) = \mathrm{argmax}_{\mathrm{t}} p(\mathbf{s}|\mathbf{t})p(\mathbf{t})$$

Thus the translation probability $p(\mathbf{t}|\mathbf{s})$ is decomposed into two separate models: a translation model $p(\mathbf{s}|\mathbf{t})$ and a language model $p(\mathbf{t})$ that can be modeled independently. In the case of phrase-based translation the source sentence $\mathbf{s}$ is segmented into a sequence of I phrases $\bar{\mathbf{s}}_1^I$ (all possible segmentations have the same probability). Each source phrase $\bar{s}_i, i = 1, 2, ..., I$ is translated into a target phrase $\bar{t}_i$ in the decoding process. This particular ith translation is modeled by a probability distribution $\phi(\bar{s}_i|\bar{t}_i)$. The target phrases can be reordered to get more precise translation. The reordering of the target phrases can be modeled by a relative distortion probability distribution $d(a_i - b_{i-1})$ as in [1], where a_i denotes the start position of the source phrase which was translated into the ith target phrase. b_{i-1} denotes the end position of the source phrase translated into the $(i-1)th$ target phrase. Also a simpler distortion model $d(a_i - b_{i-1}) = \alpha^{|a_i - b_{i-1} - 1|}$ [1], where α is a predefined constant, can be employed. The best target output sentence $\mathbf{t_{best}}$ can then be for a given source sentence $\mathbf{s}$ acquired as:

$$\mathbf{t_{best}} = \mathrm{argmax}_{\mathrm{t}} p(\mathbf{t}|\mathbf{s}) = \prod_{i=1}^{I} [\phi(\bar{s}_i|\bar{t}_i) d(a_i - b_{i-1})] p_{LM}(\mathbf{t})$$

Where $p_{LM}(\mathbf{t})$ is a language model of the target language (usually a trigram model with some smoothing, built from a huge portion of target texts).

2.1 Comparison of Decoders

We will compare SiMPaDs performance with the performance of state-of-the-art phrase-based decoder Moses in case of Czech to Signed Czech translation. And we will introduce class-based language model and post-processing method which either improve results of translation.

Decoders. The first decoder is freely available state-of-the-art factored phrase-based beam-search decoder - **MOSES** [2]. Moses can work with factored representation of words (i.e. surface form, lemma, part-of-speech, etc.) and uses a beam-search algorithm, which solves a problem of the exponential number of possible translations (due to the exponential number of possible alignments between source and target translation) for efficient decoding. The training tools for extracting of phrases from the parallel corpus are also available, i.e. the whole translation system can be constructed given a parallel corpus only. For language modeling we use the SRILM[2] toolkit.

[2] Available at http://www.speech.sri.com/projects/srilm/download.html.

The second decoder is our simple monotone phrase-based decoder - **SiMPaD**. The monotonicity means using the monotone reordering model, i.e. no phrase reordering is permitted. In the decoding process we choose only one alignment with the longest phrase coverage (for example if there are three phrases: p_1, p_2, p_3 coverage three words: w_1, w_2, w_3, where $p_1 = w_1{+}w_2, p_2 = w_3, p_3 = w_1{+}w_2{+}w_3$, we choose the alignment which contains phrase p_3 only). A standard Viterbi algorithm is used for the decoding. SiMPaD uses SRILM language models.

Data and Evaluation Criteria. The main resource for the statistical machine translation is a parallel corpus which contains parallel texts of the source and the target language. The acquisition of such a corpus in the case of SS is complicated by the absence of an official written form of both the CSE and the SC. Therefore we used a Czech to Signed Czech (CSC) parallel corpus [3] for training of decoders.

The CSC corpus contains 1130 dialogs from a telephone communication between customer and operator in a train timetable information center. The parallel corpus was created by semantic annotation of several hundred dialogs and by adding the SC translation of all the dialogs. A SC sentence is written as a sequence of CSE signs. The whole CSC corpus contains 16 066 parallel sentences, 110 033 running words and 109 572 running signs, 4082 unique words and 720 unique signs. Each sentence of the CSC corpus has assigned the written form of the SC translation, a type of the dialog act, and its semantic meaning in a form of semantic annotation. For example (we use English literal translation) for the Czech sentence: *good day I want to know how me it is going in Saturday morning to brno* we have the SC translation: *good_day I want know how _ _ go in Saturday morning to brno* and for the part: *good day* the dialog act: *conversational_domain="frame" + speech_act="opening"* and the semantic annotation: *semantics="GREETING"* . The dialog act: *conversational_domain="task" + speech_act="request_info* and the semantic annotation: *semantics="DEPARTURE(TIME, TO(STATION))"* is assigned to the rest of the sentence. The corpus contains also a handcrafted word alignment (added by annotators during the corpus creation) of every Czech – SC sentence pair. For more details about the CSC corpus see [3].

We use the following criteria for evaluation. The first criterion is **Sentence Error Rate** (**SER**): It is a ratio of the number of incorrect sentence translations to the number of all translated sentences. The second criterion is **Word Error Rate** (**WER**): This criterion is adopted from ASR area and is defined as the Levenshtein edit distance between the produced translation and the reference translation in percentage (a ratio of the number of all deleted, substituted and inserted produced words to the total number of reference words). The third criterion is **Position-independent Word Error Rate** (**PER**): it is simply a ratio of the number of incorrect translated words to the total number of reference words (independent of the word order). The last criterion is **BLEU** score [4]: it counts a modified n-gram precision for the output translation with respect to the reference translation. A lower value of the first three criteria and a higher value of the last one indicate better i.e. more precise translation.

Experiment. Both decoders are trained on the CSC corpus. SiMPaD uses a phrase table of handcrafted phrases (acquired from the handcrafted alignment of the CSC corpus; a phrase translation probability is estimated by the relative frequency [1]) and phrase-based language model (a basic unit of the language model is a phrase instead of a word). Moses uses the phrase table of automatically acquired phrases and the standard language model. The phrases were acquired from Giza++ word alignment of the parallel corpus (the word alignment established by Giza++ toolkit[3]) by some heuristics (we used default heuristics). There are many parameters which can be specified in the training and decoding process of Moses. Unless otherwise stated, we used the default values of parameters (for more details see Moses' documentation in [2]).

To improve results of translation we used two enhancements - a class-based language model and post-processing method. As well as in the area of ASR, there are problems with out-of-vocabulary words (OOV) in the automatic translation area. We can translate only words which are in a translation vocabulary (we know their translation into the target language). By the analysis of the translation results we found that many OOV words are caused by missing a station or a personal name. Because the translation is limited to the domain of dialogs in the train timetable information center, we decided to solve the problem of OOV words similarly as in work [5], where the class-based language model was used for the real-time closed-captioning system of TV ice-hockey commentaries. The classes of player's names, nationalities and states were added into the standard language model in this work. Similarly, we added two classes into our language model - the class for all known station names: STATION and the class for all known personal names: PERSON. Because the semantic annotation of the corpus contains station and personal names, we can simply replace these names by relevant class in training and test data and collect a vocabulary of all station names for their translation (the personal names are always spelled).

The post-processing method includes two steps. Firstly, we can remove the words which are omitted in the translation process (they are translated into 'no translation' sign respectively) from the resulting translation. In any case, to keep these words in training data gives better results (more detailed translation and language models). Secondly, we can substitute OOV words by a finger-spelling sign, because the unknown words are finger spelled in the SC usually. The results of a decoder comparison are in Table 1. The results of SiMPaD and Moses decoder with the phrase-based and the standard trigram language model (suffix _LM(P)) are in the first and third column. The results of decoders with the class-based language model and the post-processing method (suffix _CLM(P)_PP) are in the second and forth column.

We compared the SiMPaD's results with the state-of-the-art phrase-based decoder Moses. We found that the SiMPaD's results are fully comparable with Moses's results while SiMPaD is almost 5 times faster than the Moses decoder. Hence, the SiMPaD is convenient for the real time translation, which is sufficient for the TTSL system. We introduced the class-based language model and the

[3] Available at http://www.isi.edu/∼och/GIZA++.html.

Table 1. The results of SiMPaD and Moses decoder in Czech $\Longrightarrow$ Signed Czech translation

	SiMPaD_LMP	SiMPaD_CLMP_PP	Moses_LM	Moses_CLM_PP
SER[%]	44.84 ± 1.96	**40.59 ± 2.06**	45.30 ± 2.40	**41.97 ± 2.20**
BLEU	67.92 ± 1.93	**73.43 ± 1.78**	68.77 ± 1.72	**73.64 ± 1.84**
WER[%]	16.02 ± 1.08	**14.23 ± 1.06**	16.37 ± 1.02	**14.73 ± 1.16**
PER[%]	13.30 ± 0.91	**9.65 ± 0.78**	11.22 ± 0.82	**8.67 ± 0.73**

post-processing method which improved the translation results from about 8.1 % (BLEU) to about 27.4 % (PER) of a relative improvement in the case of SiMPaD decoder and from about 7.1 % (BLEU) to about 22.7 % (PER) of a relative improvement in the case of Moses decoder (the relative improvement is measured between the word-based model - _LM(P) and the class-based model with the post-processing - _CLM(P)_PP).

3 Conversion System

The conversion system is based on HamNoSys 3.0 notation. The notation is deterministic and suitable for the processing of the Sign Speech in a computer system. The methodology of the notation allows precise and also extensible expression of the sign. However an editor should be used for faster composition of correct selection of symbols to final string. Symbolic strings of notated signs are stored in a vocabulary.

3.1 Analysis of Symbols and Trajectories for Isolated Sign

The synthesis process is based on key frames and trajectories. Firstly, our synthesis system automatically carries out the analysis of symbolic string and generates a tree structure composed from key frames. Then, the feature and animation trajectories are created by the trajectory processing (see Figure 2).

The structure of the HamNoSys notation can be described by a context-free grammar. The grammar is defined by four symbol groups $G = (V_t; V_n; P; S)$. The inventory of terminal symbols V_t, non terminal symbols V_n, the set of parsing

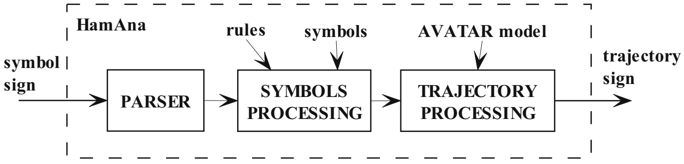


Fig. 2. The schema of our conversion system HamAna. The sign in symbolic meaning is transferred to parse tree. The symbol processing puts together the information from each symbol and the trajectory processing generates the animation trajectories.

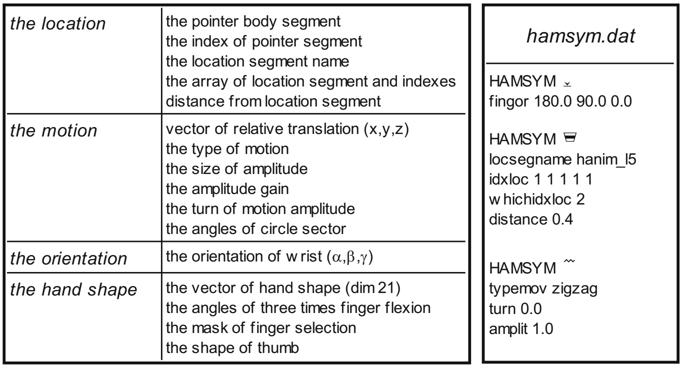
the rule	*the description*
T (001) ∂	the symbol for hand shape ∂ is generalized on the symbol **T**
HT1 (026) T MT	the generalized hand shape **T** is modified by symbol **MT**
VHO (006) ⌜ HO2 ⌿ HO2 ⌝	the processing of orientation of left and right palm separately
HAMNOSYS (202) HH1	one of output rules, **HAMNOSYS** is the starting symbol

Fig. 3. The example of grammatical rules

rules P and starting symbol S were collected. The V_t symbols were directly derived from the HamNoSys symbols, the auxiliary symbols V_n were chosen according to the generalization of a HamNoSys notation structure and in the relation to the parsing rules P. The parsing rules were created to cover all HamNoSys notation variants. Example of 4 from all the 361 rules is in Figure 3. The form of the rule is following: one left non terminal symbol, a number of action and a right side of the rule (for example $HT1(026) \rightarrow T \quad MT$, where $HT1$ is left non terminal symbol, (026) is number of rule action and $T \quad MT$ is the right side). The right side is given by one or combination of V_t and V_n symbols. The number indicates the action joined with the rule. There are 28 rule actions for the symbol processing and 11 rule actions for the following trajectory processing. We implemented Earley's algorithm for the syntactic analysis. The structurally correct notations are then represented by a tree structure.

For accepted symbolic string, we have the parse tree and also the path to each leaf node. The path from a root of the tree to the leaf node of the particular symbol is given by the sequence of rule actions. The processing of nodes is carried out by several tree walks whilst the size of the tree is reduced. Each node is described by two key frames to separate dominant and non dominant hand. The structure of key frame is composed from specially designed items. These items are read for all leaf nodes from a symbol definition file. The list of all items and the example of definition file is depicted in Figure 4. Currently, the definition file covers 138 HamNoSys symbols. In the next processing, the key frames of the remaining nodes are joined and blended according to the rule actions.

The feature trajectory is created from key frames as the time sequence of feature frames in particular leaf nodes of the tree. The structure of a feature frame is derived from the key frame structure and contains only items for the static position of hand in space, the orientation of wrist and hand shape. The feature trajectories are created by next tree walk. Due to the notation of a superposition of notated movements, the relevant subtrees have to be marked by parallel flags. To overcome this, only the start and end feature frame of leaf nodes have to be precomputed. These frames are computed from the geometry of the animation model. In our approach, the location of hand is implicitly given by a position of wrist join in 3D space. In the case of the precise contact of dominant hand to other segments of animation model, the location of wrist for



the location	the pointer body segment the index of pointer segment the location segment name the array of location segment and indexes distance from location segment
the motion	vector of relative translation (x,y,z) the type of motion the size of amplitude the amplitude gain the turn of motion amplitude the angles of circle sector
the orientation	the orientation of w rist (α,β,γ)
the hand shape	the vector of hand shape (dim 21) the angles of three times finger flexion the mask of finger selection the shape of thumb

hamsym.dat

```
HAMSYM
fingor 180.0 90.0 0.0

HAMSYM
locsegname hanim_l5
idxloc 1 1 1 1 1
w hichidxloc 2
distance 0.4

HAMSYM
typemov zigzag
turn 0.0
amplit 1.0
```

Fig. 4. On left is the list of all items. The relevant subsets are stored for each HamNoSys symbol. On right is the example of the stored items in the definition file: the orientation of hand, location on the body and the modifier of the direct movement.

the dominant hand is recomputed in the relation with the relevant pointer body segment. For this purpose, the algorithm computes firstly the location of non dominant arm thereafter the location of the dominant arm.

The following step of synthesis algorithm transfers the start and end frames to the relevant parent nodes. The transfer is controlled by the parallel flags and rule actions of the HamNoSys repetition modalities. In this state of the algorithm, the upper half-body movements or the some random motions of head have to be applied. The duration of trajectories is determined by number of frames. The frequency of frames is implicitly set to 25 frames per second. In order to get total duration of processed sign, the number of feature frames is inserted into leaf nodes and transferred into the root node. Here, the identical duration of trajectories in two parallel subtrees has to be taken into account.

The feature trajectories are computed for leaf nodes of the reduced tree according to type of a motion and other items in the relevant key frames. Next processing transfers these feature trajectories into the root node. The transferred trajectories are concatenated, merged, repeated or inversed according to rule actions of the trajectory processing. The trajectory for the dominant arm in root node is now complete and the trajectory for the non dominant arm is either empty or incomplete. If the symbols of a sign symmetry are notated then the trajectory for the non dominant arm is completed from the trajectory of the dominant arm or from the initial values. The final feature trajectories in the root node are transformed into the animation trajectories by an inverse kinematics technique to control the joints of animation model. The analysis of symbols allows the computation of trajectories only for hands and the upper half-body. Trajectories for the lip articulation and face gestures are produced by the "talking head" system separately.

3.2 Talking Head System

Trajectories for face gestures and also for articulation of lips, a tongue and jaws are created by visual synthesis carried out by the talking head subsystem. This visual synthesis is based on concatenation of phonetic units. Any word or phrase which represents an isolate sign in textual form, is here processed as a string of successive phones. The lip articulatory trajectories are concatenated by the visual unit selection method [6]. This synthesis method uses an inventory of phonetic units and the regression tree technique. It allows precise coverage of coarticulation effects. In the inventory of units, several realizations of phoneme are stored. Our synthesis method assumes that the lip and tongue shape is described by a linear model. The realization of a phoneme is described by 3 linear components for lip shape and 6 components for a tongue shape. The lip components represent a lip opening, a protrusion and an upper lip raise. The tongue components consist of a jaw height, a dorsum raise, a tongue body raise, a tip raise, a tip advance and a tongue width. The synthesis algorithm performs a selection of an appropriate phoneme candidate according to the phoneme context information. This information is built from a triphone context, the occurrence of a coarticulation resistant component (of the lip or the tongue) in adjacent phonemes and also from the time duration of neighboring speech segments. Final trajectories are computed by a cubic spline interpolation between the selected phoneme realizations.

These facial trajectories should be time-aligned with the timing of acoustic form of the relevant sign. This form is produced by an appropriate TTS system. The synthesis of face gesture trajectories is based on the concatenation and the combination of the neutral face expression and one of the 6 basic face gestures: happiness, anger, surprise, fear, sadness and disgust.

3.3 Synchrony of Facial Trajectories and Continuous Sign Speech Synthesis

The synchrony of the manual and non-manual components is crucial in the synthesis of continuous Sign Speech and ensures overall intelligibility. An asynchrony is caused by the different speech rate of the spoken speech and the Sign Speech. We designed an effective solution for Signed Czech. The synchrony method combines the basic concatenation technique with the time delay processing at the level of words. Firstly, for each isolated sign the animation trajectories from the analysis of symbols described in 3.1 and trajectories from the talking head system described in 3.2 are generated. The time delay processing determines the duration of both trajectories and selects the longest variant. The following step of processing evaluates an interpolation time for the concatenation of adjacent isolated signs in the synthesized utterance with regard to the longest variant. Thus added interpolation time ensures the fluent shift of a body pose. We select the simple linear interpolation of the frames on the boundaries of concatenated signs. The interpolation of a hand shape and its 3D position is determined by weight average, the finger direction and palm orientation is interpolated by the extension to the quaternion.

3.4 Animation Model

Our animation algorithm employs a 3D geometric animation model of the avatar in compliance with the H-Anim standard[4]. Our model is composed of 38 joints and body segments. These segments are represented by textured triangular surfaces. The setting of the correct shoulder and elbow rotations from a position of the wrist is solved by the inverse kinematics[5]. There are 7 degrees of a freedom for each limb. The settings of the remaining joints and local deformation of the triangular surfaces allows the animation of an arbitrary avatar pose. The local deformation of triangular surfaces is primarily used for the animation of the face and the tongue model. The surfaces are deformed according to animation schema based on the definition of several control points and splines functions [7]. The rendering of the animation model is implemented in C++ code and OpenGL. The animation model is shown in Figure 5.

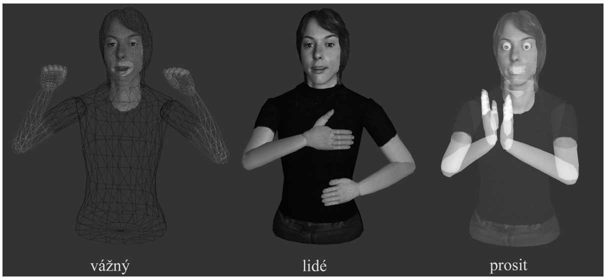


Fig. 5. The animation model for three signs: wire-frame topology, textured and blended rendering

4 Perceptual Evaluation

Two initial tests on an intelligibility of synthesized Sign Speech have been performed. The goal has been to evaluate a quality of our Sign Speech synthesizer. Two participants who are experts in the Sign Speech served as judges. We used the vocabulary of about 130 signs selected from the CSC corpus. We completed several video records of the avatar animation and also of the signing person. The video records of the signing person were taken from the electronic vocabulary [8]. The capturing of video records of our animation was prepared under two conditions.

4.1 Isolated Signs

The equivalence test was aimed at the comparison of animation movements of isolated signs with the movements of the signing person. Video records of 20

[4] Available at www.h-anim.org.

[5] Available at cg.cis.upenn.edu/hms/software/ikan/ikan.html.

pairs of randomly selected isolated signs were completed. The frontal view of the avatar model and the signing person was used in this test. The participants evaluated this equivalence by marks from 1 to 5. The meaning of the marks was:

- 1 totally perfect; the animation movements are equivalent to the signing person
- 2 the movements are good, the location of the hand, shapes or speed of the sign are a little different but the sign is intelligible
- 3 the sign is difficult to recognize; the animation includes mistakes
- 4 incorrectly animated movements
- 5 totally incorrect; it is a different sign

The results are shown in the left panel of Figure 6. The average mark of participant 1 is 2.25 and of participant 2 is 1.9. The average intelligibility is 70% (marks 1 and 2 indicate an intelligible sign). There was 65% mark agreement between participants. The analysis of signs with lower marks shows that the majority of mistakes are caused by the symbolic notation rather than inaccuracy in the conversion system. Thus, it is highly important to obtain as accurate symbolic notation of isolated signs as possible.

4.2 Continuous Speech

We created 20 video animation records of short utterances. The view of the avatar animation here was partially from the side. The participants judged the whole Sign Speech utterance. Subtitles (text representation of each sign) were added to the video records. Thus, the participants knew the meaning of the utterance and determined the overall intelligibility. The participants evaluate the intelligibility by marks from 1 to 5. The meaning of marks was:

- 1 the animation shows the signs from subtitles
- 2 the well intelligible utterance
- 3 the badly intelligible utterance
- 4 the almost unintelligible utterance
- 5 the totally unintelligible utterance

The results are shown in the right panel of Figure 6. All the utterances were evaluated by mark 1 or 2. On average, the animation of 70 % utterances shows

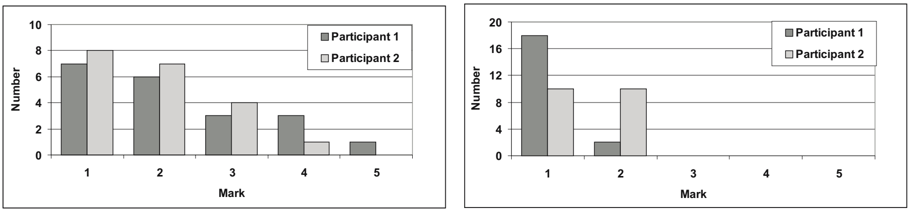


Fig. 6. Perceptual evaluation, left: isolated signs, right: continuous Sign Speech

the signs from subtitles. The results indicate that the synthesis of continuous speech is intelligible. The concatenation and synchrony method of isolated signs is sufficient.

5 Conclusion

The translation and conversion subsystems were introduced. The translation system is the phrase-based translation system which uses some heuristics (the monotone reordering and the longest phrase coverage) to speed up the translation process. The conversion system is based on the HamNoSys symbolic notation, which is capable to express the space configuration of each sign. The method of conversion the symbolic notation of sign to appropriate animation was presented.

The perceptual tests reveal that the synchrony on the level of word preserves the intelligibility for continuous Sign Speech. However the intelligibility of isolated signs highly depends on symbolic notation of particular signs in the vocabulary. Thus, it is necessary to concentrate on the acquisition of precise symbolic notation of isolated signs in future work.

References

1. Koehn, P., et al.: Statistical Phrase-Based Translation. In: HLT/NAACL (2003)
2. Koehn, P., et al.: Moses: Open Source Toolkit for Statistical Machine Translation. In: Annual Meeting of the Association for Computational Linguistics (ACL), demonstration session, Prague, Czech Republic (June 2007)
3. Kanis, J., et al.: Czech-Sign Speech Corpus for Semantic Based Machine Translation. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2006. LNCS (LNAI), vol. 4188, pp. 613–620. Springer, Heidelberg (2006)
4. Papineni, K.A., et al.: Bleu: A method for automatic evaluation of machine translation, Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center (2001)
5. Hoidekr, J., et al.: Benefit of a class-based language model for real-time closed-captioning of TV ice-hockey commentaries. In: Proceedings of LREC 2006. Paris: ELRA, pp. 2064–2067 (2006), ISBN 2-9517408-2-4
6. Krňoul, Z., Železný, M., Müller, L., Kanis, J.: Training of Coarticulation Models using Dominance Functions and Visual Unit Selection Methods for Audio-Visual Speech Synthesis. In: Proceedings of INTERSPEECH 2006 - ICSLP, Bonn (2006)
7. Krňoul, Z., Železný, M.: Realistic Face Animation for a Czech Talking Head. In: Proceedings of 7th International Conference on TEXT, SPEECH and DIALOGUE TSD 2004, Springer, Heidelberg (2004)
8. Langer, J., et al.: Znaková zásoba českého znakového jazyka. Palacký University Olomouc (2005)