Interactive HamNoSys Notation Editor for Signed Speech Annotation

Jakub Kanis, Zdeněk Krňoul

Department of Cybernetics, Faculty of Applied Sciences,
University of West Bohemia, Pilsen, Czech Republic
{jkanis, zdkrnoul, campr, mhruz, zelezny}@kky.zcu.cz

Abstract

This paper discusses the practice with an annotation of signs of signed speech and the creation of a domain-specific lexicon. The domain-specific lexicon is primarily proposed for an automatic signed speech synthesizer. The symbolic notation system based on HamNoSys notation has been adopted as a perspective solution for this purpose. We have developed two interactive editors: SignEditor and SLAPE which allow to create and to expand the lexicon. The first one is intended for the direct insertion of notation symbols and the second one is for more intuitive notation trough a graphical interface. The sign notations in both editors can be immediately converted into the avatar animation which is shown in the 3D space. It allows annotators who have no rich experiences with symbols organization to notate signs more precise. At present, our lexicon contains more than 300 signs. This initial lexicon is targeted to the domain of information systems for train connections. Further expansion will cover new areas where the automatic signed speech synthesizer can be used.

1. Introduction

The barrier in the communication between hearing impaired and hearing people should make everyday complications. The problem is that hearing people are usually not familiar with the signed speech while deaf people with the majority language. Our research aim concerns on everyday communication systems. To cope with this problem needs combination of many knowledges from different research areas, for example, the audiovisual and the signed speech recognition (Campr et al., 2007; Campr et al., 2008), the audiovisual speech (talking head) and the signed speech synthesis (Železný et al., 2006; Krňoul et al., 2008), and the bidirectional translation between the majority and the signed speech (Kanis et al., 2006; Kanis and Müller, 2007).

The goal of an automatic signed speech synthesizer is to create an avatar which uses the signed speech as a main communication form. In order to emulate the human behavior during the signing the avatar has to express manual components (hand position, hand shape) and non-manual components (face expression, lip articulation) of the performed signs. The task of the signed speech synthesis is implemented in several steps. The source utterance has to be first translated into the corresponding sequence of signs since the signed speech has different grammar than the spoken one. Then it is necessary to concatenate the relevant isolate signs to create the continuous signed speech utterance. The non-manual components should be supplement by the talking head which is for example able to articulate the words from the utterance in the case of the Signed Czech (SC) or express the face gestures in the case of Czech Sign Language (CSE).

This paper describes experiences with a representation and a collection of the signs for an avatar animation. A lexicon of isolated signs in appropriate representation is the necessary part of the synthesis system. The everyday communication system intended to a certain domain involves that the lexicon includes the relevant signs only. A notation editor is one possibility how to create and administrate such a domain-specific lexicon of the relevant signs.

2. Synthesis System Background and Data Acquisition

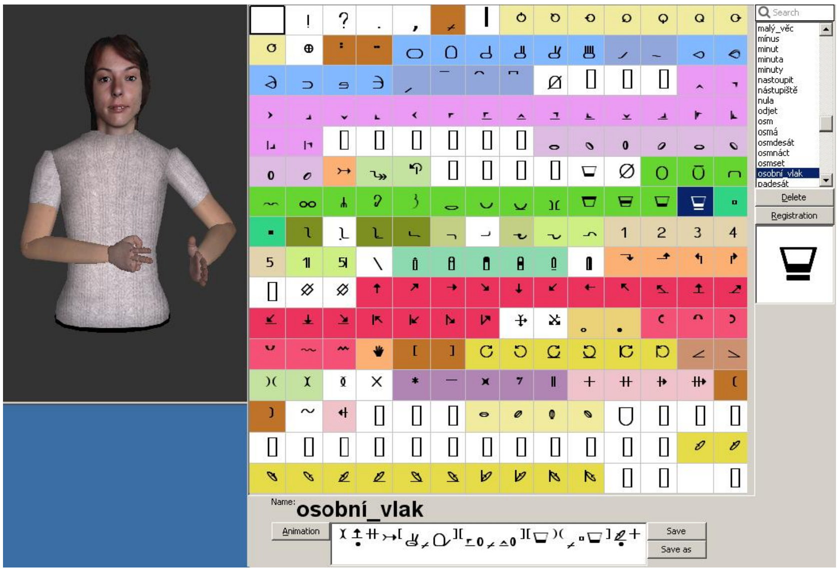
The straightforward solution of the signed speech synthesis should be based on video records of a real signing human. A concatenation of these records has better quality and realism than the avatar animation. On the other hand, the choice of the avatar animation allows the possibility of low-bandwidth communication, arbitrary 3D position and lighting, and the possibility of a change of an appearance of the animation model. There are two ways how to automatically solve the problem of the signed speech synthesis. The first one is based on the record of the real human motions in the 3D space and is called *data driven synthesis*. The second one is based on a symbolic notation of signs and is called *synthesis from the symbolic notation*. Each solution has certain advantages and disadvantages (Elliott et al., 2000; Kennaway, 2001).

The recorded data in data driven synthesis are processed and directly applied to the animation process. The advantages are the obtaining of the full 3D trajectories and the realistic motions of the animation model but the low accuracy and extensibility of recorded signs considered as the disadvantages. In addition, we need a special and expensive equipment to obtain the proper data. For example, the arm motions are recorded by some specialized motion capture systems[1]. The various shapes of hands have to be simultaneously measured by two data gloves[2] and for the acquisition of the face gestures we have to use the other motion capture system fixed on speakers head. The advantages of the synthesis from the symbolic notation are accuracy of the generated trajectories, easy editing of symbols, and easy extensibility by new notation features. A lexicon for the synthesis can be composed from different sources and created at different times. The disadvantages are complicated conversion of symbols to the articulation trajectories and the animation which looks robotic.

The reason for the choice of the symbolic notation is to provide the decomposition of signs to the smallest units,

[1] Vicon, Qualisys, BTS

[2] CyberGlove



Name: osobní_vlak

Figure 1: The screen shot of SignEditor. The notation example of sign “passenger train”.

components of the sign. This decomposition is essential for another linguistic research of a sign language. We have to mention that there is no universal sign language and the sign languages are not derived from spoken ones. For example, CSE has the specific morphology, phonetics and grammar. The basic item of CSE is the sign as in other sign languages. The sign mostly matches one word or concept in spoken language but this do not hold true in any case. The main difference between spoken Czech and CSE is that the CSE is visual-spatial language. It means that CSE is not perceived by ears but eyes, is based on shapes and motions in space. For example, the hand shapes are combined with the finger orientations in particular relationships between the dominant and the non dominant hand. In the case of the 3D trajectories acquired by the motion capture system this decomposition of sign can not be easily made. The question is how to transform these trajectories and representation for the “same” sign which is signed in different place of the sign space.

Hence, we have designed the rule based synthesizer (Krňoul et al., 2008) which uses the lexicon based on the symbolic notation. Two sign editors for administration of the lexicon are presented (Section 3.). The first editor is intended for the direct insertion of notation symbols and the second one is for more intuitive notation trough a graphical interface. Both editors share a feedback given by the avatar animation as support for the created or edited signs.

3. Notation System and Editors

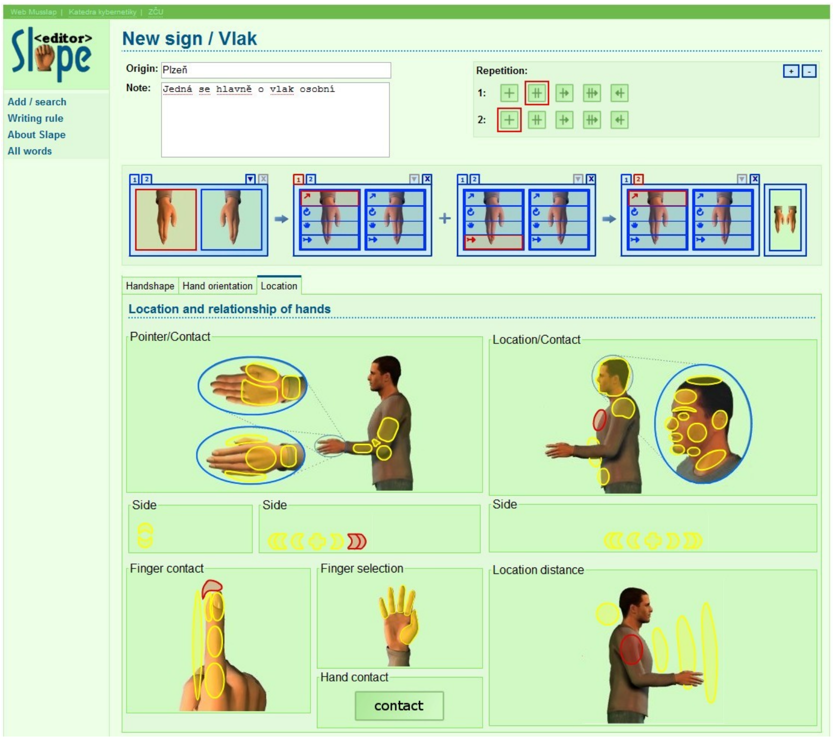
We consider the following assumption for the notation system. Each sign is composed from two components: the manual and non-manual. The non-manual component expresses the gesture of face, the motion and position of head and other parts of upper body. The manual component is expressed by shapes, motions and positions of hands. The signs are realized in a sign space. The sign space is approximately specified by the top of head, elbows sideways raise, and horizontal line below stomach.

We can found several notations for a general purpose and also for gestures of various sign languages (Stokoe et al., 1976; Liddell and Johnson, 1989; Macurová, 1996; Rosenberg, 1995). The majority of notations comes from the linguistic research of various sign languages where they substitute the written form. We have made the analysis of these notations with primal interest in the manual component and with the respect to the notation ambiguity for an automatic computer processing. The Hamburg notation system (HamNoSys) (Hanke and Schmaling, 1989) was chosen. HamNoSys version 3.0 was preferred for a low degree of ambiguity, good meaning of symbols, description of arm movements and hand shapes. However, we consider that a converter from one notation system to other should be developed in our future work.

3.1. SignEditor

This editor is intended for a direct notation of the signs in HamNoSys symbols. The main component of the editor is a table of all defined HamNoSys symbols (it is just only a character map of the HamNoSys font[3]). The symbols are divided into color groups which associate the symbols with a similar function. For example, the symbols for hand shapes are blue, the symbols for location are green, etc. The user can choose the particular symbols by double clicking on the picture of the symbol in the table. The selected symbols are directly entered into the edit line below the symbol table (the standard edit commands can be used in this line). The created sign can be named, saved, and processed to get its spatial form (the feedback avatar animation). The editor allows browsing the created lexicon and searching for the particular signs too. In Figure 1 is the screen shot of the SignEditor. There is the feedback animation on the left, the symbol table in the center, and the browsing window on the right.

[3] Available at http://www.sign-lang.uni-hamburg.de/Software /HamNoSys/HamNo33.zip



New sign / Vlak

Origin: Plzeň

Note: Jedná se hlavně o vlak osobní

Figure 2: The screen shot of SLAPE editor. The notation example of sign “passenger train”.

3.2. SLAPE

The direct notation requires the full familiarity with the given notation system. Therefore we have developed the editor SLAPE (Sign Language Portable Editor) to make the notation available for all users (including hearing impaired). The main role of SLAPE is expansion of the sign lexicon just by the hearing impaired users. SLAPE enables to notate new signs in a simple graphic way and edit already saved signs. The notation process requires only fundamental familiarization with the symbolic notation. The sign notation consists from the selection of relevant graphical icons. These icons represent the particular sign components. The selection is repeated until the whole sign is not completed. All selections are converted to the representation in the predefined notation system. Primarily, we have implemented the conversion to the HamNoSys.

The principle of the notation by SLAPE editor is based on the items which are arrange to an arbitrary length sequence. Each item consists of two panels for the dominant and non dominant hand. Users can use one or both panels to select icons for particular hand shape, orientation, location, movement or select icons for a hand symmetry. The items are successively filled according to passage of notated sign. The connection determines the time relationship of the neighboring items. By clicking on connection, user can determine which items will be performed sequential or simultaneous. The items can share additional properties, for example the repetition or movement modalities. The screen shot of editor is depicted in Figure 2.

The SLAPE is implemented as a client-server web application. The server is implemented in Java. It executes user’s requires and provides a storing of the signs in the database. The Hibernate tool is used to implement the object relation mapping on the persistent layer for the communication with the database. The JBoss Seam framework is used as base structure to integrate Java Server Faces and Facelets tools. The client is implemented in the HTML and JavaScript code and runs in an arbitrary web browser. The Flash technique is applied for the design of notation forms and icons. The client provides good portability on various operation systems and platforms.

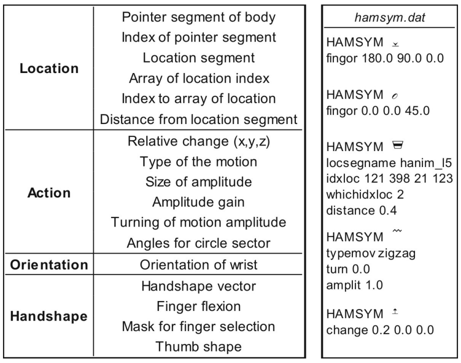
4. Feedback Animation

The usage of HamNoSys notation without any feedback makes the possibility of the structural mistakes. Therefore, our editors are supplemented with a feedback module to provide the correctness of the notation and immediately visualization of the created sign. The feedback module can be divide to a module for a rendering of the animation model and to a module for a forming of the animation trajectories (a trajectory generator).

4.1. Rendering of Animation Model

Our animation algorithm employs a 3D geometric animation model of the avatar in compliance with the H-Anim standard[4]. The animation model covers 38 joints and body segments. Each segment is represented as textured triangular surface with the normal vector per vertex. The segments are connected by joints of an avatar skeleton. One joint per segment is sufficient for this purpose. A controlling of the skeleton is carried out thought the rotation of joints around three axises (x, y, z). The rotations of the shoulder, elbow, and wrist joints are not directly controlled but they are completed from 3D positions of the wrist joints. The inverse

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



Location	Pointer segment of body Index of pointer segment Location segment Array of location index Index to array of location Distance from location segment
Action	Relative change (x,y,z) Type of the motion Size of amplitude Amplitude gain Turning of motion amplitude Angles for circle sector
Orientation	Orientation of wrist
Handshape	Handshape vector Finger flexion Mask for finger selection Thumb shape

hamsym.dat

```
HAMSYM
fingor 180.0 90.0 0.0

HAMSYM
fingor 0.0 0.0 45.0

HAMSYM
locsegname hanim_l5
idxloc 121 398 21 123
whichidxloc 2
distance 0.4

HAMSYM
typemov zigzag
turn 0.0
amplit 1.0

HAMSYM
change 0.2 0.0 0.0
```

Figure 3: Left panel: The list of all items. Right panel: An example of the items stored in the definition file.

kinematics[5] is employed to perform the analytic computation in the real time.

The further control of the animation model is performed by a local deformation of the triangular surfaces. The local deformations are for the detail animation of an avatar pose. It is primarily used for the animation of the avatar's face and the tongue. The triangular surfaces are deformed according to the designed animation schema in our synthesis approach (Krňoul et al., 2008). The deformation of the given triangular surface is defined by a transformation individually given for each vertex. These transformations are derived from influence zones defined on the triangular surface by spline functions which are constructed from several 3D points. The rendering of the animation model is implemented in C++ and OpenGL language. The animation model is shown in Figure 1 on the left.

4.2. Trajectory Generator

The HamNoSys is detailed enough. However, it is difficult to define some rules and actions for all symbol combinations to cover the entire notation variability. We made a few restrictions in order to preserve maximum degree of freedom. In this assumption, the annotation of the sign have a good meaning for the user familiar with HamNoSys as well as signs are obvious enough for the transformation to the avatar animation.

The trajectory generator automatically carries out the syntactic analysis of the symbolic string on the input and creates a tree structure. For structurally correct symbolic string, we have one parse tree where each no leaf node is determined by one parsing rule. Each node of the tree is described by two key frames to distinguish the dominant and non dominant hand. The structure of the key frame is composed from a set of items specially designed for this purpose (Figure 3 on the left). These items are filled in each leaf node from the symbol descriptors stored in the definition file (Figure 3 on the right). Currently, the definition file covers 138 HamNoSys symbols. The generator uses 374 parse rules to perform syntactic analysis of the input string. In addition, the 39 rule actions were added in manner that one rule action is connected with each parse rule. The number of used symbols, parse rules, and actions is in Figure 4.

Block	HamNoSys Symbols		Parser	
	#Base	#Auxiliary	#Rules	#Actions
Symmetry	4	0	8	8
Handshape	12	11	35	6
Finger and palm orientation	26	3	34	5
Location	30	17	85	14
Action	40	42	204	15
Link of blocks	0	0	8	1

Figure 4: The statistic of symbols, rules, and actions used by the HamNoSys parser.

The processing of the parse tree is carried out by several tree walks whilst the size of the tree is reduced. The initial tree walks put together the items of the key frames according to the type of the rule actions. The reduced tree is processed by the next tree walks to transform the key frames to the trajectories accordance with the timing of the particular nodes. Finally, we obtain the final trajectories for both hands in the root node of the tree. The final step is transforming the trajectories into the avatar animation.

The acceptance of signs defined as a string of HamNoSys symbols by the parser causes some limitations. The order of the HamNoSys general notation structure defined as block sequence of a symmetry operator, starting point configuration, and actions is completely preserved. For the block of the starting point configuration, the hand shape and a finger orientation is without any restriction as well as the block of symmetry operators in all eight variants. The variants of hand location for separate pose of dominant or non dominant hand agree to HamNoSys body location symbols table. The only limitation is in the notation of two handed locations. The location symbols for finger, hand and arm are involved in relation to two handed location where the notation of the precise contact is extended. We have implemented two precise variants of the relationship of the dominant hand:

- Relationship between the dominant hand and body: We have to select one symbol to determine pointer location of the dominant hand, further symbol to determine the target body location and finally symbols to define the type of notated relationship.
- Relationship between the dominant hand and non dominant hand: We have to select one symbol to determine pointer location of the dominant hand, one symbol to determine the dominant hand target location, further the symbol for the type of the relationship and finally the symbol for the hand location.

The example of annotation is showed in Figure 5. In contrast to HamNoSys definition, the type of the relationship should be one of the list: behind the body, in contact with body, near to the body or the farthest distance.

Such relationship of hands can be used for arbitrary hand segment and location in correspondence with the notation obvious for synthesis process. The block of starting point configuration is followed by block of actions or as well block of movements. The HamNoSys definition of movements on absolute and relative is preserved too. The rela-

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

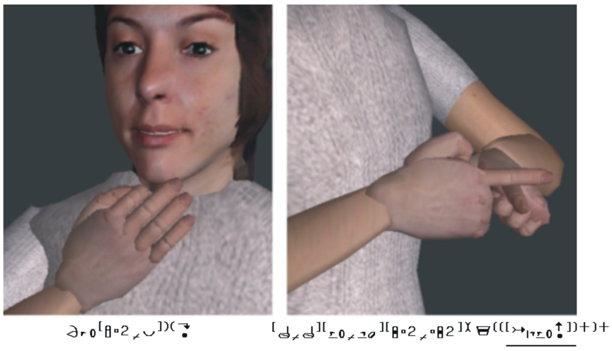


Figure 5: The variants of the dominant hand relationship. On the left is close relationship between index finger and chin, on the right is precise contact of index fingers.

tive movements are notated as base movement with modification to determine the path of wrist, for example small straight movement followed circular fast movement with the decreasing diameter. The local movement, as "fingerplay" or wrist movement, are considered as relative movements and are fully implemented.

However, the difference is in the relative movement describing a replacement of hand shape and orientation. It is put together with the notation of an absolute movement thus that the notation of replacement of hand shape and orientation is preserved and is extended about possibility notation of location symbols. Such the notation variant shares same notation structure as starting point configuration and can be used for arbitrary absolute movement. The example of these notation variants is depicted in Figure 5 on the right.

The separated notation of two-handed movements is implemented according to HamNoSys manual but with one limitation. Two-handed movements and the symmetry symbols exclude each other. The order of notated movements is implicitly sequential. The notation of simultaneously performed movements is implemented in the original meaning but the notation of the symbol sequence for a fusion of movements is not supported.

5. Lexicon Creation

We have created the domain-specific lexicon for our synthesis system from railway station domain. The signs which need to be notate were collected by the inspection of the Czech to Signed Czech (CSC) parallel corpus (Kanis et al., 2006) and translations of train announcements. The CSC corpus contains 1109 dialogs from the telephone communication between a customer and an operator in a train timetable information center.

Further, we discuss the actual experience with the lexicon creation process. We have began the trial annotation process with six annotators to test the convenience of the SignEditor and the feedback animation. We have divided the annotation process to two steps. In the first step, four annotators who are not familiar with CSE were employed to insert the signs in the direct editor. The annotators use the video dictionaries of CSE (Potměšil, 2004; Potměšil, 2005; Langer et al., 2004; Gura and Ptáček, 1997) as a source for the sign notation. In the second step, two remaining annotators (inspectors, familiar with CSE) were employed to correct the entered signs. The inspector use the SignEditor to replay signs and put comments about the correctness of the rendered animations.

Function	Symbol	Signs [%]
Handshape	▢	30
Finger direction	⌐	31
Palm orientation	⊙	50
Target location	⊌	58
Pointer location	▯	20
Direct movement	↥	15
Circular movement	↻	1

Figure 6: The overview of the most frequent symbols for the particular operations.

The feedback animation forces the annotators to use the structural correct sequence of symbols. It ensures that the signs in lexicon are still in correct form while the annotation process runs. At the begging of annotation work, the annotators were not familiar with the HamNoSys notation which leads them to create needlessly complicated sequences of symbols. For example, they have not used the symbols for symmetry or a symbol sequence for the precise contacts and they were not able to annotate some seemingly easy signs. After antiquation of these initial troubles, the average annotation speed with the SignEditor was approximately two signs per hour.

The lexicon currently contains approximately 330 signs. By the inspection of the lexicon, we can observe that all signs include some variant of the starting point configuration. The most frequent symbols for each block of the notation are summarized in Figure 6. It is interesting that the most frequent symbols in the starting point configuration are the hand shape symbol described open hand and the location symbol for thorax. The most frequent sequences of symbols in the movement block are these for the relative change of hand shape and the finger orientation or absolute displacement of the wrist position. More than 55% of all signs in the lexicon contain these sequences. The majority of this sequences is anotated in combination with another simultaneously performed movements[6]. The 54% of all signs are notated with the symbols for straight movements which form the path of writs. The symbols of circular movemnets were chosen for 7% of all signs only. The symbols for the movement repetation are aproximatelly used for 30% of all signs.

The annotators have had problems with notation of very small movements. The modification symbols for small, normal and large movements seems not to be sufficient. Furthermore, they have had also problems with location symbol for thorax. It seems that the current annotation vari-

[6] In the new concept of HamNoSys 4.0 version is posible to anotate this compound movement with notation of the wave symbol under the symbol for the finger direction or the palm orientation. This can be simplification for anotators but it does not give full substitute for mentioned compound movement.

ant is not enough in this case. There could be more possibilities how to annotate more precise this location. These experiences partially agrees with comments by inspectors who check signs in the lexicon. The most frequent inspector's comments are related to incorrect notation using symbols for:

- hand shape including configuration of thumb
- finger or palm direction
- location given by thorax symbol
- number of repetition
- speed of movements

Nevertheless, the incorrect notation is caused by a collision of dominant hand with body. The some collided signs can be corrected the notation sequence for the precise contact. However, some collisions are caused by moves of hands in the proximity of body and can not be thus corrected in the same way.

The several limitations of notated signs are also caused by missing features in current implementation of the trajectory generator. The missing rule action for movement modality, for example the notation of fast or slow movements, are very important and have to be implemented for the following lexicon creation. The important feature, which should be included too, is the symbol sequence contains "between" symbol. Thus several hand shapes and locations should be repair and represent more precisely. The notation variant with symbols for contact in action block is not yet implemented. This variant will be implemented by a solving of the body segment collisions in more general way to avoid all possible collisions occurred in the synthesis process.

6. Conclusion

We have discussed the experiences with the domain-specific lexicon for the automatic signed speech synthesis. Two editors for notation of signs in HamNoSys symbols were introduced. The first one is SignEditor which is intended for the direct insertion of the notation symbols. The second one is SLAPE which is designed for more intuitive notation trough the graphical interface and for the good portability. Both editors share a feedback given by the avatar animation as support for the created or edited signs. The SignEditor was used to create our lexicon for the railway station domain. Nowadays, the lexicon contains more than 300 signs of CSE.

7. Acknowledgments

This research was supported by the Grant Agency of Academy of Sciences of the Czech Republic, project No. 1ET101470416 and by the Ministry of Education of the Czech Republic, project No. MŠMT LC536.

8. References

Pavel Campr, Marek Hrúz, and Miloš Železný. 2007. Design and recording of czech sign language corpus for automatic sign language recognition. In *Proceedings of the Interspeech 2007*. Antwerp, Belgium.

Pavel Campr, Marek Hrúz, Jana Trojanová, and Miloš Železný. 2008. Collection and preprocessing of czech sign language corpus for sign language recognition. In *LREC 2008*. Workshop proceedings: Construction and Exploitation of Sign Language Corpora, ELRA.

R. Elliott, J.R.W. Glauert, J.R. Kennaway, and I. Marshall. 2000. The development of language support for the visicast project. In *4th Int. ACM SIGCAPH Conference on Assistive Technologies (ASSETS 2000)*. Washington DC.

Tomáš Gura and Václav Ptáček, 1997. *Dictionary of Czech Sign Language (in Czech)*. ČUN, Praha, Olomouc, Czech Rep.

Thomas Hanke and Constanze Schmaling, 1989. *HamNoSys, Version 3.0.* University of Hamburg, http://www.sign-lang.uni-hamburg.de/Projects /HamNoSys.html.

Jakub Kanis and Luděk Müller. 2007. Automatic czech - sign speech translation. *Lecture Notes in Artificial Intelligence 4188*.

Jakub Kanis, Jiří Zahradil, Flilip Jurčíček, and Luděk Müller. 2006. Czech-sign speech corpus for semnatic based machine translation. *Lecture Notes in Artificial Intelligence 4188*.

J.R. Kennaway. 2001. Synthetic animation of deaf signing gestures. In Ipke Wachsmuth and Timo Sowa, editors, *Lecture Notes in Artificial Intelligence*, pages 146–157. London.

Zdeněk Krňoul, Jakub Kanis, Miloš Železný, and Luděk Müller. 2008. Czech text-to-sign speech synthesizer. *Machine Learning for Multimodal Interaction, SeriesLecture Notes in Computer Science*, 4892:180–191.

Jiří Langer, Václav Ptáček, and Karel Dvořák, 2004. *Vocabulary of Czech Sign Language (in Czech)*. Univerzita Palackého v Olomouci, Olomouc, Czech Rep.

S K Liddell and R E Johnson. 1989. American sign language: the phonological base. *Sign Language Studies*, 64:195–277.

Alena Macurová. 1996. Why and how to notate signs of czech sign language (notes to discussion) (in czech). *Speciální pedagogika*, 6:5–19.

Miloň Potměšil. 2004. *Dictionary of Czech Sign Language O-Ž (in Czech)*. Fortuna, 1 edition.

Miloň Potměšil. 2005. *Dictionary of Czech Sign Language A-N (in Czech)*. Fortuna, 1 edition.

Amy Rosenberg. 1995. Writing signed languages in support of adopting an asl writing system. Master's thesis, University of Virginia.

William C. Stokoe, Carl Croneberg, and Dorothy Casterline. 1976. *A Dictionary of American Sign Language on Linguistic Principles*. Silver Spring, 2 edition.

Miloš Železný, Zdeněk Krňoul, Petr Císař, and Jindřich Matoušek. 2006. Design, implementation and evaluation of the czech realistic audio-visual speech synthesis. *Signal Procesing, Special section: Multimodal human-computer interfaces*, 86:3657–3673.