# TuniSigner: A Virtual Avatar to Interpret SignWriting Notations

Yosra Bouzid & Mohamed Jemni

Research Laboratory LaTICE, University of Tunis, Tunisia

The International SignWriting Symposium, 21-24 July, 2014

# Overview

Sign language is an integral part and an identifying feature of membership in the Deaf Culture

> According to WFD, there are about 70 million deaf people who use sign language as their first language or mother tongue.

Sign language is a complex natural language with its own grammatical rules and syntax, but does not have until now a widely established writing system.

# Overview

The lack of a standard writing system for SL limits the possibility to provide information (e.g. on the web) directly in a form equivalent to the signed content.

Deaf people are often required to access information and knowledge in a language that is not natural to them, and this can cause serious accessibility problems in their daily lives especially for those how have low literacy skills.

Around 80% of deaf people worldwide have an insufficient education, literacy problems and lower verbal skills.

# The advantages of SL written forms

The main benefit of having a SL written form is that Deaf signers could :

- Express, share, and record their ideas and thoughts on paper without translating it all the time.
- Learn new things and skills outside of oral communication.
- Improve their ability to comprehend and acquire the written versions of oral language.
- ……

# SL Writing Systems

A good writing system for a signed language

- should have an approximately one-to-one correspondence between symbol and sign formational aspect.
- should handle the three-dimensionality of signing
- should not be difficult to write or read

There are co-existing proposed writing systems for sign language, of which the following are some examples: Stokoe system, Hamburg Notation System – HamNoSys and SignWriting.

# SL writing systems

**Stokoe Notation**

- The first phonemic script used for sign languages.
- It closely reflects a linguistic analysis of SL structure focused primarily on the signs' ***manual*** components.
- It does not include non-manual components like facial expressions and body movements.
- It was not meant to be used for writing full sentences.
- It has been used mostly by linguistics and researchers.

# SL writing systems

**HamNoSys Notation**

- HamNoSys is designed to be able to write any signed language precisely
- It provides a linear representation of SL constituent units
- It does not provide any easy way to describe the NMFs
- It is extremely difficult to use it for transcribing sequences of signs and actual signed discourse
- It has been used mostly by linguistics and researchers.

# SL writing systems

**SignWriting Notation**

- SignWriting is designed to be appropriate for any sign language
- It uses a set of highly iconic symbols that can be combined to describe any sign
- It can easily indicate facial expressions, body movements and long speech segments
- It is conceived to be used in writing sign languages for the same purposes hearing people commonly use written oral languages.

# SL writing systems

**SignWriting Notation**

Although SignWriting closely visually resembles the concrete signs, a training to learn to interpret the static transcriptions is needed for deaf signers who are accustomed to use their native language in a visual-gestural modality.

The bi-dimensional representation of SignWriting notations may inadvertently create confusion and ambiguity to these signers since the three-dimensional nature of signing cannot be fully reflected into a symbolic transcription.

# Contribution

We propose an avatar based system to automatically interpret the exact gestures represented within SignWriting transcriptions.

The virtual avatar is driven by an animation software which generates motion data in real time from a scripting language called SML (Sign Modeling Language) designed for describing signing gestures.

Signing avatar provides a cost effective and efficient way to make sign language notation content more accessible for Deaf users.

# Contribution

A virtual avatar driven by animation software provides an attractive alternative to video:

- Signed content can be created by one person on a desktop computer. No video capture equipment is required.
- The user has extra control that is not possible with video. The view angle can be continuously adjusted during playback.
- Details of the animation content can be edited without having to rerecord whole sequences.
- Disk space demands to store sign description are negligible.

# System Description

Our system architecture is divided mainly into three parts:

- The first part is devoted to parse and process the SignWriting notations which are provided in an XML based format (SWML).
- The second part is dedicated to provide a linguistic representation for each notation in order to specify how correspondent signs are articulated.
- The third part is devoted to convert the obtained linguistic representations to SML (Sign Modeling Language) for rendering avatar animations.

# System Description

***SignWriting notation***

```
<signbox>
 <sym width="25" height="17" left="0" top="8">01-05-055-01-01-02</sym>
 <sym width="13" height="13" left="-1" top="28">02-01-007-01-01-01</sym>
 <sym width="36" height="46" left="-18" top="-23">04-05-011-01-01-01</sym>
 <sym width="15" height="14" left="16" top="27">02-05-001-01-01-03</sym>
</signbox>
```

**Parsing and processing the SW notation**

**Identifying the explicit linguistic representation of the sign**

**Scripting and generating animations (XML Sign Modeling Language, SML)**

# System Description: (Part 1)

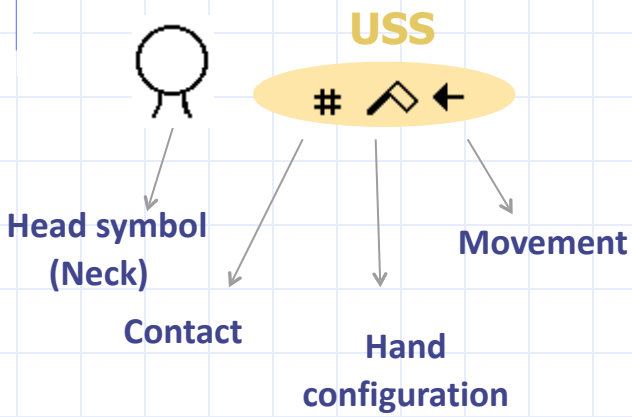
The SignWriting Markup Language (SWML) is an encoding format for SignWriting documents, using XML (extensible Markup Language).

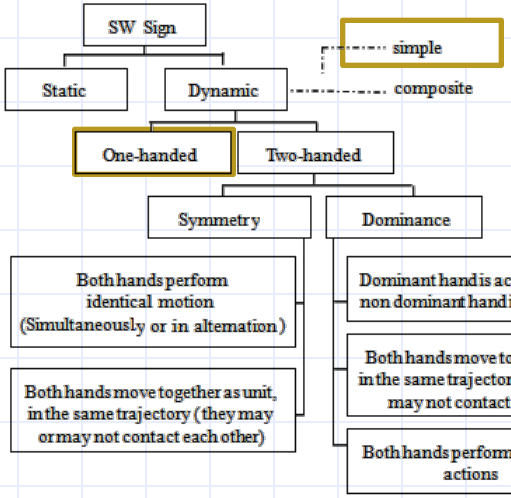
→ SWML does not save any order in which the symbols are entered to create a sign, the symbols are simply positioned in 2D signbox.

→ SWML does not describe the relation between the symbols, while their relation can have various meanings.

the SWML encoding of the sign « have » does not provide any information to indicate if the contact occurs between the two hands or between hands and the signer's body.

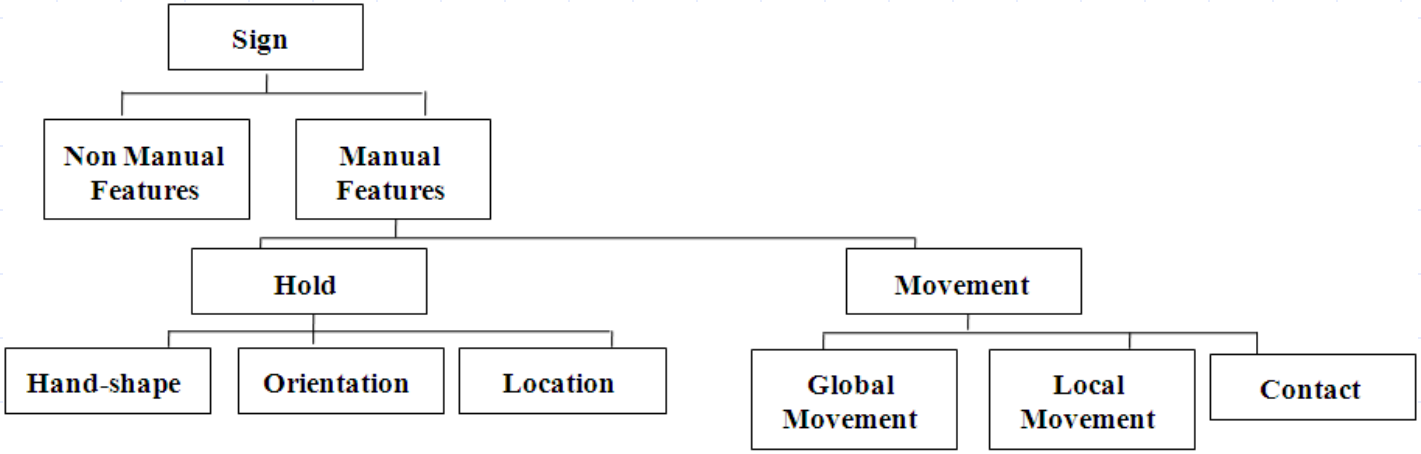

```
<signbox>
  <sym width="25" height="17" left="0" top="8">01-05-055-01-01-02</sym>
  <sym width="13" height="13" left="-1" top="28">02-01-007-01-01-01</sym>
  <sym width="36" height="46" left="-18" top="-23">04-05-011-01-01-01</sym>
  <sym width="15" height="14" left="16" top="27">02-05-001-01-01-03</sym>
</signbox>
```




# System Description: (Part 2)

**Rendering sign language**, in the form of 3D **animations**, requires the definition of all relevant features of signing gestures (phonemes).




However, SWML is not complete enough and phonologically-based enough to be used for the underlying linguistic representation of a sign. It is merely an XML adaptation of SignWriting which can provide information about the relative position of each basic symbol in the notation.

# System Description: (Part 2)

The linguistic model of the sign needs to be constructed in order to ensure the correct performance of avatar gestures.

```
<sign hands=single inUnity=false symmetry=false >
 <Posture>
   <Right Hand  shape=H131  orientation=WP-Face  rotation=45   x=-1.0 y=1.6  z=5.5  />
 <Posture/>
 <Movement>
   <RightHand  globalMovement=FP_straight rept=1  size=small  dir=left  speed=normal />
 <Movement/>
 <Movement>
   <RightHand  contact=strike  repeat=1  x=-1.0 y=1.6  z=5.5 />
<Movement/>
<sign/>
```

# System Description: (Part 2)

```
<sign hands=single inUnity=false symmetry=false >
 <Posture>
   <Right Hand  shape=H131  orientation=WP-Face  rotation=45  x=-1.0 y=1.6  z=5.5  />
 <Posture/>
 <Movement>
   <RightHand  globalMovement=FP_straight rept=1  size=small  dir=left  speed=normal  />
 <Movement/>
 <Movement>
   <RightHand  contact=strike  repeat=1  x=-1.0 y=1.6  z=5.5 />
<Movement/>
<sign/>
```

```
<Sentence>
  <word>
    <join name="l_thumb1">
      <mouvement>
        <duration>1.0</duration>
        <rotation type="euler">
          <heading>45</heading>
          <attitude>50</attitude>
          <bank>40</bank>
        </rotation>
      </mouvement>
    </join>
    <join name="l_thumb2">
      <mouvement>
        <duration>1.0</duration>
        <rotation type="euler">
          <heading>50</heading>
          <attitude>57</attitude>
          <bank>72</bank>
        </rotation>
      </mouvement>
    </join>

    <join name="l_wrist">
      <mouvement>
        <duration>0.8</duration>
        <rotation type="euler">
          <heading>35.0</heading>
          <attitude>45.0</attitude>
          <bank>50.0</bank>
        </rotation>
      </mouvement>
    </join>
    <join name="l_wrist">
      <mouvement>
        <duration>0.8</duration>
        <translation>
          <x>-1.0</x>
          <y>1.6</y>
          <z>5.5</z>
        </translation>
      </mouvement>
    </join>
  </word>
</Sentence>
```


*Gesture Description* → **Movement Specification Using SML** → *Animation*

# System Description: (Part 3)

The Sign Modeling Language (SML) is an XML-based descriptive language developed by WebSign team to provide an extra layer around X3D and facilitate the 3D virtual agent manipulation.

SML describes the avatar animations in terms of translations and Euler rotations of a group of joints in a fixed time. It is able to control not only hand gestures but also facial expressions and body movements.

SML script is interpreted by an animation solver based on inverse kinematics to perform the analytic computation of avatar joints in the real time.

# Demonstration

# Demonstration

# Conclusion

We have presented a new approach for automatically synthesizing 3D signing animations from SignWriting notation using avatar technology.

tuniSigner has interpreted more than 1200 notations from different sign languages (American Sign Language, French Sign Language, Egyptian Sign Language, Brazilian Sign Language, Tunisian Sign Language).

Unlike the previous works, VSign and SASL projects, that generate MPEG-4 BAP sequences directly from the SWML signbox to drive a virtual signer, this system has used a simple gesture description to reformulate the different features of the sign and convert it then into SML for rendering the corresponding signing animations.

Thank you for your attention