

SWift, a user-centered digital editor for SignWriting within SWORD project

Fabrizio Borgia^{1,2}, Claudia S. Bianchini³, Maria De Marsico²

¹ Université Toulouse III - Paul Sabatier

² Sapienza Università di Roma

³ Université de Poitiers

+ Outline

- Introduction
- SignWriting digital editors - Overview
- SignWriting digital editors - Challenges
- SWift - Introduction
- SWift – Overview
- SWift - Advanced features
- SWift – Supporting different ISWA versions
- SWift - Testing
- Conclusions and Future

+ Introduction

- A pencil and a piece of paper are the only required items to produce signs using SignWriting
- Since the early years the need to produce a digital version of the system was evident
- The informatization of the system, started in 1896 with the SignWriter computer program, allowed SignWriting to achieve a wider diffusion through:
 - Newspaper
 - Books
 - Websites
 - Other digital resources

+ Introduction

- The glyphs were joined by their digital duplicates
- The organization was ensured by assigning to each of them a unique ISWA code



+ SignWriting digital editors

Overview

- SignWriting digital editors are the tools that enable the creation of digital SL resources written in SignWriting
- In other words, they are critical for the informatization of SignWriting
- Many applications have been produced by different teams, delivered in different ways, ranging from desktop to web applications

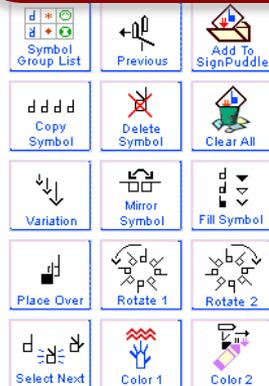
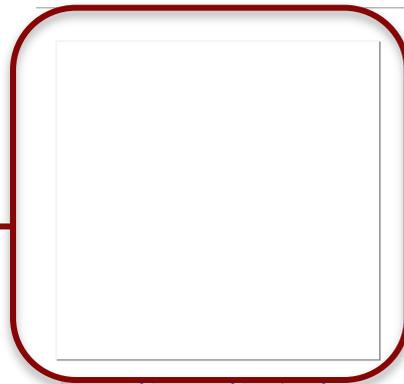
+ SignWriting digital editors

Overview

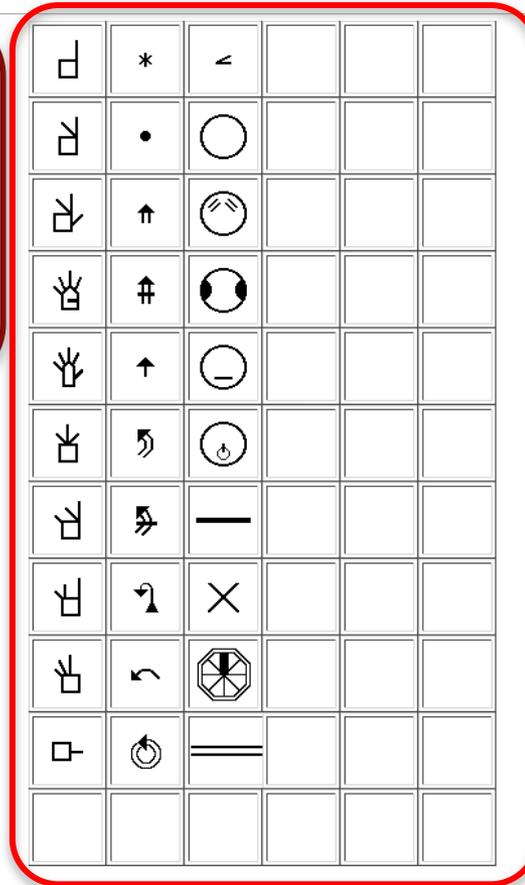
- Most SignWriting digital editors basically provide the same functionalities
- Despite differences in design and implementation existing from one editor to another, such functionalities are:
 - Search for (or type) glyphs which belong to the ISWA
 - Insert the chosen glyphs onto an area which is designated for the composition of the sign
 - Manage the glyphs on the sign composition area
 - Save the sign in one (or more) formats

+ Digital SignWriting Overview

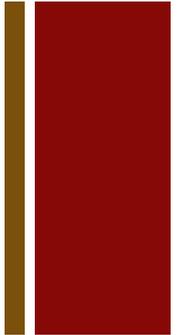
Sign Composition



Glyph Search



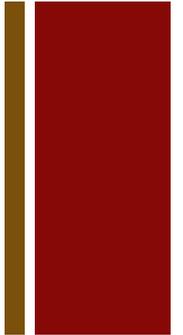
+ Digital SignWriting Challenges



General:

- The system should provide at least:
 - An area to compose the desired sign(s)
 - An area to search for the desired glyphs(s)
- Optimal level of usability
- Optimal level of accessibility (at least deaf-oriented accessibility)
 - Minimize the use of Vocal Language
 - Visual-oriented organization

+ Digital SignWriting Challenges



Glyph Search Area:

- The large amount of glyphs requires a robust yet intuitive organization of the glyph search area
- The organization of the glyph search area should reflect the organization of the International SignWriting Alphabet (ISWA)
- The glyph search area should provide a user-friendly navigation context for finding and fetching glyphs
- SignWriting beginners should be able to find their orientation within the glyph search area too

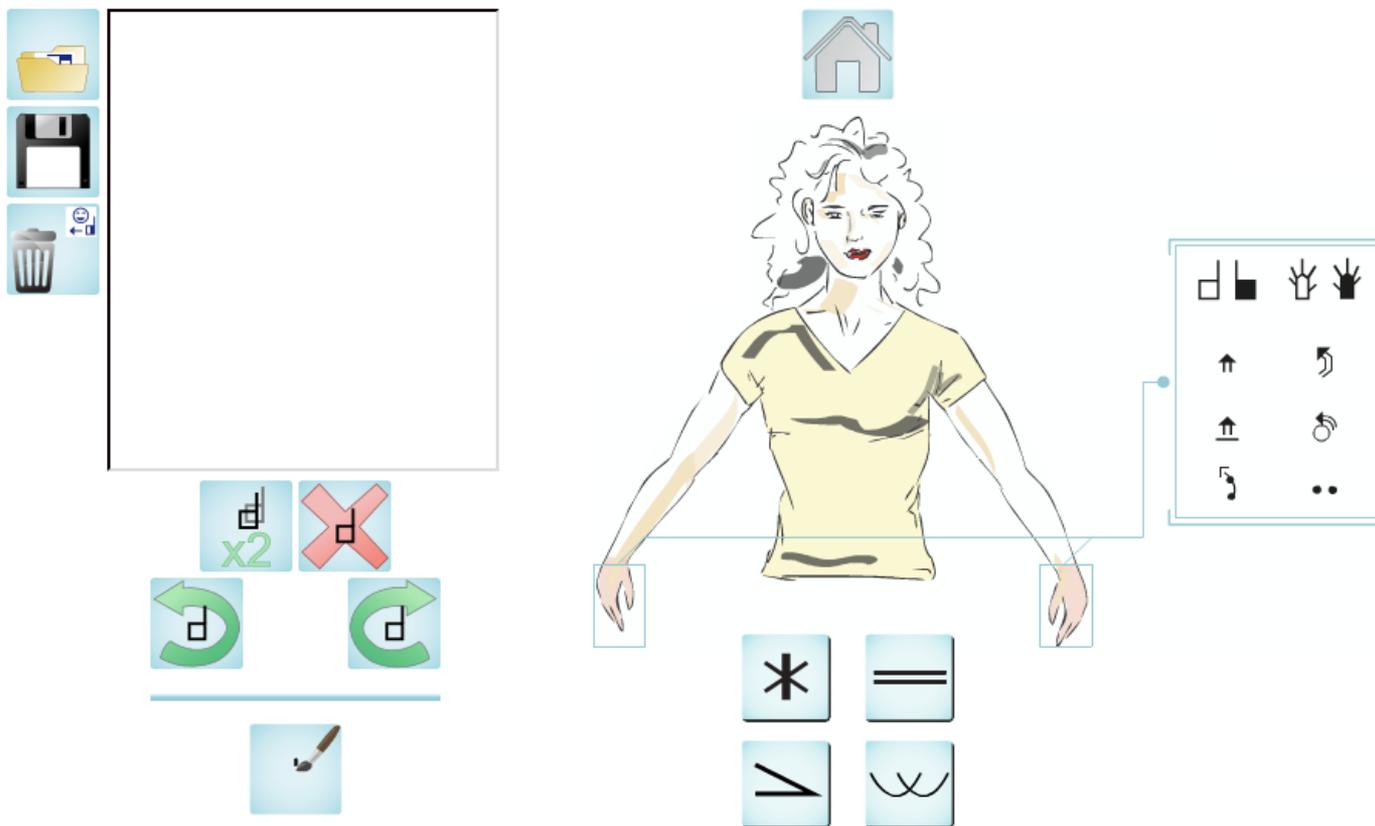
+ SWift Introduction



SignWriting improved fast transcriber

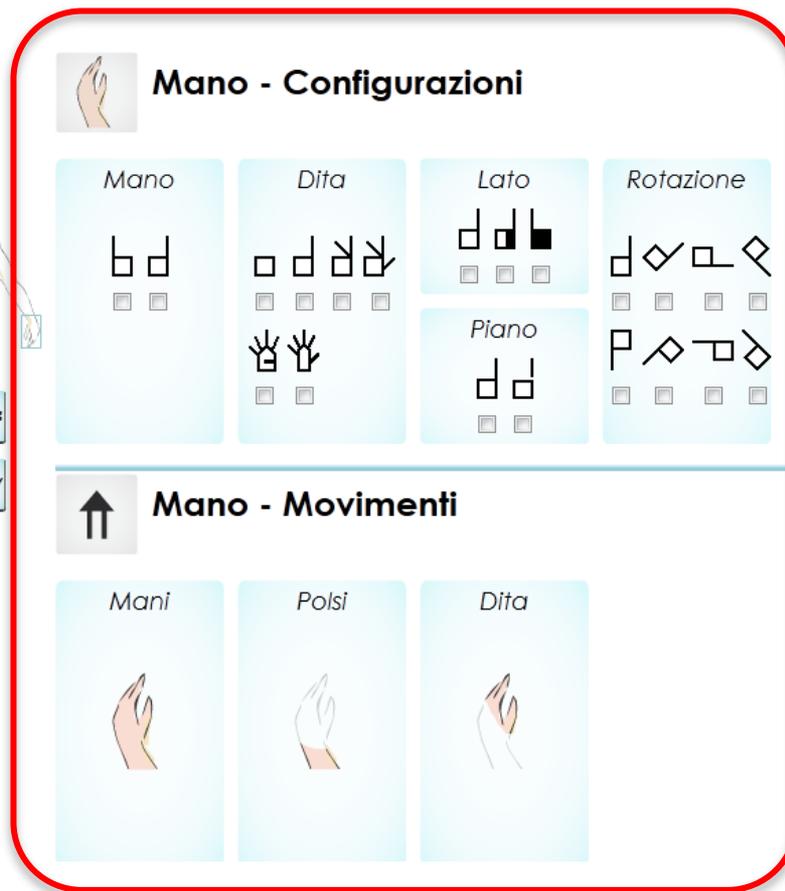
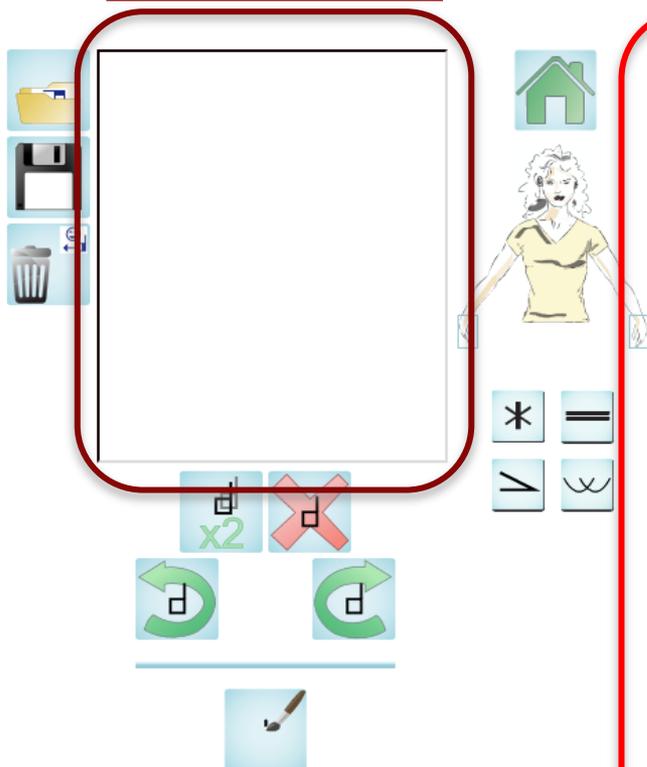
- Provides all the features of a SignWriting digital editor
- Produced by “Sapienza” University of Rome, in partnership with the research team at ISTC-CNR (which includes many deaf people)
- New features with respect to other digital editors
- Key requirements: usability and accessibility
- Application tested with deaf participants

+ SWift



+ SWift

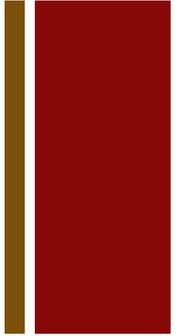
Sign Composition



Glyph Search

+ SWift

Overview

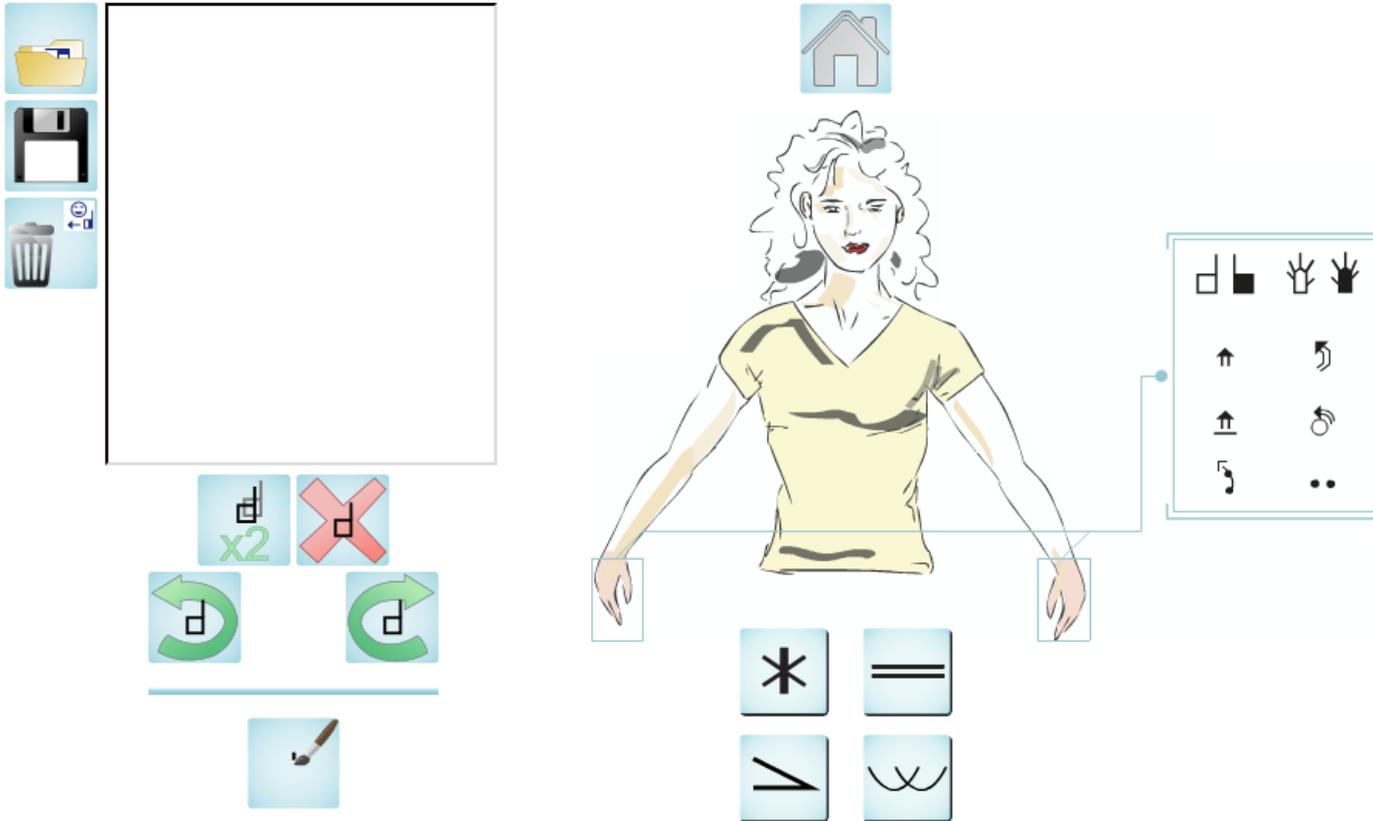


Basic features:

- Sign composition area and glyph search areas
 - Interaction via click, and drag-and-drop
- Functionalities for glyph editing
 - Rotation, duplication, deletion, etc
- Multiple save options
 - PNG, XML and remove save
- “Open” functionality to edit signs composed previously

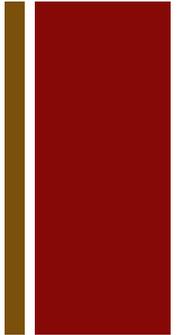
+ SWift

Overview



+ SWift

Overview

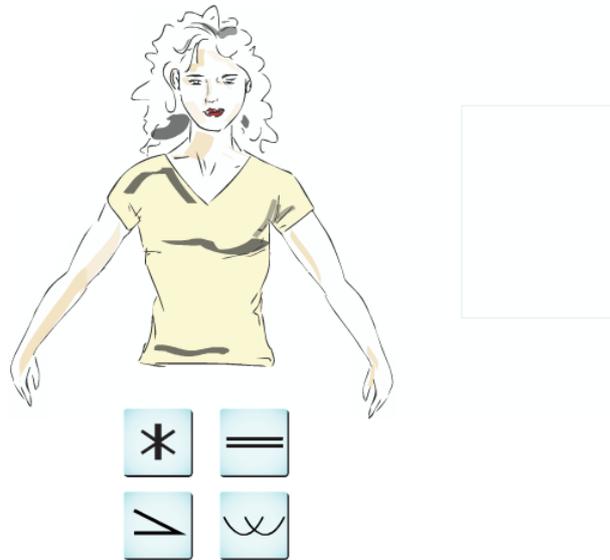


- Support for SignWriting beginners
 - Presence of graphic elements within the glyph search area, to guide the user towards the correct glyph
 - Use of mouseover-activated animations to provide additional information to the user
 - Buttons that are used to modify glyphs have very simple icons

+ SWift

Overview

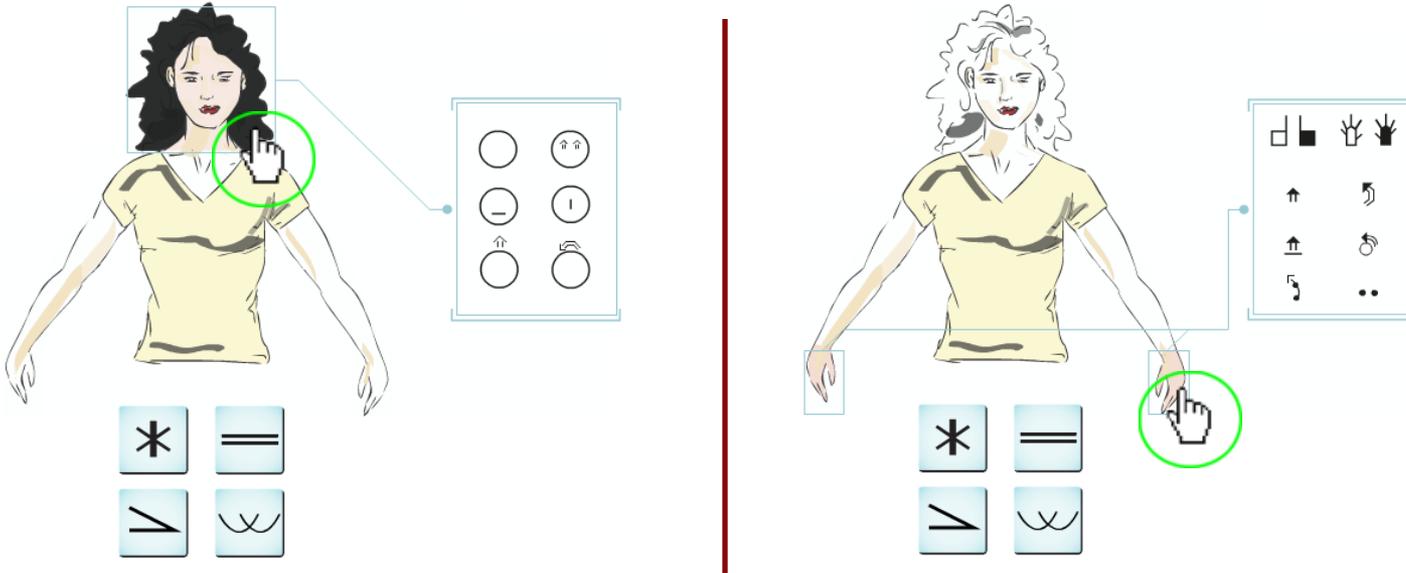
- Support for SignWriting beginners
 - The puppet: a stylized human figure helps the user to choose the appropriate anatomic area (head, hands, etc.) of the glyph
 - Animations on mouseover show a set of example glyphs for any area



+ SWift

Overview

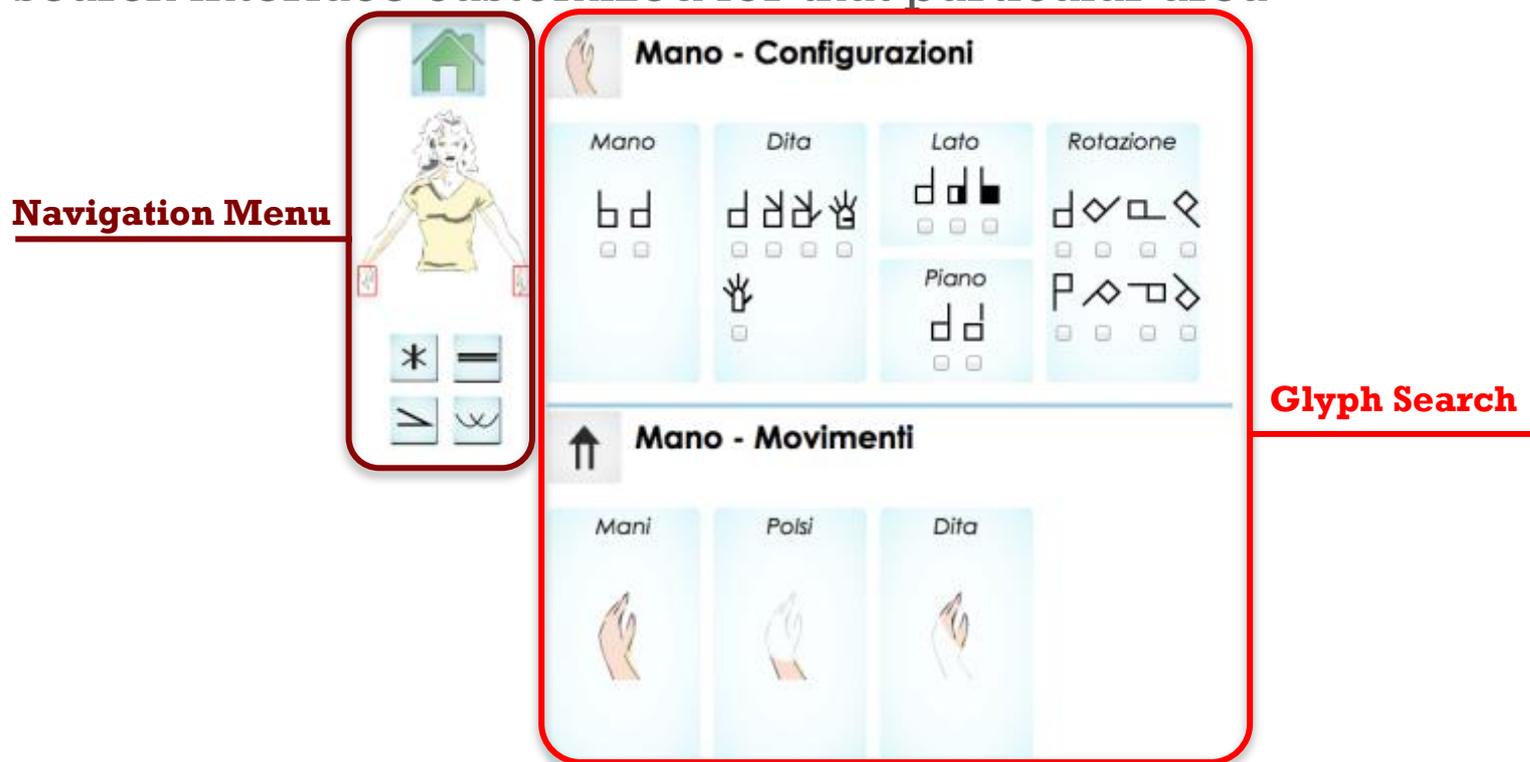
- Support for SignWriting beginners
 - The puppet: a stylized human figure helps the user to choose the appropriate anatomic area (head, hands, etc.) of the glyph
 - Animations on mouseover show a set of example glyphs for any area



+ SWift

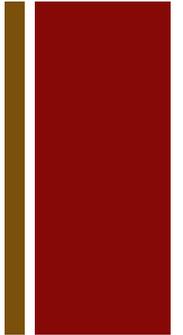
Overview

- As the user chooses an glyph area, he/she is presented with a search interface customized for that particular area



+ SWift

Overview



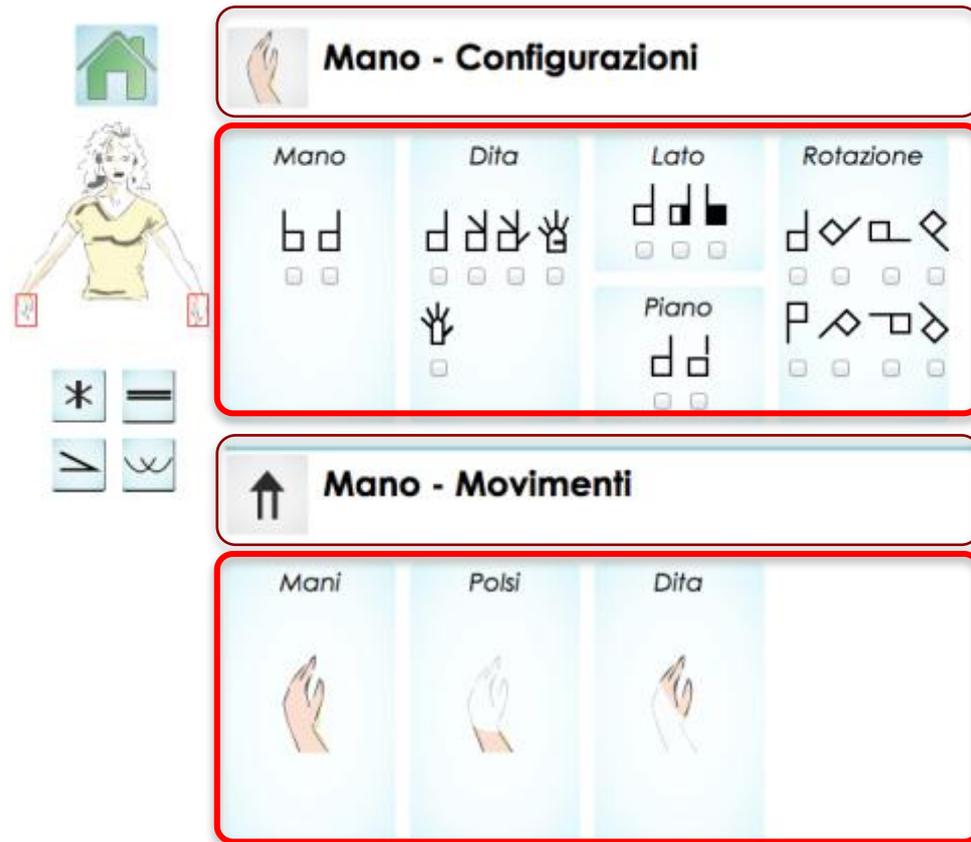
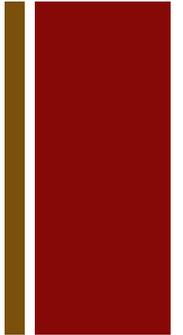
The screenshot displays the SWift interface. On the left, there is a home icon, a user profile icon, and a set of four navigation icons (asterisk, equals, less-than, and greater-than). The main content area is divided into two sections:

- Mano - Configurazioni**: This section is highlighted with a red border. It contains four columns of hand configurations:
 - Mano**: Shows two hand icons with checkboxes below them.
 - Dita**: Shows two hand icons with checkboxes below them.
 - Lato**: Shows two hand icons with checkboxes below them.
 - Rotazione**: Shows two hand icons with checkboxes below them.
- Mano - Movimenti**: This section is also highlighted with a red border. It contains three columns of hand movement icons:
 - Mani**: Shows a hand icon.
 - Polsi**: Shows a hand icon.
 - Dita**: Shows a hand icon.

A label and an icon explain the user what kind of glyphs are available through the group of boxes beneath

+ SWift

Overview

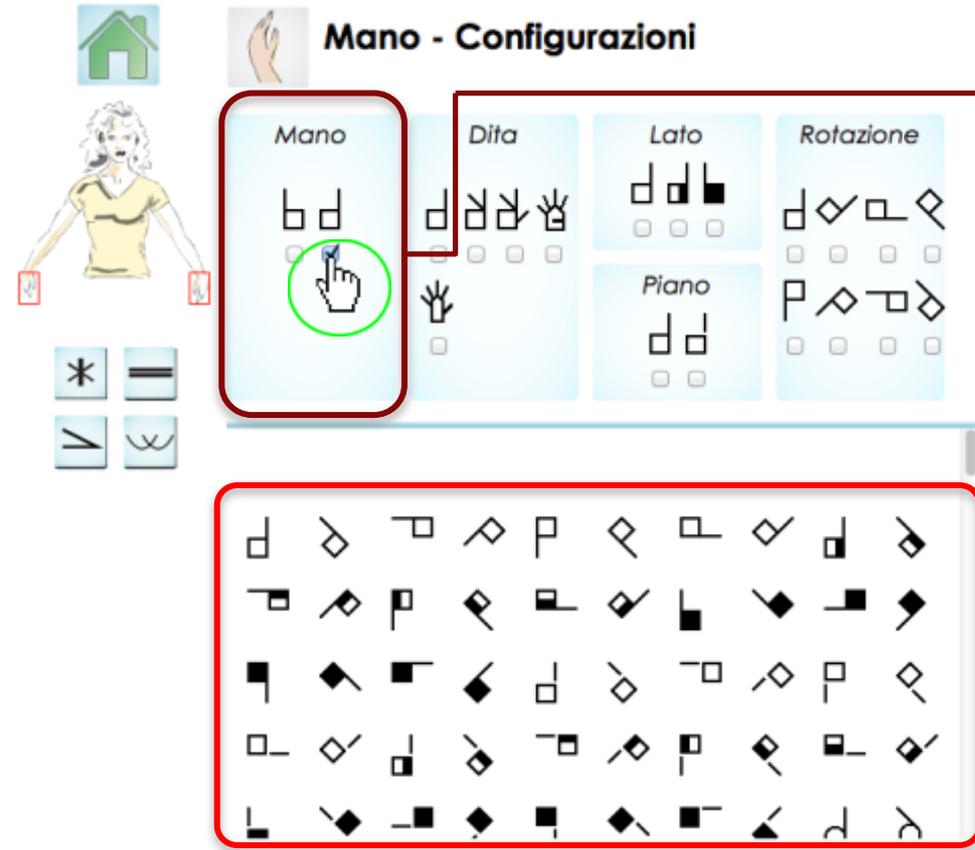
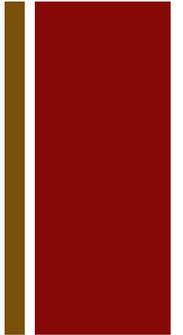


A label and an icon explain the user what kind of glyphs are available through the group of boxes beneath

Choose Boxes:

- **Provide independent choices for a certain glyph feature**
- **The user can use any number of combined criteria**
- **No predefined order of choice**

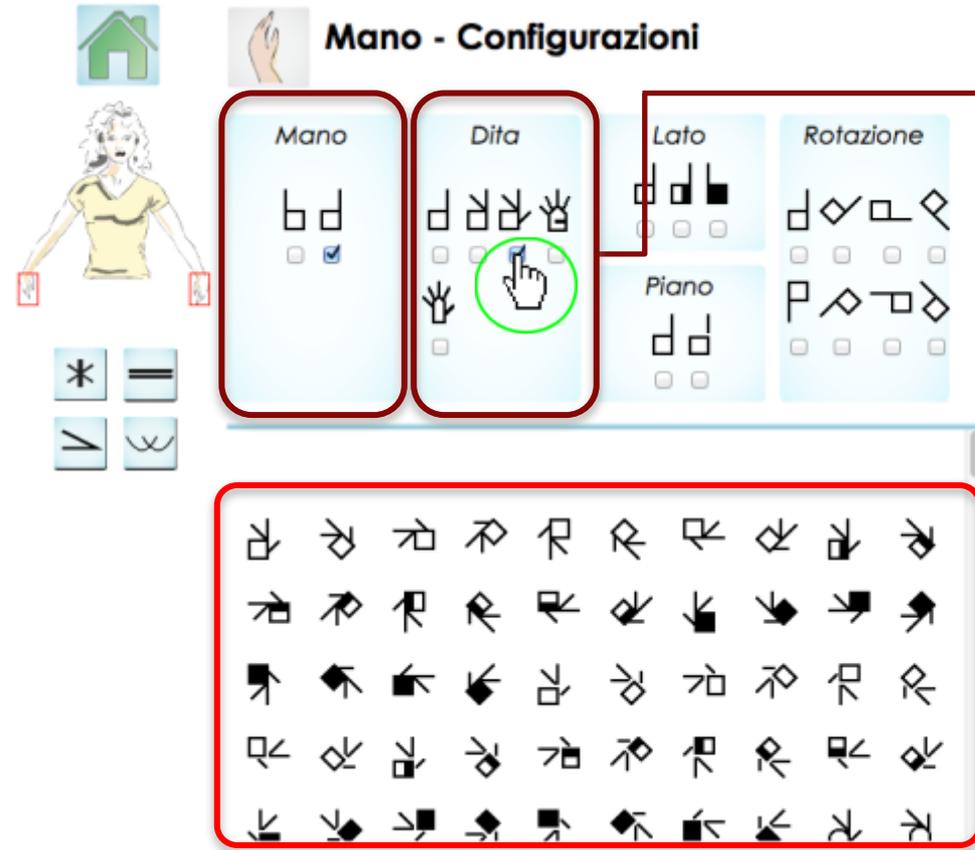
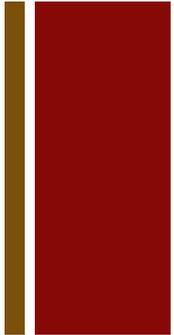
+ SWift Overview



As the user performs a choice...

The set of glyphs meeting the requirements of the user is updated.

+ SWift Overview

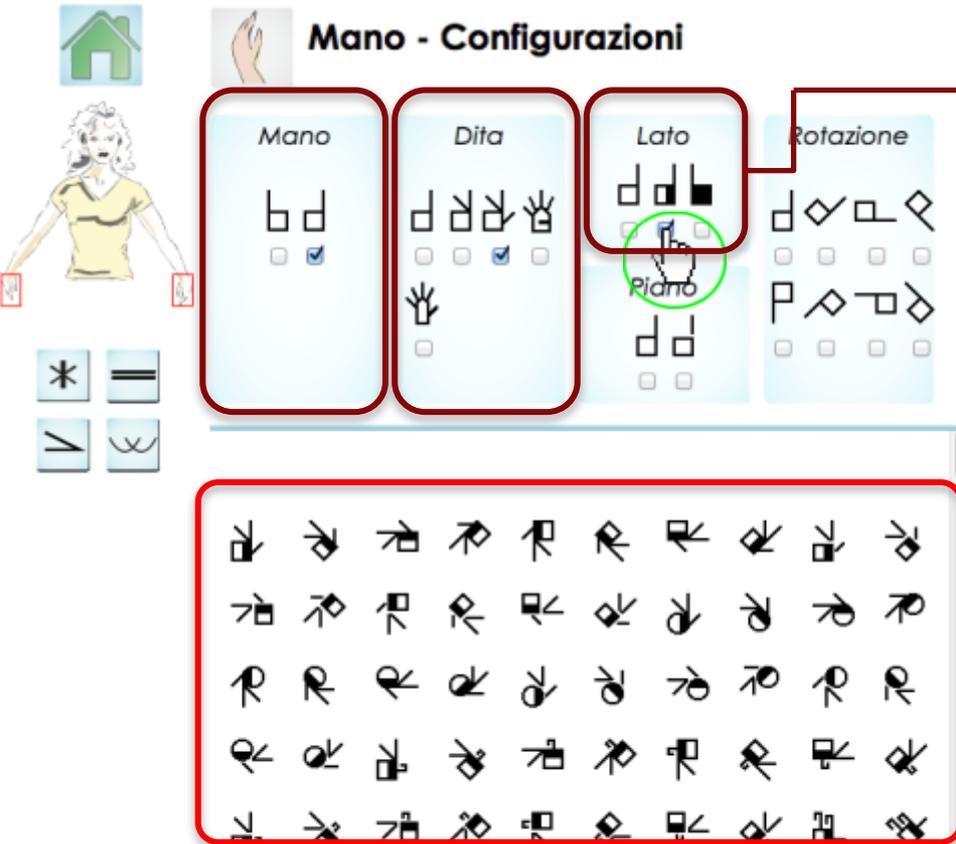
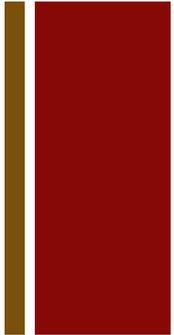


More choices the user makes...

More accurate and restricted becomes the set of result glyphs

+ SWift

Overview



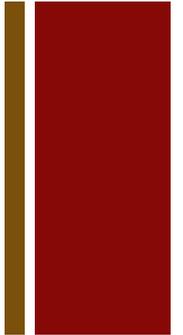
More choices the user makes...

More accurate and restricted becomes the set of result glyphs



SWift

Advanced features

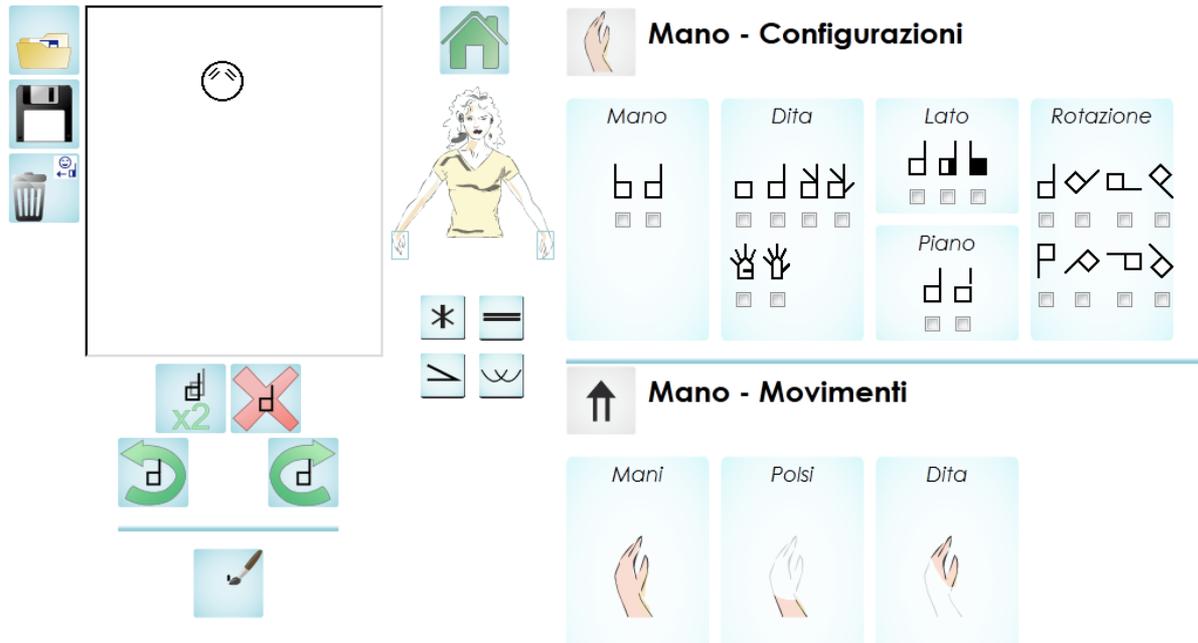
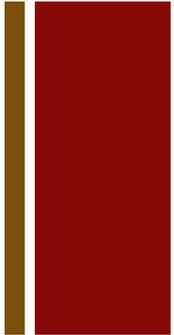


- **Glyph Hint Panel**
 - Shows a set of glyphs that are compatible with those the user inserted in the sign composition area.
 - The glyphs are immediately draggable on the sign composition area.
 - Analysis based on the frequency of co-occurrences of the glyphs in the sign database.
 - Statistics are updated immediately after any sign is saved in SWift format.
 - The proposed glyph exclusively belong to the anatomical area the user is currently searching.
- Sign Hint Panel implemented but untested



SWift

Advanced features

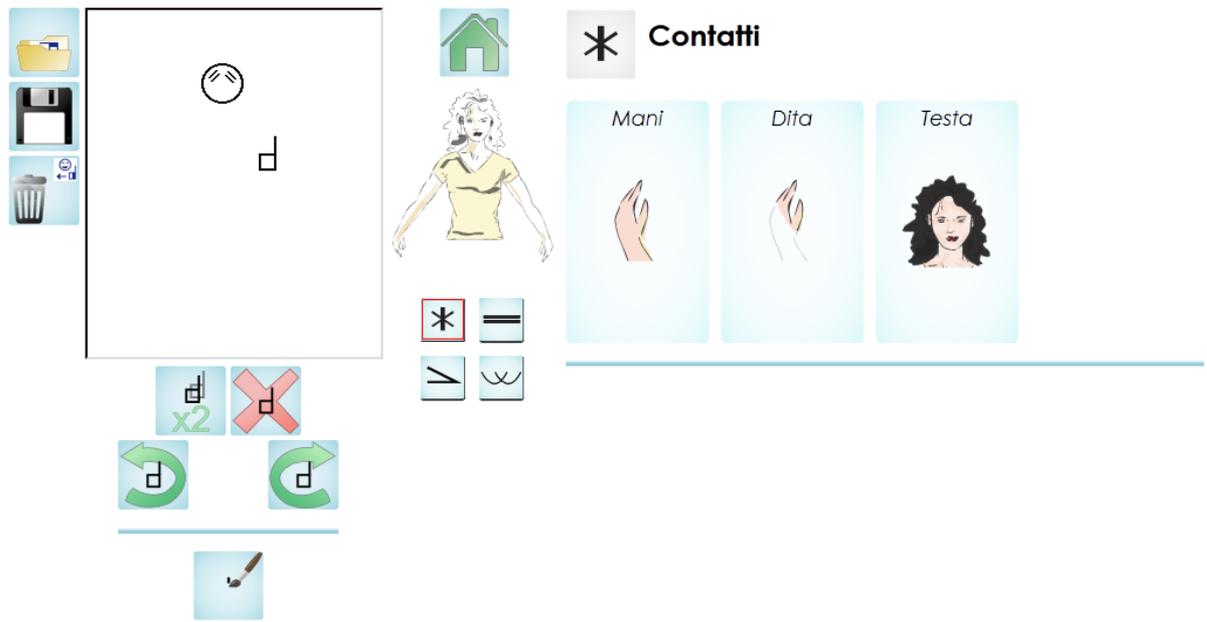
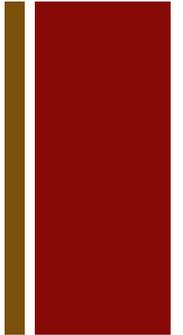


Glyph Hint Panel

- **Compatible glyphs belonging to the current area (hands).**
- **Immediately draggable on the sign composition area**

+ SWift

Advanced features



Glifi Compatibili (2)

* **

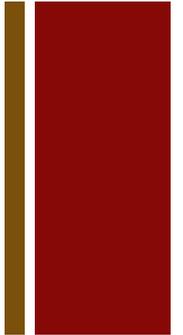
Glyph Hint Panel

- Compatible glyphs belonging to the current area (contacts).
- Immediately draggable on the sign composition area



SWift

Advanced features



- Custom glyphs handwriting support.
- Many SignWriting users tend to “invent” *ad-hoc* glyphs during the writing process
 - This happens when the user is not able to find the glyph he/she is looking for
 - The *ad-hoc* glyphs are consistent with the rules and organization of SignWriting, so they can be understood by other users
- We designed and developed a functionality to support the handwriting of *ad-hoc* glyphs



SWift

Advanced features

- Custom glyphs handwriting support.



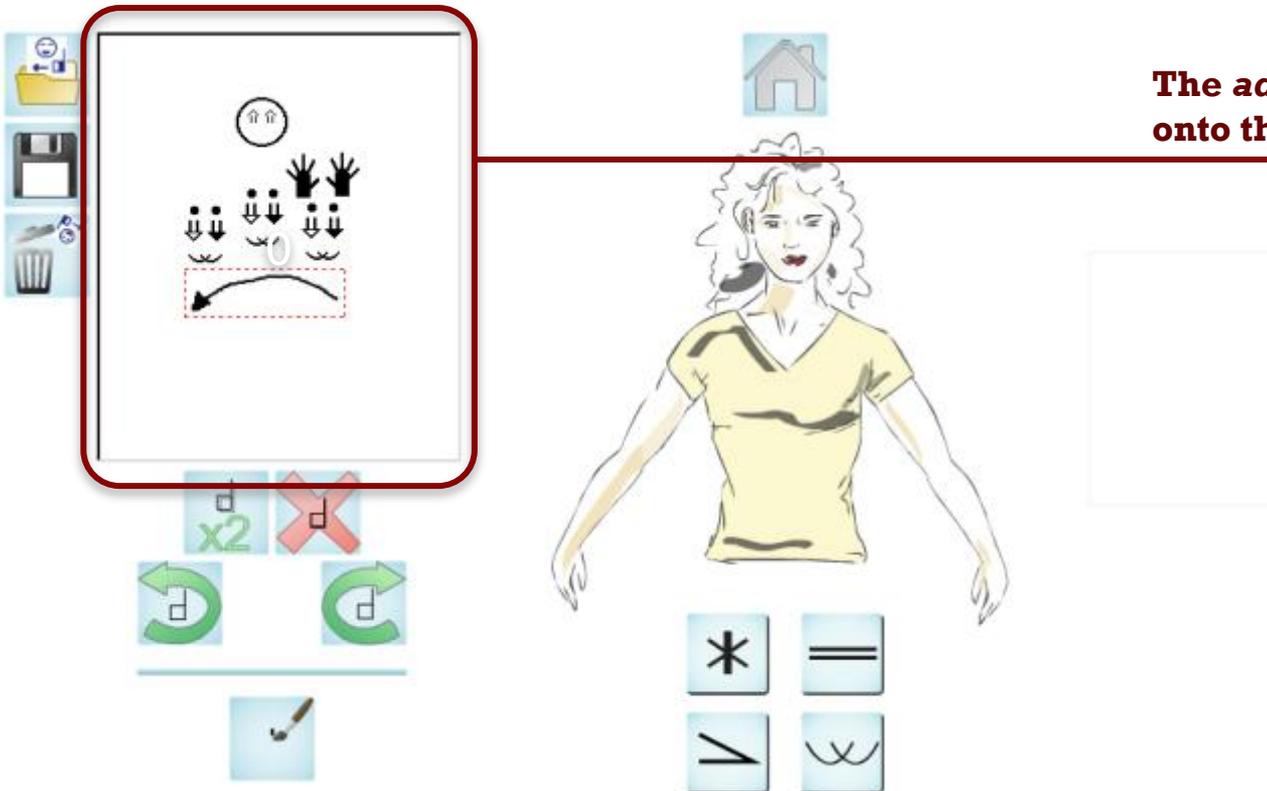
The user handwrites a *ad-hoc* glyph



SWift

Advanced features

- Custom glyphs handwriting support.



The *ad-hoc* glyph is inserted onto the sign composition area

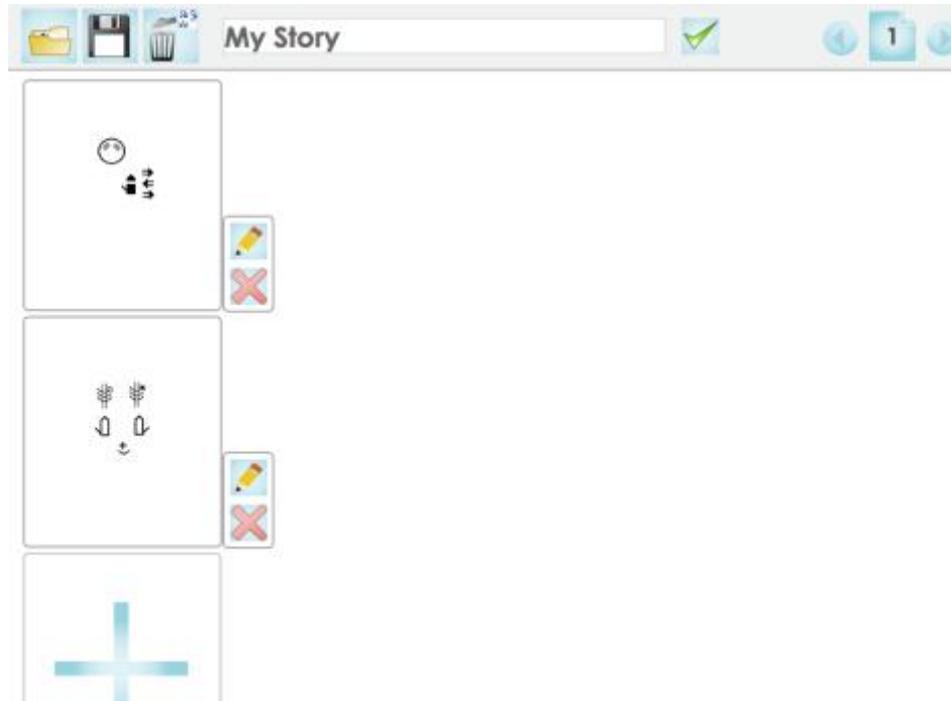


SWift

Advanced features

- Signed Stories Support

- SWift can be used to compose whole signed stories, rather than single signs



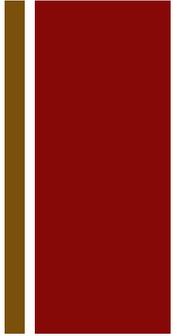


SWift

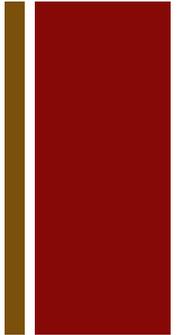
Supporting different ISWA versions

- SWift can support multiple ISWA versions:
 - ISWA2008
 - ISWA2010
 - ISWA Bianchini

- However, the business logic of the glyph search area is based on ISWA Bianchini

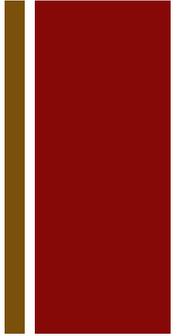


+ SWift Testing



- SWift has been tested with a number of SignWriting-proficient participants
- A deaf-oriented adaptation of the Think-Aloud Protocol (by C. Lewis and J. Rieman) was necessary in order to perform the test
- The devised test was named “Think by Signs” protocol
- The results of the test underlined a very good usability level
- Critical areas of intervention were identified within the application

+ SWift Testing

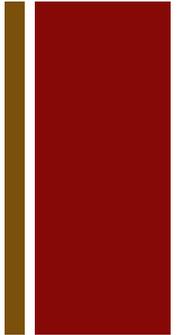


For more information about the testing of SWift, and for detailed result, please refer to:

Bianchini, C.S., Borgia, F., Bottoni, P., De Marsico, M. (2012).
SWift - A SignWriting improved fast transcriber.
In *Proceedings of AVI2012* (Capri, 21-25 May 2012)

Bianchini, C.S., Borgia, F., De Marsico, M. (2012).
SWift - A SignWriting editor to bridge between deaf world and e-learning.
In *Proceedings of ICALT2012* (Rome, 7-10 July 2012)

+ Conclusions and Future



- SWift is a new digital editor with novel capabilities with respect to other digital editors
- SWift is framed within the SWORD (SignWriting Oriented Resources for the Deaf) project
- SWORD will represent a step towards full integration of deaf people in digital society

SWift, a user-centered digital editor for SignWriting within SWORD project

Fabrizio Borgia¹, Claudia S. Bianchini², Maria De Marsico¹

¹ fabrizio.borgia@uniroma1.it / demarsico@di.uniroma1.it - Dept Computer Science, "Sapienza" University of Rome, Via Salaria 113, 00198 Rome, Italy

² claudia.savina.bianchini@univ-poitiers.fr - UFR Lettres et Langues / EA3816 FoReLL, Université de Poitiers, 1 rue R. Cantel, 86073 Poitiers, France

Since the early years of SignWriting (SW), both the research and the deaf community acknowledged its importance for digital communication. The applications designed to support its use range from the first "SignWriter Computer Program" (1986) to the modern digital editors [1].

Heterogeneous requirements must be taken into account: the large amount of symbols requires a robust yet intuitive organization of the search space; the organization and rules of the International SignWriting Alphabet (ISWA) must be considered too; finally, both a user-friendly navigation context for finding and fetching symbols (glyphs), as well as a space to compose the desired sign, have to be provided; an often underestimated key requirement, underlying the whole design and implementation processes, is an optimal usability level of the produced application.

The present work introduces SWift (SignWriting improved fast transcriber) [2] - a SW digital editor designed, developed and tested in close cooperation with the ISTC-CNR research team, mostly composed of deaf people. We followed the principles of User-Centered and Collaborative Design to produce an application for, and with, the Deaf Community.

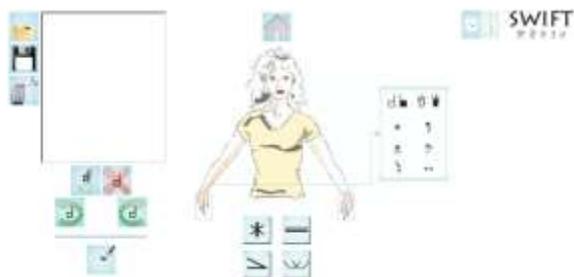


Fig. 1: Home screen of SWift, the hand symbol area being highlighted by the user

SWift provides all the functions of any SW digital editor, plus some more. The symbol fetching system was designed anew, in order to fit the needs of SW-beginners too. The system leads the user to the desired symbol category (head, hands, etc.) using a stylized human puppet (Fig. 1) as starting point. Instead of browsing the whole symbol space, SWift allows the user to search for the desired symbol through an iterative reduction of the answer set of symbols, obtained by refining the searched features (e.g., for the hand, side, orientation, etc.). The system does not force to choose such features in a predetermined order, but rather allows deciding the one to start with. For instance, one user may decide to begin looking for a specific hand symbol by entering the number of fingers, while another may decide to enter the hand orientation first.

The user interface (UI) has been thoroughly designed. To make the composition process easy and fast, we minimized the use of text elements, and avoided UI cluttering.

SWift allows the user to save the created signs in multiple formats: as a text file (XML) - via the SWML [1] format; as an image file (PNG); and remotely – storing the sign name and the association with its component glyphs in the SWift's own database. It is also possible to store phrases and longer stories. Appropriate statistics computed on the database, whose reliability improves over time, allow to include an automatic suggestion engine, which analyses in real-time the sign that the user is composing and suggests likely glyphs to include next, or even a complete sign, therefore relieving the user from the burden of starting a whole new search.

SWift is designed to use different ISWA versions as symbol base. At present, SWift adopts a revised and enhanced version of the ISWA, following Bianchini [3].

One key stage in the life cycle of any software is the usability testing phase, which was one of our main concerns. Basing our efforts on the work of [4], we designed the Sign-Along Protocol [5] - a deaf-oriented usability test protocol based on the well-known Think-Along Protocol. The test results show an acceptable usability of the application and they are summarized in the present work.

[1] Sutton V. (n.d.). SignWriting for Sign Languages. retrieved from <http://www.signwriting.org/> on 13/11/2013.

[2] Bianchini C.S., Borgia F., Bottoni P., De Marsico M. (2012). SWift - a SignWriting improved fast transcriber. *in Proc. AVI2012 (Capri, 21-25/05/2012)*.

[3] Bianchini C.S. (2012). Analyse métalinguistique de l'émergence d'un système d'écriture des Langues des Signes: SignWriting et son application à la Langue des Signes Italienne (LIS).Ph.D. thesis, Université de Paris 8 - Università degli Studi di Perugia.

[4] Roberts V.L., Fels D.I. (2006).Methods for inclusion. *Int. J. Human-Computer Studies*, 64(6): 489-501.

[5] Bianchini C.S., Borgia F., De Marsico M. (2012). SWift - a SignWriting editor to bridge between deaf world and e-learning. *in Proc. ICALT2012 (Rome, 07-10/06/2012)*.