Accessibility as a Service: Augmenting Multimedia Content with Sign Language Video Tracks

Tiago Maritan U. de Araújo, Felipe Lacet S. Ferreira, Danilo Assis Nobre S. Silva, Eduardo de Lucena Falcão, Leonardo Dantas de Oliveira, Leonardo de Araújo Domingues, Yurika Sato Nóbrega, Hozana Raquel G. Lima, Alexandre Nóbrega Duarte and Guido Lemos S. Filho

Digital Video Application Lab, Federal University of Paraíba, Cidade Universitária, João Pessoa – Paraíba, Brazil
{maritan, lacet, danilo, eduardolf, leodantas, leonardo.araujo, yurika, hozana, alexandre, guido}@lavid.ufpb.br

In this paper we explore the concept of “accessibility as a service” by proposing a cloud computing service to help deaf people to access digital contents. The proposed service automatically generates and embeds a sign language video track into multimedia contents. The service organizes the collaboration of sign language experts to dynamically adjust the system that runs on a cloud computing infrastructure. As a case study, we made an implementation of the service for providing support for the Brazilian Sign Language (LIBRAS) and some preliminary tests with Brazilian deaf users to evaluate the proposed solution.

Keywords: accessibility; sign language; machine translation; multimedia contents; cloud computing; sign animation.

ACM Classification and Subject Descriptors: I.2.7 (Natural Language Processing): Machine Translation; H.3.5 (On-line Information Services): Web-based services; H.5.1 (Multimedia Information Systems): Animations; H.5.3 (Group and Organization Interfaces): Collaborative computing.

1. Introduction

Deaf people have serious difficulties in accessing information. Sign languages are their primary means of communication, but information and communication technologies (ICTs) rarely provide support for them. In addition, it is difficult for deaf people to understand and communicate by using texts in spoken languages. For example, some reading comprehension tests performed by Wauters (2005) with deaf children aged 7–20 in the Netherlands showed that only 25% of them read at or above the level of a nine-year-old hearing child. In Brazil, about 97% of the deaf people do not finish high school (IBGE, 2000).

There are several works in the scientific literature developed to address their communication limitations (Lee et al., 2007; Lee et al., 2005; Starner et al., 1998). These works offer technological solutions to daily activities, which enable deaf people to watch and understand television (Lee et al., 2007), to interact with other people (Lee et al., 2005; Starner et al., 1998), among others. Others’ works are related to the use of machine translation methods to translate spoken languages to sign languages (SLs) (Veale et al., 1998; Zhao et al., 2000; Morrissey, 2008; San-Segundo et al., 2008; Gallo
et al, 2009). Veale et al (1998), for example, described a multilingual translation system used for translating English texts into Japanese Sign Language (JSL), American Sign Language (ASL) and Irish Sign Language (ISL).

Zhao et al (2000) developed an interlanguage-based approach for translating English text into ASL. Input data are analyzed and an intermediate representation (IR) is generated from their syntactic and morphological information. Then, a sign synthesizer uses the IR information to generate the signs. Morrissey (2008) proposed an example-based machine translation (EBMT) system for translating text into ISL. To do this task, an EBMT approach needs a bilingual corpus. However, due to the lack of a formally adopted or recognized writing system for SL, it is hard to find corpora in SL. Thus, the authors generated their bilingual corpus by using annotated video data.

San-Segundo et al (2008) and Gallo et al (2009) proposed an architecture for translating speech into Spanish Sign Language (LSE) focused on helping deaf people when they want to renew their identity card or driver’s license. This translation system is composed of three modules: a speech recognizer, a natural language translator and an animation module. The speech recognizer is used for decoding the spoken utterance into a word sequence. The natural language translation module converts the word sequence into a sequence of signs in LSE and the animation module plays the sign sequence.

There are also other works that specify formal representations (Fusco, 2004), languages to generate SL animations (Kaneko et al, 2010), dictionaries (Buttussi et al, 2007) and recognition (Starner et al, 1998) in SL.

In this paper we explore the concept of “accessibility as a service” by proposing a collaborative service that is able to automatically generate (i.e., without the direct interference of an interpreter) and embed a sign language video track into a regular multimedia content, improving its presentation to address needs of deaf people. The sign language video is generated from the subtitle tracks of the multimedia content. The service is collaborative because sign language experts can collaborate to improve and evolve it over time. The service has some mechanisms that allow collaborators to improve the quality of translation and presentation (e.g., by editing or adding the translation rules and the signs of the dictionary). In addition, the service runs on a cloud computing infrastructure, which allows the service to be scalable and optimize the use of computational resources.

The rest of this paper is organized as follows. In Section 2, we review the main concepts about sign languages. In Section 3, we describe the proposed service. In Section 4, we describe a strategy to deploy our solution on a cloud computing infrastructure. In Section 5, we describe an implementation of the service for the Brazilian Sign Language (LIBRAS). Some tests to evaluate the proposed solution are described in Section 6. Final remarks are given in Section 7.

2. Sign Languages

Sign languages are visual languages used by deaf people as their primary means of communication (Wang et al, 2007). According to Brito (1995), they are considered natural languages, because they emerged from the interaction between deaf people and they can express any descriptive, concrete, rational, literal, metaphorical, emotional or abstract concept.

Like the spoken languages, each sign language has its own grammar, containing several linguistic levels, such as morphology, syntax and semantics (Brito, 1995). They also have lexical items that are called signs (Stokoe, 1980). The signs differ from each other in their visual-spatial use. Another
difference is related to the structure of the languages. While spoken languages have a sequential structure, i.e., phonemes are produced sequentially in time, SLs have a parallel structure and can produce signs using several body parts at the same time.

The signs have basic units called phonemes. According to Stokoe (1980), two different signs differ by at least one phoneme. Handshape (i.e., the position of fingers and their movement), location (i.e. the part of the body where the sign starts to be performed), hand movements and facial and/or body expressions (non-manual features – NMF) are examples of phonemes.

3. The proposed service

In this section we briefly describe the architecture of the proposed service (see Figure 1). As mentioned earlier, the proposed service automatically generates and embeds a SL video layer into regular multimedia contents. The service generates the SL video track from subtitles of the multimedia content.

An important aspect of our proposal is the use of collaborative strategies to improve/evolve the quality of translation and presentation of the service over time. The idea is SL specialists can help
to improve the service, for example, through the improvement of the translation rules, including new signs, etc. To do this task, we define languages to describe translation rules (the Rule Description Language) and signs (the Sign Description Language), and a collaborative tool to handle them (the WikiSL tool).

According to Figure 1, the service works as follows. Initially, the submitted multimedia content is applied for a Filtering component, which extracts the subtitles track. Afterwards, a Subtitle Extraction component converts this subtitle stream into a sequence of words in the source-spoken language. Then, this sequence of words is automatically translated (using the Machine Translation component) to a sequence of words in the target SL (i.e., a sequence of glosses), according to a set of translation rules. The sequence of glosses is then sent to an Exhibition component that associates each gloss with a visual representation of a sign stored in a SL Dictionary. Thus, the sequence of glosses is mapped to a sequence of visual representations that will be synchronized with the subtitle track to generate the SL video. Finally, the SL video is embedded in the regular multimedia content as an extra video layer (by Embedding component), making it accessible to the deaf.

Synchronization between the regular multimedia content and the SL video is performed based on axes-based synchronization model (Blakowski and Steinmetz, 1996). It defines synchronization points inserted in the content using timestamps based on a global timer. In our solution, the global timer is the reference clock of the subtitle track. This clock is extracted from the subtitle track and is used to generate the presentation timestamps for the signs of the SL video. More details about the components of the service are presented in the following subsections.

3.1 Machine Translation Component

The Machine Translation component converts a textual representation in the source spoken language to a textual representation (sequence of glosses) in the target SL. The Machine Translation schematic view is illustrated in Figure 2.

![Figure 2: Schematic view of the machine translation component](image-url)
According to Figure 2, initially the text in the source spoken language is split into a sequence of words or tokens (tokenizer module). Afterwards, the tokens are classified into morphological-syntactic categories (Morphological-syntactic Classifier module). To do this task, we use the PPM-C algorithm (Moffat, 1990), that is a variant of Prediction by Partial Matching (PPM) (Cleary and Witten, 1984). PPM is a statistical data compression method based on N-order Markov models. It was chosen due to its ability to build accurate statistical models (Batista and Meira, 2004) and its previous use in other classification problems (Bratko et al, 2006; Mahoui et al, 2008; Medeiros et al, 2011).

PPM builds a statistical model and uses it to store the frequency of different sequences of elements found. After the model is built, the next element of the sequence can be predicted according to its previous N elements. The PPM-C variant is more efficient than the original implementation in terms of run time and data space in exchange for marginally inferior compression.

The Morphological-syntactic classifier models morphological and syntactic classes as elements in PPM-C. This model stores sequences of morphological-syntactic classes taken from a corpus of morphological-syntactic classified texts in the source spoken language. Once a sentence is received for classification, the most likely morphological-syntactic class of each token is selected according to its context in PPM model.

After classifying the tokens, we apply some translation rules to translate these tokens (or words) for a representation in gloss notation. The translation rules are loaded from a Translation Rules database and are described using a defined language, called Rule Description Language. We describe the Rule Description Language in details in Section 3.4.

3.2 Exhibition Component and SL Dictionary

As mentioned earlier, the Exhibition component is responsible for converting the sequence of glosses generated by the Machine Translation component into a SL video track. In order to perform this task, the Exhibition component uses a SL Dictionary, which contains a visual representation (e.g., an animation or video file) for each sign.

Formally, the SL Dictionary can be defined as a set of t tuples in the following format:

\[ t = \langle g, v \rangle \]

where:

• g is the gloss (or code) of the sign;
• v is the visual representation of the sign.

Thus, the Exhibition component gets the sequence of glosses, associates each gloss with its visual representation (according to the SL Dictionary) and generates the SL video from these visual representations. Since these visual representations of signs are completely independent, providing smooth transition between consecutive signs is not a trivial task. A neutral configuration (i.e., position, background color, brightness, etc.) was defined to be applied at the beginning and the end of each sign as well as during silent intervals.

Additionally, the Exhibition uses the global timer and the timestamps of the subtitle track (i.e., the Sync. info in Figure 1) to generate the correct presentation timestamps of the signs in the SL video track, synchronizing it with the input multimedia content.

As well as the translation rules, the signs of the SL Dictionary can also be improved over time. To do this task, we developed a Sign Description Language, which allows deaf and SL specialists to describe signs. From this description, the signs can be rendered by the WikiSL tool using a 3D-
virtual agent model (a model of a 3D-avatar). The Sign Description Language will be described in details in Section 3.4.

3.3 Embedding Component

After the generation of the SL video, the service embeds the SL video (generated by the previous components) in the input multimedia content, adding sign language content to it. This task is performed by the Exhibition component using one of the following ways:

1. Mixing the SL video: In this case, the frames of the SL video are displayed in a window over the frames of the multimedia content, making the Accessible video (original multimedia content + SL content) independent of the video player. However, one problem of this approach is that after the mixing process is performed, it is no longer possible to disable or remove the SL video window.

2. Multiplexing the SL video: In this case, the SL video is coded as a separate and independent video stream and encapsulated together with the input multimedia content into a single encapsulated transport stream, based on the MPEG-2 Transport Stream (ISO/IEC 13818-1, 2000) standard. This way we have one single Transport Stream containing two video tracks. This approach makes the Accessible video dependent on players that need to be able to play both videos at the same time. On the other hand, it makes it possible to enable, disable, resize or relocate the SL video.

3.4 Rule and Sign Descriptions Language

3.4.1 Rule Description Language

As mentioned earlier, the Rule Description Language is used to describe the translation rules that will be applied by the machine translation component. In this language, each translation rule is defined as an $r$ tuple in the following format:

\[
r = < e_0, e_1, \ldots, e_c >,
\]

where $e_0, e_1, \ldots, e_c$ is a set linguistic elements ordered according to the input sequence and $c$ is the number of linguistic elements. The linguistic elements $e_i$ are defined as

\[
e_i = < ms_{class}, n_{pos}, n_{prop} > , \quad i = 1, \ldots, c,
\]

where $ms_{class}$ identify it morphologically or syntactically class. $n_{pos}$ indicates the new positioning of the element after the rule is applied with a value of “-1” meaning that the element must be removed. $n_{prop}$ is an optional field which indicates possible changes in the element (e.g., every verb in LIBRAS must be in the infinitive form).

Based on these definitions, we specify a XML representation to represent the attributes of the rules defined above. Figure 3 illustrates an example of the XML representation of a rule. It indicates that whenever a sequence of a verb followed by a substantive and preposition is found, the words in the translated text should be rearranged so that the preposition would come first, followed by the substantive and the verb.

As XML representations are usually not intuitive for SL specialists, the WikiSL tool should provide a friendly user interface to assist them in the definition of new rules or in the edition of existing rules from the Rule Description Language. In Section 3.5, we will describe the WikiSL tool in details.
3.4.2 Sign Description Language

The Sign Description Language is used to describe the signs that will make the SL Dictionary. From this description, the WikiSL tool can render a visual representation of the sign based on a 3D-virtual agent model (a model of a 3D-avatar).

In this language, a sign can be defined as a set of movements, where each movement has an initial and final configuration of hands, arms and face, a type of trajectory (e.g., rectilinear, circular, semicircular, etc), a direction (e.g., from left to right, from inside to outside, etc) and flags to indicate which hands are used in the movement (e.g., left, right or both).

Formally, we define a sign $s$ as follows:

$$s = <gl, m_1, ..., m_n>, \quad \text{where}$$

$$m_i = <c_{f_{ini}}, c_{f_{fin}}, tr, dir, lhf, rhf>, \quad i = 1, 2, ..., n,$$

$$c_{f_{t}} = <hs_l, hs_r, or_l, or_r, loc_l, loc_r, fe>, \quad t = ini, fin,$$

Where $gl$ is the gloss of the sign and $m_1, ..., m_n$ are the set of movements. $c_{f_{ini}}$ and $c_{f_{fin}}$ are the initial and final configuration of each movement ($m_i$), respectively. $tr, dir$ are the trajectory and direction, respectively. $lhf, rhf$ are flags which indicates whether the left and right hands are used in the movement, respectively. $hs, or, loc, fe$ are the handshape, orientation of palm (e.g., facing up, facing down, facing in, etc.), location and facial expression phonemes of each configuration. The $l$ and $r$ indexes of $hs, or, loc$ phonemes indicate whether that phoneme refers to left or right hand, respectively.

Based on these definitions, we also define a XML representation to specify the signs. Figure 4 (on the following page) illustrates an example representation of LIPS sign in Brazilian SL (LIBRAS).

3.5 WikiSL Tool

As mentioned earlier, the WikiSL tool allows that collaborators improve the quality of the service over time through adding new translation rules and signs or editing existing ones. Figure 5 illustrates a schematic view of WikiSL.
According to Figure 5, initially the collaborators access the collaborative environment through a Web interface. Then, the users can configure new rules or signs or search in the signs or rules (for possible editing of them). When the user sets up a new translation rule, the Rule Description Generator module converts the user interaction in a XML representation, according to the Rule Description Language. This XML representation is then stored in a temporary database to be approved by SL specialists (i.e., a supervision stage is applied over it), which prevent the addition of incorrect rules in the Translation Rule database. Upon approval, the new rule is added to it. In addition, users can search on existing rules and propose edits on them.

When the user sets up a new sign, a Sign Description Generator module converts user interactions in a XML representation according to the Sign Description Language. Afterwards, this representation is converted by the Parser module to a set of parameters of the 3D-agent virtual model (see Section 3.4.3) and an animation is rendered (by the Render module) from these parameters. The generated animation is then returned to the user, so that he can validate the animation. Upon user approval, the animation is also sent to the temporary database to be supervised by specialists.
4. Integration into a Cloud Computing Infrastructure

In order to have the best relation between a number of simultaneous users and the cost of deployment and maintenance, we decided to deploy our solution on a cloud computing infrastructure, exploring its high scalability, dynamic provisioning and easy access (Zhang et al., 2010). The main idea is to explore the computational resources provided by these infrastructures (more specifically, an IaaS – Infrastructure as a Service – provider) to allow the simultaneous execution of multiple instances of the service, making it scalable to an increasing number of users, without dealing with the costs of acquiring and maintaining the hardware. Figure 6 illustrates the model we adopt to deploy our solution on the cloud.

According to Figure 6, users access the service through a Web interface and submit their multimedia content for processing. This processing request is received by a component called Controller, which acts as a broker, responsible for scheduling the user requests over a set of processing machines. Once receiving a request the Controller checks if the current system load is below a predefined limit. If so, it schedules the execution for one of the virtualized machines provided by the IaaS provider. If the system load is already too high the Controller requests a new machine from the IaaS provider and proceeds with the scheduling as described before. The load distributing follows a round-robin strategy, where the work is assigned to the machine with the smallest number of instances running.

Finally, whenever a processing task is concluded, the Controller checks if the system load is below a predefined limit. If so, it terminates idle machines until the load rises above the limit.

5. A Brazilian Sign Language Case Study

In this section, we describe the implementation of the proposed service for the Brazilian Sign Language (LIBRAS).
LIBRAS is the SL used by most of the Brazilian deaf and is recognized by law as the official Brazilian SL. The signs in LIBRAS are composed of five phonemes: handshape, locations, hand movements, orientation of palm and NMFs. The possible values for each of these phonemes are discussed in Fusco (2004). LIBRAS also has its own vocabulary and grammar rules, which are different from the Brazilian Portuguese (BP).

5.1 Implementation of the 3D-virtual agent model
The 3D-avatar model was developed using Blender software (www.blender.org) with an armor composed of 82 bones distributed as follows:
- 15 bones in each hand to setup handshape;
- 23 bones on the face to setup facial expressions and movements;
- 22 bones in arms and body to setup arm and body movements and
- 7 auxiliary bones (i.e., bones that do not deform the mesh directly).

Figure 7(a): The 3D-virtual agent model, (b): Emphasis on bones of face, (c): hand and (d): body
Thus, to configure the movements of the fingers, for example, it is necessary to define parameters of location and rotation for each of these 15 bones. The same should be done to the bones of the face of the avatar. The arm movement is performed by moving only two bones. The first one is located on the pulse of the avatar and the second one is an auxiliary bone which controls the deformation of the elbow and forearm. We use inverse kinematics to relate the deformation between bones related. Thus, if there is a movement in the wrist bone, for example, it will spread to the bones of the arm and forearm.

The 3D-avatar model (with all bones) is illustrated in Figure 7a. Figures 7b, 7c and 7d illustrate the emphasis on the bones of the face, hand and body of this 3D-model, respectively.

5.2 Implementation of main components

The elements responsible for Filtering, Subtitle Extraction and Synchronization as well as the Machine Translation, Exhibition and Mixing components were implemented using the C++ programming language. In this version of the service, we use videos in the MPEG-2 Transport Streams (MPEG-2 TS) format subtitled in Brazilian Portuguese (BP) as input. The Filtering and Subtitle Extraction components were developed based on the definitions of the MPEG-2 System (ISO/IEC 13818-1, 2000) and ABNT NBR 15606-1 (2007) specifications, respectively. The MPEG-2 System defines how to extract the different elementary streams of different content (e.g., audio, video, closed caption), and the ABNT NBR 15606-1 defines how to extract text from closed captions in these contents.

The machine translation component receives the BP sentences and translates them to a sequence of glosses in LIBRAS, according to Figure 2. The Morphological-syntactic classifier module was developed based on a corpus in the Portuguese language, called “Bosque”. This corpus was developed by Syntactic Forest project (Freitas et al, 2008) and has 9,368 sentences and 186,000 words, obtained from “Folha de São Paulo” (www.folha.uol.com.br) Brazilian newspaper and “Público” (www.publico.pt) Portuguese newspaper. The sentences were morphologically and syntactically classified and reviewed by linguists. In our case, we use only the Brazilian Portuguese sentences of the corpus.

From the BP Bosque sentences, the PPM-C algorithm is applied to classify morphologically and syntactically the tokens. The Markov order defined empirically for the PPM model was 5. This value was chosen in order to maintain a good threshold between accuracy and run time. Finally, the translation rules (described with the Rule Description Language) are applied to translate the tokens to a sequence of glosses.

The Exhibition component receives the sequence of glosses from Machine Translation and looks for their visual representation in the SL dictionary to compose the SL video. The signs of the SL Dictionary were generated from the implementation of the 3D-avatar described on Section 5.1. A neutral configuration is applied at the beginning and the end of all signs of the SL Dictionary. The neutral position was defined according to the suggestion of LIBRAS interpreters, placing the hands and arms extended in a straight line down and with a neutral facial expression (i.e., without applying movement in the facial bones). We also developed a “neutral sign” with the 3D-avatar in the neutral configuration during 1 second. The Exhibition component repeatedly displays this sign on silent intervals between the beginning of the video and the first subtitle as well as silent intervals between two consecutive subtitles.

In order to generate synchronized sign language content, the Exhibition component extracts the first Program Clock Reference – PCR, (the global timer) of the original video. This clock reference
Accessibility as a Service: Augmenting Multimedia Content with Sign Language Video Tracks

is set as initial clock of the SL video. The timestamps of all signs of the SL video are generated based on the PCR and the presentation timestamps (PTS) of the related closed caption. In the end of the processing, the Exhibition component generates a complete SL video stream.

The Mixing component receives the original multimedia content and the SL video, providing parameters as size and position to customize the superposition. To be overlayed in a synchronous way, the first step is to verify the fps (frames per second) rate of the videos and equating them in case of difference. After this task, the videos are transcoded and mixed through system calls, using ffmpeg (ffmpeg.org) to abstract low level details. As a result of this process, the component generates a new video file, where the SL video is overlayed on the original multimedia content, making it accessible.

Figure 8 illustrates two screenshots of the accessible multimedia content generated by the implementation of the service. See that the generated SL video is displayed over the user content (a part of a movie).

5.3 Implementation of WikiSL Tool

The Web interface of the WikiSL tool was developed using PHP programming language with the aid of AJAX (developers.sun.com/scripting/ajax/index.jsp) and jQuery (jquery.com) technologies to validate the interface and make it dynamic. The Rule and Sign Description Generator modules, responsible for the generation of the XML representation of the rule or the sign, were also developed in the PHP programming language.

The manipulation of the 3D-avatar model described in Section 5.1 is done automatically using scripts developed in Python programming language. These scripts are responsible for interpreting the intermediate language, configure the phonemes and render the signs using libraries of pre-recorded poses. For the configuration of hand shape, location, and facial expression, we developed a set of poses libraries, which have the coordinates of location and rotation of the bones used in each pose. For each facial expression, for example, you must configure the rotation and locations of the 23 bones on the face of the 3D-avatar model.

Figure 9 illustrates some screenshots of the WikiSL tool. Figures 9a and 9b illustrate the configuration of some parameters of a new sign. Figure 9c illustrates the configuration of a new rule. Figure 9d illustrates actions of searching and editing an existing rule.
5.4 Integration into an IaaS provider

To evaluate the integration of our service implementation into a cloud computing infrastructure, we decided to use the Amazon Elastic Compute Cloud (Amazon EC2 – aws.amazon.com/ec2/) with support from Amazon’s AWS in Education Research Grant, which funded the instances we used for the experiments.

Amazon’s EC2 provides general purpose virtual machines, enabling cloud users to have full control over server instances in remote data centers.

We deployed our solution using the EC2 Extra-Large instances, which costs U$ 0.68 per hour and provides 15GB of memory, eight EC2 compute units (four virtual cores with two EC2 Compute units per core) and 1.690 GB of disk space.

Figure 9: Some screenshots of the WikiSL tool
(a) Setting some parameters of a new sign: handshape, facial expression (b) Orientation and location of a new sign (c) Setting a new rule; (d) Searching and Editing an existing rule
The Controller component was implemented in the C++ programming language to manage a set of \( n \) instances in the Amazon EC2. To do this task, it uses a vector with \( n \) elements, where each element represents a counter of the number tasks running on each EC2 instance. Whenever a user makes a new request, the Controller creates a new task to run the service in the EC2 instance with the smallest number of tasks already running, updating the vector accordingly.

6. Results and Discussion

In order to perform a preliminary evaluation of the proposed solution, we performed a set of tests with the prototype of the service described in Section 5. These tests include some objective measures and a subjective evaluation with Brazilian deaf users.

6.1 Objective evaluation with Amazon EC2 instances

Initially, we performed some tests to obtain objective measures of system performance and scalability. In these tests, different service demands were simulated (running the solution in the Amazon EC2 instances), and the average response time and cost was measured using two different scenarios. In the first scenario, the average response time and cost was calculated using a single EC2 instance and all service requests were directed to it (sequential approach). In the second scenario, the average response time was calculated using up to 10 EC2 instances and the requests were distributed by these instances according to the approach described on Sections 4 and 5.2 (our distributed approach).

The simulations were performed using a uniform request, where the input is a high-definition video with BP closed captioning. This video is in MPEG-2 TS format, has a frame rate of 45 fps (frames per second) and lasts 27 seconds. Each simulation was repeated three times for the two approaches and the average response time was measured. Figures 10 and 11 illustrate the average response time and average cost, respectively, for requests in the two scenarios described above.

Figure 10: Average response time for different requests in two scenarios
According to Figures 10 and 11, with respect to the average response time, from 12 requests, we had a speedup greater than 5x with “our distributed approach”. Considering the cost increased by about US$ 6.12 per hour (i.e., US$ 0.68 per hour × 9 instances = 6.12 per hour), we had a performance gain of approximately 81% in time per dollar. For example, for 24 requests, the distributed solution reduced the time from 397.13 seconds in the sequential approach to 60.92 seconds, giving a speedup of 6.39x. In this case, the performance gain is greater than 100% per dollar per hour.

6.2 Preliminary tests with deaf users

We performed some preliminary tests with Brazilian deaf users. The purpose of these tests was to provide qualitative measures about some aspects of the solution, such as, the translation, ease of understanding the LIBRAS video tracks generated by the solution and its naturalness, among others.

These tests were performed with five Brazilian deaf users (we tried to find a larger set of users, but, unfortunately, we did not find more volunteers). The group of users consisted of three men and two women between 24 and 36 years old with an average value of 29.2 years. They also have varied degrees of education and assess their level of knowledge in LIBRAS and BP on 1-to-5 scale (see Table 1).

Users were invited to watch an accessible content generated by the service and to complete a questionnaire about some aspects of the solution. The accessible content was generated from a part of a movie produced by UFPB TV (the TV of Federal University of Paraiba – UFPB on Brazil), developed with academic purposes. This video consists of a dialogue between two characters and lasts 65 seconds. To perform the test, we developed a SL Dictionary with 119 signs in LIBRAS.

The applied questionnaire had six questions. The first five questions rated these contents on a 1-to-5 scale for LIBRAS grammatical correctness, understandability, naturalness, quality of

Figure 11: Average cost for different requests in two scenarios

According to Figures 10 and 11, with respect to the average response time, from 12 requests, we had a speedup greater than 5x with “our distributed approach”. Considering the cost increased by about US$ 6.12 per hour (i.e., US$ 0.68 per hour × 9 instances = 6.12 per hour), we had a performance gain of approximately 81% in time per dollar. For example, for 24 requests, the distributed solution reduced the time from 397.13 seconds in the sequential approach to 60.92 seconds, giving a speedup of 6.39x. In this case, the performance gain is greater than 100% per dollar per hour.

6.2 Preliminary tests with deaf users

We performed some preliminary tests with Brazilian deaf users. The purpose of these tests was to provide qualitative measures about some aspects of the solution, such as, the translation, ease of understanding the LIBRAS video tracks generated by the solution and its naturalness, among others.

These tests were performed with five Brazilian deaf users (we tried to find a larger set of users, but, unfortunately, we did not find more volunteers). The group of users consisted of three men and two women between 24 and 36 years old with an average value of 29.2 years. They also have varied degrees of education and assess their level of knowledge in LIBRAS and BP on 1-to-5 scale (see Table 1).

Users were invited to watch an accessible content generated by the service and to complete a questionnaire about some aspects of the solution. The accessible content was generated from a part of a movie produced by UFPB TV (the TV of Federal University of Paraiba – UFPB on Brazil), developed with academic purposes. This video consists of a dialogue between two characters and lasts 65 seconds. To perform the test, we developed a SL Dictionary with 119 signs in LIBRAS.

The applied questionnaire had six questions. The first five questions rated these contents on a 1-to-5 scale for LIBRAS grammatical correctness, understandability, naturalness, quality of
Accessibility as a Service: Augmenting Multimedia Content with Sign Language Video Tracks

presentation, among others. We also used an additional question to check whether users really understood the content being transmitted. This question asks the user to select which of three alternatives (choice A, B or C) match with the content being transmitted. Among the three alternatives presented only one is correct. For example, in the video, a couple discusses financial problems. So, the question asks the users why the couple discusses: (A) jealousy, (B) financial problems, or (C) problems with children. Table 2 and 3 show a sample of this questionnaire and the average results of the users evaluation, respectively.

According to Table 3, the user assessment was moderated giving a 3.35 global score. The understandability had the highest score (3.6). This result is probably compatible with the match-success test, since 80% of users chose the correct alternative. The grammar and the sign naturalness, on the other hand, had the lowest scores (2.1 and 2.8, respectively). This probably indicates that more effort should be invested to improve translation quality and naturalness of 3D-avatar. As in San-Segundo et al (2011), some probable causes observed during the tests were the following:

- There were discrepancies between users about the structure of same sentences in LIBRAS. As other SLs (e.g., Spanish Sign Language (San-Segundo et al, 2011)), LIBRAS has an important level of flexibility in the structure of sentences.
- Avatar naturalness is not comparable to a human signing. As mentioned earlier, other studies already showed the low acceptance of avatars-based solutions by the deaf. It is necessary to invest more effort on increasing expressiveness and naturalness of the avatar.

<table>
<thead>
<tr>
<th>Users</th>
<th>Degree of education</th>
<th>Knowledge in LIBRAS</th>
<th>Knowledge in BP</th>
</tr>
</thead>
<tbody>
<tr>
<td>User1</td>
<td>Complete elementary school</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>User2</td>
<td>Incomplete high school</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>User3</td>
<td>Complete high school</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>User4</td>
<td>Undergraduate</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>User5</td>
<td>Master degree</td>
<td>5</td>
<td>4</td>
</tr>
</tbody>
</table>

Table 1: User’s profile in terms of degree of education and knowledge in LIBRAS and BP

<table>
<thead>
<tr>
<th>Questions</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) Easy to understand? (5 – Clear, 1 – Confusing)</td>
<td>5 4 3 2 1</td>
</tr>
<tr>
<td>2) Good LIBRAS grammar? (5 – Perfect, 1 – Bad)</td>
<td>5 4 3 2 1</td>
</tr>
<tr>
<td>3) The signing is natural? (5 – Move likes a person, 1 – Like robot)</td>
<td>5 4 3 2 1</td>
</tr>
<tr>
<td>4) Hand and arm movements are corrected? (5 – perfect, 1 – Bad)</td>
<td>5 4 3 2 1</td>
</tr>
<tr>
<td>5) Facial expressions are corrected? (5 – perfect, 1 – Bad)</td>
<td>5 4 3 2 1</td>
</tr>
</tbody>
</table>
| 6) Which alternative on the right matches with the content? (Why the couple discuss?) | A – jealousy  
  B – financial problems  
  C – problems with children |

Table 2: Sample of questionnaire
Accessibility as a Service: Augmenting Multimedia Content with Sign Language Video Tracks

7. Final Remarks
In this paper, we presented the architecture of a service for improving the access of deaf people to digital multimedia content. This proposal addresses accessibility features by automatically embedding sign language videos into multimedia contents submitted by the user and has an infrastructure to edit or add rules and signs, allowing collaborators to improve the quality of translation and presentation.

We also present an implementation of the proposed service adapted to run in a cloud computing environment and developed a case study for the Brazilian SL (LIBRAS).

As future works, we planned to incorporate a speech recognition component in the service to allow the generation of the SL video from speech (apart from subtitles). In addition, we think that more effort must be concentrated to make virtual signing more natural for the deaf. Thus, another future work involves the incorporation of some capture equipments in the WikiSL tool, such as Microsoft Kinect (www.xbox.com), to enable the generation of new sign from motion capture.

References

<table>
<thead>
<tr>
<th>Aspects</th>
<th>Mean scores</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understandability</td>
<td>3.6</td>
<td>0.89</td>
</tr>
<tr>
<td>Grammatically</td>
<td>2.1</td>
<td>1.23</td>
</tr>
<tr>
<td>Naturalness</td>
<td>2.8</td>
<td>1.48</td>
</tr>
<tr>
<td>Quality of movements</td>
<td>3.4</td>
<td>1.51</td>
</tr>
<tr>
<td>Quality of facial expressions</td>
<td>3.6</td>
<td>1.67</td>
</tr>
<tr>
<td>Global</td>
<td>3.35</td>
<td>–</td>
</tr>
<tr>
<td>Match-success</td>
<td>80%</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 3: Average scores for the questions

• There were also discrepancies between users about the correct signing of some signs. For example, users disagreed about the correct signing of the CAFÉ (coffee) and MERCADO (market) signs.

As examples, the following table shows the average scores for the questions:

<table>
<thead>
<tr>
<th>Aspects</th>
<th>Mean scores</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understandability</td>
<td>3.6</td>
<td>0.89</td>
</tr>
<tr>
<td>Grammatically</td>
<td>2.1</td>
<td>1.23</td>
</tr>
<tr>
<td>Naturalness</td>
<td>2.8</td>
<td>1.48</td>
</tr>
<tr>
<td>Quality of movements</td>
<td>3.4</td>
<td>1.51</td>
</tr>
<tr>
<td>Quality of facial expressions</td>
<td>3.6</td>
<td>1.67</td>
</tr>
<tr>
<td>Global</td>
<td>3.35</td>
<td>–</td>
</tr>
<tr>
<td>Match-success</td>
<td>80%</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 3: Average scores for the questions
Accessibility as a Service: Augmenting Multimedia Content with Sign Language Video Tracks


Biographical Notes

Tiago Maritan Ugulino de Araújo is a Professor in the Department of Science and a researcher of Digital Video Applications Lab (LAViD) at the Federal University of Paraíba (UFPB) – Brazil. He received the B.S. and Master degrees in computer science at UFPB and Ph.D. degree in computer engineering at Federal University of Rio Grande do Norte (UFRN) – Brazil. He worked as one of the designers of Ginga Middleware, adopted as standard by the Brazilian Digital TV System, and his research interests include accessible technologies for deaf people, multimedia systems and digital television.

Felipe Lacet Silva Ferreira received the bachelor’s degree in computer science from the Federal University of Paraíba (2010), Brazil, and is currently an M.Sc. candidate in informatics at the same university. He took part in the implementation of the open version of the Brazilian Digital Television Middleware, Ginga, and currently works as a researcher and project manager at the Digital Video Applications Lab, where he is a member of a group for research and improvement of accessibility features for deaf people. His interests include distributed and multimedia systems, focusing on digital television, digital cinema, accessibility and middleware.

Danilo Assis Nobre dos Santos Silva is an M.Sc. candidate and researcher of Digital Video Applications Lab (LAViD) at the Federal University of Paraíba (UFPB). He graduated in computer science in 1999 and since 1998 has been working on some projects in LAViD (TS Analyzer, WikiLIBRAS, GingaCDN, Multiprotocol Encapsulation, among others). His research interests include accessibility, multimedia systems and digital television.

Eduardo de Lucena Falcão has a B.Sc. and is currently an M.Sc. candidate in computer science at the Federal University of Paraíba (UFPB) – Brazil. He is a researcher of Digital Video Applications Lab (LAViD), integrating the project Accessibility as a Service, which aims at providing a scalable service (based on cloud computing) for automatic generation of accessible videos. He is an enthusiast in the field of Virtual Reality on the following topics: 3D for Web, multiprojection systems, stereoscopic vision.

Leonardo Dantas de Oliveira is studying internet systems at Centro Universitário de João Pessoa (Unipê) – Brazil and is a researcher of Digital Video Applications Lab (LAViD). He has experience in modeling, UV mapping, Rigging/Skinning and 3D animation. At LAViD, he works as a developer in projects related with accessibility for the deaf in cinema and TV. His research interests include accessibility, animation and motion capture (MoCap).
Accessibility as a Service: Augmenting Multimedia Content with Sign Language Video Tracks

Leonardo de Araújo Domingues is completing his bachelor degree in information systems at the Federal University of Paraíba – Brazil. He is a researcher of Digital Video Applications Lab (LAViD) and, currently, he integrates the project Accessibility as a Service, which aims at providing a scalable service (based on cloud computing) for automatic generation of accessible videos. His research interests include accessibility, multimedia systems and digital television.

Yúrika Sato Nóbrega is completing her bachelor degree in pedagogy at the Federal University of Paraíba (UFPB) – Brazil. She is a Brazilian Sign Language (LIBRAS) interpreter, a member of the Deafness Research Group at UFPB and member of Digital Video Applications Lab (LAViD). Currently, she is working as member of the project Accessibility as a Service of LAViD.

Hozana Raquel Gomes de Lima is studying a bachelor degree in computer science at the Federal University of Paraíba – Brazil. She is deaf, a member of the Deafness Research Group at UFPB and a researcher of the Digital Video Applications Lab (LAViD). Currently, she is working as a designer in the project Accessibility as a Service of LAViD.

Alexandre Duarte received a PhD degree in computing science in 2010 from the Federal University of Campina Grande, Brazil, with a sandwich period at CERN, Switzerland, for his work on fault diagnosis on computational grids. In the same year he joined the Department of Informatics of the Federal University of Paraíba, where he is currently an Associated Professor. His main research areas are in distributed systems and protocols, with special interest in grid computing, cloud computing and many-task computing.

Guido Lemos de Souza Filho is a Professor in the Department of Computer Science and leader of the Digital Video Advanced Applications Lab (LAViD) at the Federal University of the Paraíba – Brazil. He is a member of the Digital Television Brazilian System Forum Board, which is in charge of the definition of the Brazilian Digital TV standards. He worked as one of the designers of Ginga Middleware, adopted as standard by the Brazilian Digital TV System. At LAViD, he is responsible for the execution of the following: research in distributed multimedia systems, digital television, digital cinema and accessibility.