Data-driven development of Virtual Sign Language Communication Agents

Agathe Balayn[1], Heike Brock[2] and Kazuhiro Nakadai[2]

Abstract— **Engaging deaf and hearing people in common discussions requires interfaces to help them understand each other, such as robot agents that translate spoken language into Sign Language (SL) expressions and vice-versa. However, the recognition and generation of signed sentences is a complex task of high dimensionality that cannot be solved in sufficient quality yet. Thus, it is necessary to develop new technologies of improved performances. The sequence to sequence neural network model, traditionally used for machine translation, is adapted to the above two tasks by treating a SL sequence as a multi-dimensional sentence. We defined an encoding of the SL annotations and conducted experiments on the network structure to define a most accurate translation model. This study proves the network trainable and possibly applicable in real-life with an extended dataset, which shall be tested for deployment in virtual translation assistants in the following.**

Index Terms— **Deep Learning; sequence to sequence; Sign Language recognition; Sign Language generation**

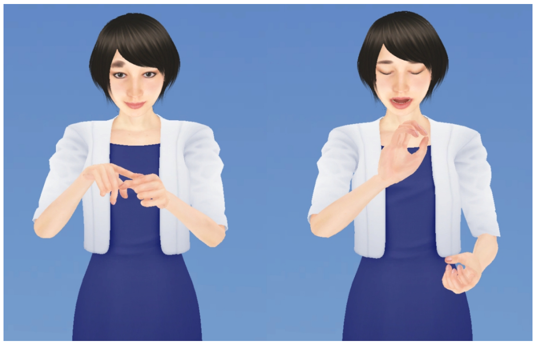

Fig. 1: Our avatar for JSL animation generated from Japanese text.

I. INTRODUCTION

Hearing loss affects over 5% of the world population and more than 1/3 of all people aged 65 years or above [1]. Deaf people use Sign Language (SL) to communicate with each other, Japanese SL (JSL) is the native language of 60.000 people and is spoken by approximately 317.000 people [2]. However, there remains a lack of communication tools to support interactions between hearing people and SL speakers. The former usually are not proficient in SL and have problems reading a conversation signed in usual speed, even when having learned SL. SL native speakers on the other hand have difficulties understanding written texts and would benefit considerably from an information display in their native or preferred language [3]. Thus, it is useful to build a bidirectional system able both to translate signed sentences to text for hearing people, and to generate signed sentences from written or spoken text via a 3D avatar (Fig.1) for deaf people.

However, SL is not a regular language: it does not make use of the voice but of movements and multi-modal content indications (fingers, hands, arms, body gestures, facial expressions). Thus, translation between SL and another language can not be tackled similarly to other Machine Translation (MT), and common translation tasks like video captioning [4] need to be adapted to improve accuracy and reliability. Systems developed to recognize SL words or to generate SL animations generally do not account for all aspects of a signed sentence, such as facial expression, natural signing speed, transitions between words and temporal and spatial context information [5], what makes them incomplete and hard to interpret. To represent the multi-dimensional aspects of SL and solve the issues related to its continuous movements, we investigate the use of deep Machine Learning (ML) models, useful for many domains. In concrete, we discuss the application feasibility of a deep sequence to sequence (Seq2Seq) learning model on a corpus of JSL sentence expressions, that could easily be adapted to any SL. To the best of our knowledge, it is the first time bidirectional SL translation is tackled with such a network, and hence it is important to identify parameters and strategies enabling model adaptation and improvement. This supports the engineering of highly accurate and reliable communication systems in the future, that both recognize and generate new JSL sentences to visualize on a 3D avatar.

The paper is structured as follows. First, we review the previous techniques employed for SL translation. Then, we detail our available SL corpus (Sec. III), and the set-up of our system (Sec. IV). We present the results of the experiments we conduct on the system (Sec. V), and discuss them in Sec. VI. Finally, we draw conclusions on the feasibility of Sign Language translation with recurrent neural networks and present possible future improvements (Sec. VII).

II. RELATED WORKS AND OBJECTIVES

A. Sign Language Communication Systems

Although systems that facilitate communication between spoken and signed languages would improve engagement and

[1] Agathe Balayn is a student in Delft University of Technology, The Netherlands, and worked on the paper during an internship at the Honda Research Institute, Japan A.M.A.Balayn@student.tudelft.nl

[2] Heike Brock and Kazuhiro Nakadai are with the Honda Research Institute Japan Co., Ltd., 8-1 Honcho, Wako-shi, Saitama 351-0188, Japan h.brock@jp.honda-ri.com and nakadai@jp.honda-ri.com

978-1-5386-7980-7/18/$31.00 ©2018 IEEE

integration of deaf people, technologies for the recognition and generation of SL expressions lack behind the quality of spoken language interfaces and of coarse full body activity data activity recognition. One reason for this is that SL, as a minority language, is subject to less research on related aspects like linguistic knowledge. Besides, only few ML corpora are publicly available for research given the visual aspects of a signed utterance that impose the need for a specialized data collection as well as expert knowledge for annotation. As a result, it is still common for both SL recognition (SLR) and SL synthesis (SLS) to use shallow MT methods.

*1) **SLR**:* The most recent review of approaches for the recognition of SL was published in 2005 [6]. In general, introduced methods were based on well-established techniques such as Hidden Markov Models (HMM), Principal Component Analysis (PCA), random forest, nearest neighbours or rule-based methods. Furthermore, most works focused only on hand gestures captured with image, video, Kinect sensors or PowerGloves and were trained to recognize single words only [7], [8], [9]. This is an unnatural assumption considering that SLs do not only consist of hand gestures but also face and body movements and are commonly expressed in full sentences. Therefore, to date, most methods could not be applied in real translation interfaces.

Since then, only few deep neural networks were used that aim to include all important aspects of a signed expression. Full body studies reach high accuracies on small corpora: 91.7% accuracy for 20 words by segmentation and automatic feature extraction with Convolutional Neural Networks (CNN) [10], and 86% accuracy for 73 words by automatic extraction of the most discriminative frames [11]. Most recent efforts to translate consecutive signs recognize short sentence expressions in Chinese SL with conditional random fields with 90% accuracy based on manually designed features [12]. Transition modelling reaches 87.4% accuracy [13], while methods inspired from speech recognition achieve 33.4% error rate for a single signer dataset [14], and CNN achieve 62.8% accuracy over 60 classes [15].

*2) **SLS**:* Signed expressions are synthesized by translating textual sentences into gloss annotation sentences following the SL grammar, and rendering them into an avatar. Sequence generation is generally tackled in two main different ways.

One approach is to render the signs with kinematics calculations based on their visual annotations such as in Moemedi [16]. However, this method is of high computational complexity and resulting animations do not look natural since each sign is exactly signed in the same way, with the same starting and ending positions. Zhao et al. [17] add specific information (location in space of the sign, strength of the signed gesture, speed and flow of the sign) to the annotated SL words to distinguish between similar words. Moreover, they annotate pauses in the sentence, and negations of words, question intonations, passive or active voice of the words over certain of the annotated words since the words themselves have similar hand-gestures with only small variations to express these variations in their meanings. Using inverse kinematics, they can generate gestures depending on these annotations, and they deal with transitions between words using Parallel Transition Networks (PaT-Net).

The second approach is to map the annotations to movements stored in a database collection, consequently each sign is signed similarly with identical starting and ending positions. For example, Suszczańska et al.[18] pass descriptions of 600 SL words to an OpenGL application to generate a signed sentence. Tokuda et al. [19] use a rule-based method, searching the closest SL word in a dictionary from the input Japanese word, and display this signed word. Here, it is common to interpolate transitions between signs for more naturalness. For example, Lu et al. [8] manually add control codes such as pauses in between the annotated sentence, and interpolate linearly the transitions between words. Ohki et al. [20], [21] map words to collected PowerGloves data input in the Computer Graphics program, and interpolate the transitions between words for more naturalness.

However, as long as facial expressions and non manual signs are not conveyed, such synthesized animations achieve poor ratings among deaf individuals [22]. Research to add facial variations on top of the manual signs is done by rendering the avatar with manual movements and adding eyebrow and head movements [23]. Similarly, Xu et al. [24] constitute a dictionary of SL words but they add to the hand gestures facial expressions to express the mood of the sentence. Kacorri [25] generates facial animations with data-driven models and shows that continuous profile models give better results than previous methods, comparing them using multivariate Dynamic Time Warping (DTW).

B. Intended improvements

Direct interaction of hearing and deaf individuals shows a need for SL communication agents. In particular in situations where professional translation services are not available, such as internal company meetings, fast and accurate translation of both spoken and signed statements is an important factor for better accessibility of information and hence enhanced inclusion of all associates. We adapt an approach able to learn long-term dependencies of spoken languages, deep Recurrent Neural Networks (RNNs) with Long-Short Term Memory (LSTM) cells. Since MT methods are used to generate and at the same time understand sequences, this allows us to perform our double task with one identical model, and to utilize full sentences to train a network that can both recognize and generate SL utterances. These utterances incorporate full body motion information as well as facial expressions representing lexical and grammatical content of JSL, while the gloss annotations are enhanced with additional descriptive data.

For SLS, models trained on full sentences are expected to improve the overall quality of a signing avatar animation by intrinsically learning transitions between words, speed and spatial and temporal dependencies in a natural way. For SLR, the combination of multi-channel information (body, face) could improve recognition accuracy within continuous sentence utterances. Moreover, RNNs do not need temporal

978-1-5386-7980-7/18/$31.00 ©2018 IEEE

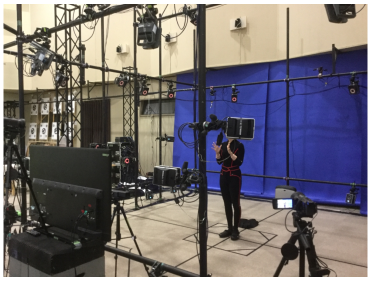

Fig. 2: Data recording set-up: 42 3D motion capture cameras and a Microsoft Kinect v2.

segmentation and would hence save a lot of corpus annotation work in the future, considering that it requires a large amount of data to train on.

III. Machine Learning Corpus

A. *Description of the corpus*

379 sentence structures were signed in 2 to 3 different speeds (total of 812 sentences with a vocabulary of 195 words) by one fluent signer (Child of Deaf Adults) and simultaneously recorded utilizing a markerless and a marker-based motion capture systems [26]. To train the SLR, video and depth data of the JSL sentences were acquired using a Microsoft Kinect because this set-up is cheap, portable and thus usable in the real world. For SLS, highly detailed 3D motion capture data of full body, face and finger were acquired by a dense Vicon system of 42 cameras (Fig.2): the point cloud data are high-dimensional and suitable inputs to animate a 3D avatar.

Within this corpus, groups of 4 to 6 sentences with similar vocabulary and grammar structures were composed to ensure the repetitive occurrence of the word content. We use 2 sentences of each group for testing (244) and the rest for training (568).

B. *Data augmentation*

JSL data augmentation is performed since the corpus is small compared to MT tasks corpus. We investigate different methods to multiply the data amount. In concrete, data is 1) multiplied by 4 by adding noise between 0.25 and 1 standard deviation of the original data; 2) multiplied by 2 by downsampling (skipping every second sample) or upsampling (utilizing linear interpolation); 3) multiplied by 16 by combining all the techniques together.

IV. System Overview - Experiments

A. *General system pipeline*

We address the problem of translating JSL gloss annotations to JSL motion sequences and vice-versa (Fig.3) as it is generally done for SL generation. Indeed, translating directly to Japanese or English language would lead to lower accuracies because we have few data and the networks would not be able to learn such a complex task (recognition or generation of single words, and their combination into a meaningful grammatically-correct spoken language sentence). We show in Fig. 4 an example of signed sentence and its associated representations along the translation process.

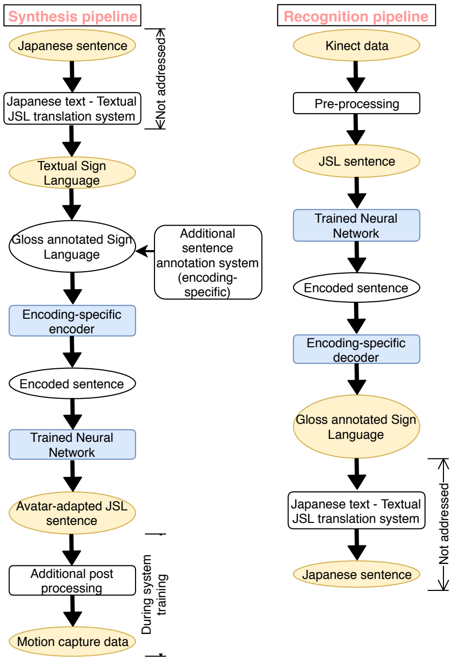
Synthesis pipeline
Japanese sentence
Japanese text - Textual JSL translation system
Not addressed
Textual Sign Language
Gloss annotated Sign Language
Additional sentence annotation system (encoding-specific)
Encoding-specific encoder
Encoded sentence
Trained Neural Network
Avatar-adapted JSL sentence
Additional post processing
During system training
Motion capture data
Recognition pipeline
Kinect data
Pre-processing
JSL sentence
Trained Neural Network
Encoded sentence
Encoding-specific decoder
Gloss annotated Sign Language
Japanese text - Textual JSL translation system
Not addressed
Japanese sentence

Fig. 3: Pipeline of the system. Left: Sign Language Synthesis (SLS), right: Sign Language Recognition (SLR).

Annotated sentences and motion data are encoded into sequences of fixed size vectors. Then, two separate Seq2Seq models are trained for the two tasks, taking as input and output the encoded annotations and normalized JSL motion sentences. Network outputs are post-processed by removing repeated output words.

B. *Gloss annotations encoding*

*1) **Previous efforts to annotate Sign Language**:* Several annotation systems of Sign Language have been developed in the past, with the aim of describing the SL words for linguistic studies such as the HamNoSys [27] (Hamburg Notation System for Sign Language).

We base our annotation on the definition of SignWriting and its corresponding SignWriting Markup Language [28]. This representation considers that the words are composed of several entities (hands, head, movement, body, dynamic) and that each entity has a unique encoding (concatenation of symbol number, variation, fill, rotation, category, group). It enables to describe the signs very precisely, for example, it contains information on which hand is signing the word, on the orientation of the hand, and on where the hand is placed in the signing space.

Moreover, Sign Language differs from spoken language in the way degree variations are expressed, such as how appreciation at different levels is shown. Zhao et al. [17] call this aspect of SL inflectional morphology and encode it using different numbers. We set up to transcribe these

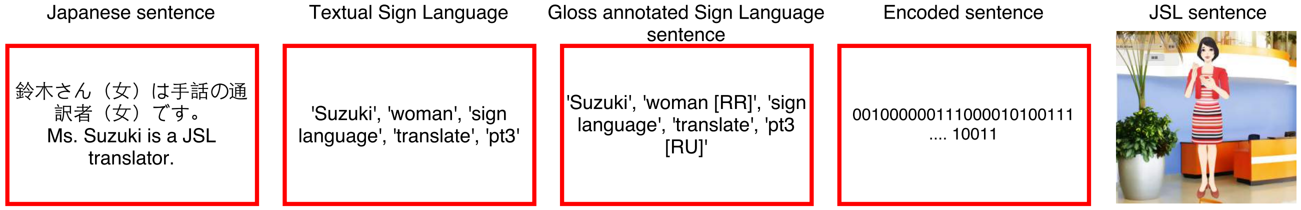
Japanese sentence
Textual Sign Language
Gloss annotated Sign Language sentence
Encoded sentence
JSL sentence
鈴木さん（女）は手話の通訳者（女）です。
Ms. Suzuki is a JSL translator.
'Suzuki', 'woman', 'sign language', 'translate', 'pt3'
'Suzuki', 'woman [RR]', 'sign language', 'translate', 'pt3 [RU]'
00100000001110000101001111 10011

Fig. 4: Example of sentence types. From left to right, 1) the Japanese sentence and its English translation, 2) the gloss annotation sentence (the words follow the JSL order), 3) the gloss annotation sentence with additional information, 4) the corresponding encoded sentence, and 5) the JSL sentence.

characteristics in our encoding. Besides, to describe the signing location into space, we base our model on the signing space described by Zardoz (section 7.1 [29]): it divides the signer space into multiple areas and reports the hand positions into these areas.

*2) **The three encodings**:* We define three different encodings. The simplest encoding 1) is a one-hot encoding of all defined corpus words.

For the second encoding 2) (Fig. 5), words carrying related meanings and signed similarly are considered identical, but distinguishable with additional indications (adjective intensity, question, genitive, passive voice, signing hand, beginning and ending directions). Hence, this encoding is identical to the first one with a smaller number of words, to which are concatenated the one-hot encoded indications. These additional indications can only be detected when using facial and body movements information in addition to the hand gestures. Thus, this encoding would increase the interpretability of the generated models and the precision of the recognized sentences. An example of such words is ”to receive” and ”to give”, which are signed with the same gesture but in inverse direction (”rewinded”).

The third encoding 3) (Fig. 5) concatenates the first one and one-hot encoded SignWriting [28] descriptions of the gestures. As SignWriting words have variable lengths, one word is encoded into several vectors.

C. Translation model

*1) **Model training process**:* After learning an encoding for the corpora of the two languages, a network is trained. We utilize a variation of RNN, the Seq2Seq model of Sutskever et al. [30] for English-French translation, similar to the encoder-decoder model of Cho et al. [31]. This network consists of a first RNN or LSTM encoder network which reads a variable-size input sentence and maps it to a fixed-size vector (network internal state); and a second identical decoder network conditioned on the first one, trained to predict the translation of the sentence. For the SLR, an additional layer performs a softmax function separately on the different parts of the encodings (seen as classes), and chooses the word with the higher probability each time. The loss function employed is the cross entropy (SLR scenario), respectively mean-squared error (MSE) or Soft DTW loss [32] (SLS scenario).

To train and test the model, the input sequence is fed to the first network and the second network then returns an output. When training the network, this output is compared to the expected one to compute the loss, which is back-propagated to the two networks. To get an output from the system, the input is passed -in reverse order as it was shown to give higher accuracy [30]- through the first network whose internal state is copied to the second network. Afterwards, the second network is fed with a beginning of sentence token, and it gives out one output. Depending on the mode chosen, this output is fed back to the second network (”feed-previous” mode (FP)) which outputs a second output, or the expected output is fed to the second network (non ”feed-previous” mode (nFP)) (Fig. 6). This process is repeated until an end of sentence token appears in the output. To decode the outputs, the most likely translation is found using a beam search decoder, with beam size of 1.

*2) **Model details**:* In order to indicate the sentence separation, special tokens are added to the word corpus: the beginning of sentence (BoS) and end of sentence (EoS) tokens. Additionally, since the model needs fixed-size entries, a padding token is used to complete sequences which are too short. Lastly, we introduce an ”unknown” token used to replace the least frequent words within the corpus. The sentence size varies from 5 words to 25 words, and the signed sequence from 100 steps to 400 steps. Therefore, we also make use of several buckets in order to manage varying sentence size and group the sequences into smaller groups of similar length sentences.

D. Experiments: network training

The network is implemented on TensorFlow [33] and run on one GPU. It is optimized using the gradient descent optimizer.

According to Sutskever et al [30], training in the nFP mode is faster, but networks trained with the FP mode are more robust to errors in the outputs. Moreover, the best performances are achieved with deep LSTMs (4 layers) rather than shallow ones. Thus, we started training the network with a large architecture size, but further experimented with different variations of the network. Some of the tested parameters are listed in Table I.

V. Results

Generally, it is noted that reducing the corpus size does not improve the accuracy, the network never recognizes the

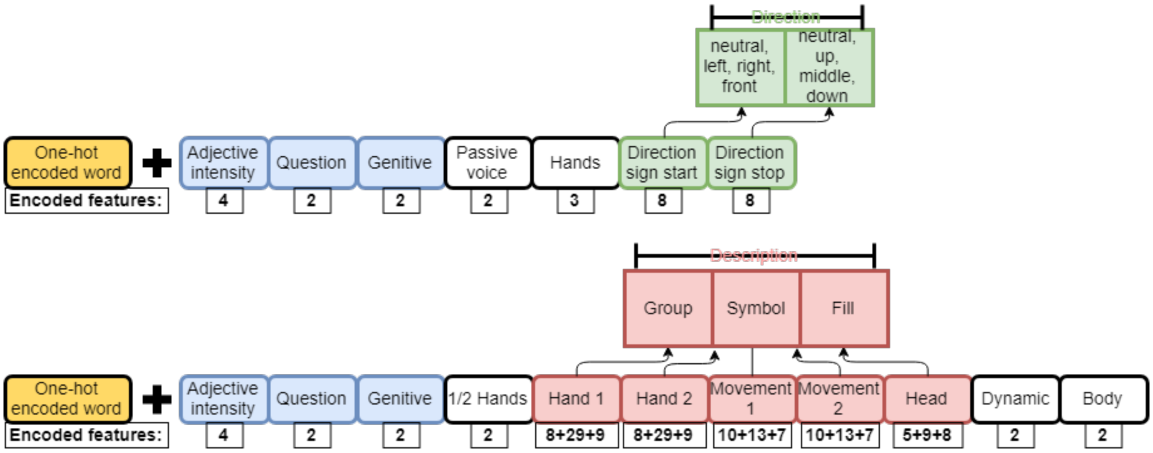

Fig. 5: Description of encodings (2) and (3). The numbers are the numbers of features used to encode each corresponding gesture indications. Top: Description of the second encoding. Size: corpus size + 3 tokens + 29 specifications. Bottom: Description of the third encoding. Size: corpus size + 3 tokens + 180 SW + 8 specifications

Type	Values
Neural network architecture	
Cell	LSTM, GRU, RNN
Nb. of layers	1 to 5
Nb. of cells/layer	64 to 500
Nb. of buckets	between 1 and 5
Training process tuning	
Gradient clipping	no or 5
Dropouts	0 to 0.4
L2-reg.	0.0001 to 0.1
Training process	FP or nFP
Inputs and outputs treatment	
Nb. total words	150 or 100 (frequency $>$ 4 or 10)
Encoding	1), 2), 3)
PCA	100% (45 dim.) or 90% ($\approx$ 37 dim.) of the variance

TABLE I: Experiment variables

”unknown” token which replaces many words. Whereas basic RNN cells cannot learn the data dependencies, LSTM and GRU cells are of better and similar performance.

A. Recognition task

The network suffers of overfitting which was decreased by the applied data augmentation. Regularization reduces both overfitting and accuracy. This suggests that the network performance could increase with more data: overfitting would decrease, larger networks could be used, so the accuracy would raise. When using the best performing architecture (namely 1 layer of 256 LSTM cells), encoding 3) has both training and test accuracies slightly higher than the two word-based encodings. The former one explicitly describes sign specificities such as hand shapes and directional information, that could support the network in weight learning to distinguish similar words. With the second encoding, the variational indications are not regularly accurately recognized. Generally, shorter sentences are recognized with few errors and only adjectives are confused, suggesting that the network is able to learn the sentence structures (Fig.7 and Fig.8). Longer sentences require more training epochs for similar accuracies. When including resampled data, the decoded sentences have repetitions of words and post processing is necessary. In the decoded sentences, the rarest words are less often recognized. The EoS token never appear, whereas the word ”pt” (referential pointing gestures) is always found. This is due to the unbalanced dataset: words do not have the same frequency in the corpus and hence slow down and bias the learning process. In JSL, the EoS token is less frequent than the ”pt” word used as context reference in the middle and end of a sentence. Consequently, the network mixes both words.

B. Generation task

Since the system accuracy remained low when employing the whole set of features (642 dimensions), lower dimensionality outputs were tested: 1) PCA selected data streams with high variance (492 dimensions), 2) all data streams excluding lower body and facial expressions (219 dimensions), 3) features of one kinematic chain (right arm) (12 dimensions). The accuracy decreases in the first 1500 epochs and then remains constant without overfitting, indicating that using a larger network would improve the accuracy. The beginning of the generated sequences is correctly predicted (Fig.9). However, the output positions stay identical after 200 time steps (over $\approx$ 1000 total steps) and the correct trajectory is not entirely followed. The network learns but lacks a sufficient number of parameters to learn complete sentence expressions. The FP mode does not enable any training. Moreover, the soft DTW loss leads to better performances than the MSE loss.

VI. Discussion

A. SLR

For SLR, most papers are not interested in recognizing continuous unsegmented data, but concatenations of separated gestures. Since it is not the same task, it cannot be used as a baseline. Recent papers however present methods

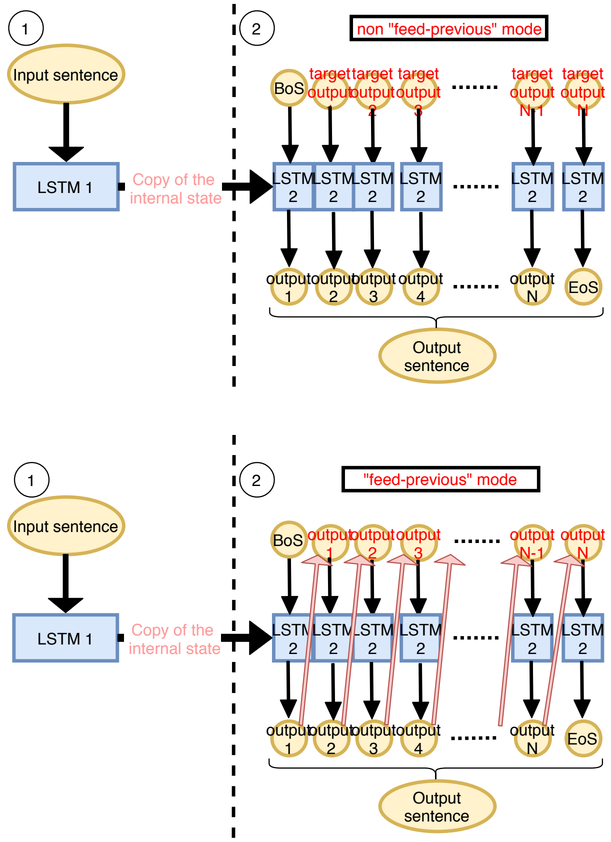

Fig. 6: ”Feed-previous” (FP) and non ”feed-previous” (nFP) training modes.

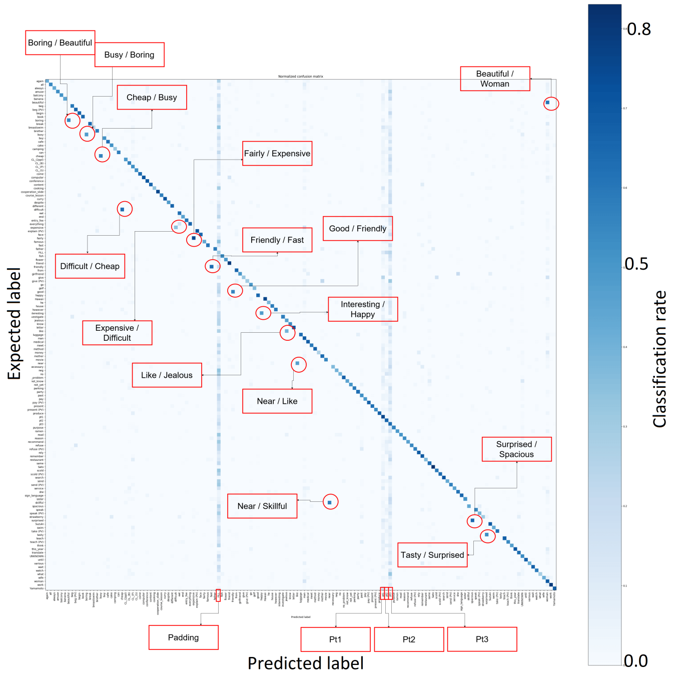

Fig. 7: SLR confusion matrix on training data. The axes represent the different words in the corpus and the colour their (mis)classification rate. Adjectives, ”pt” and padding are confused, the network does not learn to distinguish the different adjectives. The network might have learned relationships between words (such as where adjectives are employed) in a sentence.

for continuous recognition with the Word Error Rate (WER) evaluation metric -which is a more optimistic metric than our accuracy measurement. Koller et al. [34] obtain 26.8% WER and Camgoz et al [35] 40.7% WER on a hand gesture dataset. We cannot compare directly the performances since the datasets are different, ours taking into account the full-body gestures. Our SLR system shows encouraging results with a maximum accuracy of 53%, but to employ the translation system in real life, it is required to achieve recognition rates of more than 80%. Here, it should be noted that the applied accuracy metric is not fully tuned to our task, and WER would give higher performances. Currently our evaluation simply compares the words in the target and the output sentences at their specific location inside the sentence. Thus, if one word is repeated twice in the output, all the following words are shifted in the sentence and none of them are accounted as correct even if they are. Our results have slightly lower performances than recent works but they are more portable on robotics platforms since we use Kinect data which require less computing power (smaller dimensionality) and these platforms usually have Kinect sensors and not full-image cameras. Additionally, the recognized information are more complete since it uses full-body gestures instead of hands only, what enables to recognize more various words whose differences are expressed in face and body movements. Besides, the following additional changes could help to reach this target number. To avoid overfitting to improve recognition of complex sentence structures, our main recommendation is to collect a larger dataset, while making sure the word frequencies are balanced. This could mean to include shorter sequences such as short frames or simple word composita, to learn the rarest words. A better way to represent and differentiate ”unknown” words should also improve the accuracy. Furthermore, since PCA on the Kinect data reduces overfitting, similarly a word embedding as used in MT could reduce the dimensionality of the gloss annotations and hence the need for data. Moreover, using

Sentence1:pt1,mother,CL_2ppl,cafe,CL_P,tasty,banana,cake,eat,end
epoch400:Sato,mother,pt3,pt3,CL_P,CL_P,pt3(x11)
epoch800:pt1,mother,pt3,cafe,CL_P,surprised,cake,cake,eat,end,end,pt3(x3),pt2(x2)
epoch1200:pt1,mother,CL_2ppl,cafe,CL_P,surprised,banana,cake,eat,end,pt(x6)

Sentence2:pt1,mother,cafe,CL_P,tasty,cake,eat,end,speak(PV),pt3
epoch400:pt1,mother,pt3,pt3,CL_P,CL_P,pt1,pt1,pt3(x9)
epoch1200:pt1,mother,CL_2ppl,pt3,cafe,CL_P,strawberry,cake,eat,neg,pt3,neg,pt(x5)
epoch2000:pt1,mother,pt3,cafe,CL_P,surprised,cake,eat,eat,pt3,pt3,end,pt3,movie(x3)

Fig. 8: Recognition of two sentences (before post-processing) during the training process. In blue the correctly recognized words and in read the incorrect ones. More training epochs are needed to learn the longer sentence, suggesting that more training is also required for full sequence generation.

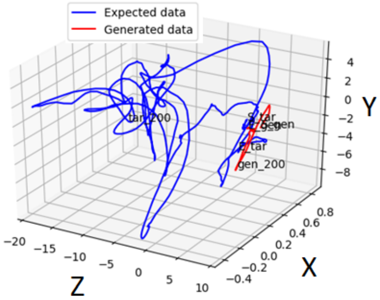

Fig. 9: Comparison of the generated and expected collar rotations over time along the Z, X, Y axes. Only the beginning of the sequence is similar. The annotations on the plot represent the targeted and generated rotations at different times (start and end times, and after 200 units of time.)

a CNN to embed the inputs to the LSTM could learn a meaningful representation of the data while reducing the size of the feature space. For now, the system is usable on simple sentences of common vocabulary. Recognition of complex and rare sentences is not accurate enough yet and a sentence language model could support the sentence prediction. Finally, to help the recognition of the additional indications reported in encoding 2), training the system on separate words with different indications should also be helpful.

B. SLS

As explained in the related work analysis, previous techniques for SL generation simply concatenate pre-collected movements to form sentences (with possibly superposition of additional pre-saved facial expressions), what does not render them in a natural way. No literature directly synthesizes the SL gestures from the sentences and thus we do not have a baseline to compare our system on. Our system, if higher performances were obtained, would enable to synthesise SL sentences with multiple variations represented in the encodings, such as sign-speed changes depending on the adjective intensity.

During SLS, only one to two low dimension words are outputted. Thus, learning separate models for different parts of the body (left, right arms, hands, facial expressions) and merging them together is a solution to explore. Generating facial expressions for example would give nuance indications on the emotional state of the signer. Since dimensionality reduction increased the performance, we assume that training a larger network with more GPU memory would extend the length of the outputs. Besides, the inclusion of an attention model could help to learn longer term dependencies. In MT, input and output sentences have relatively similar lengths compared to our sequences of annotations which are approximately ten times shorter than the target JSL sequences. We suppose that this length gap is a limiting factor and suggest helping the network to learn by duplicating the words in the input sequence. Lastly, the network might benefit of being pre-trained using single words before full sentence training: easily generating individual words, it would focus on understanding transitions and dependencies. This requires (automatic) sentence segmentation or collection of new training samples.

VII. Conclusion

We introduced and evaluated a Seq2Seq learning model for SL communication interfaces. Results indicate that the model performs well on simple common sentences, and that extensions would help it on longer translation tasks. These results shall be further tested to achieve a complete communication system including sign recognition and animation creation as a final goal. This new tool would enable more fluent conversations between hearing and deaf people, and easier access to written and oral resources for SL native speakers. Further training is necessary for real-life set-ups, but the employed architecture shows promising performances for two tasks that could not be handled simultaneously yet. Besides, embedding this system on a human-like robotic agent would enable to carry out the double task easily. On the one hand, the system would only have to direct its visual sensors to the current person speaking SL to process the gestural speech and output a textual or spoken translation. On the other hand, audio sensors could enable it to obtain a written text of the currently spoken sentences (or textual sentences could directly be sent to the system), and the neural network model would process them and output the commands for the robot to sign the translated sentences. Transferring from positions inputted to the 3D avatar to commands for a robot would only require a mapping between the avatar and the robot coordinate referential frames, considering that certain robotic systems are able to interpolate intermediate positions and the torques applied to each joint of the robot between two given positions in order to create full gestures. Real-time use appears feasible since the delay to obtain outputs from the neural network and post-process them is short (a few seconds) but still too long for instant translation.

References

[1] "World health organization: Deafness and hearing loss - fact sheet," {http://www.who.int/mediacentre/factsheets/fs300/en/}, note = Accessed: 2018-03-03.

[2] "ethonologue: Japanese sign language," {http://www.ethnologue.com/18/language/jsl/}, note = Accessed: 2018-03-03.

[3] S. Goldin-Meadow and R. I. Mayberry, "How do profoundly deaf children learn to read?" *Learning Disabilities Research & Practice*, vol. 16, no. 4, pp. 222–229, 2001.

[4] S. Kawas, G. Karalis, T. Wen, and R. E. Ladner, "Improving real-time captioning experiences for deaf and hard of hearing students," in *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*. ACM, 2016, pp. 15–23.

[5] M. Huenerfauth, "Generating american sign language classifier predicates for english-to-asl machine translation," Ph.D. dissertation, University of Pennsylvania, 2006.

[6] S. C. Ong and S. Ranganath, "Automatic sign language analysis: A survey and the future beyond lexical meaning," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 6, pp. 873–891, 2005.

[7] A. Karami, B. Zanj, and A. K. Sarkaleh, "Persian sign language (psl) recognition using wavelet transform and neural networks," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2661–2667, 2011.

[8] S. Lu, S. Igi, H. Matsuo, and Y. Nagashima, "Towards a dialogue system based on recognition and synthesis of japanese sign language," in *International Gesture Workshop*. Springer, 1997, pp. 259–271.

[9] H. Cooper, E.-J. Ong, N. Pugeault, and R. Bowden, "Sign language recognition using sub-units," *Journal of Machine Learning Research*, vol. 13, no. Jul, pp. 2205–2231, 2012.

[10] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in *Workshop at the European Conference on Computer Vision*. Springer, 2014, pp. 572–578.

[11] C. Sun, T. Zhang, and C. Xu, "Latent support vector machine modeling for sign language recognition with kinect," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 6, no. 2, p. 20, 2015.

[12] H.-D. Yang, "Sign language recognition with the kinect sensor based on conditional random fields," *Sensors*, vol. 15, no. 1, pp. 135–147, 2014.

[13] K. Li, Z. Zhou, and C.-H. Lee, "Sign transition modeling and a scalable solution to continuous sign language recognition for real-world applications," *ACM Transactions on Accessible Computing (TACCESS)*, vol. 8, no. 2, p. 7, 2016.

[14] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, 2015.

[15] O. Koller, H. Ney, and R. Bowden, "Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3793–3802.

[16] K. A. Moemedi, "Rendering an avatar from sign writing notation for sign language animation," Ph.D. dissertation, University of the Western Cape, 2010.

[17] L. Zhao, K. Kipper, W. Schuler, C. Vogler, N. Badler, and M. Palmer, "A machine translation system from english to american sign language," *Envisioning machine translation in the information future*, pp. 191–193, 2000.

[18] N. Suszczańska, P. Szmal, and J. Francik, "Translating polish texts into sign language in the tgt system," in *Proc. of the 20th IASTED Multiconference Applied Informatics, Innsbruck, Austria*, 2002.

[19] M. Tokuda and M. Okumura, "Towards automatic translation from japanese into japanese sign language," *Assistive Technology and Artificial Intelligence*, pp. 97–108, 1998.

[20] M. Ohki, H. Sagawa, T. Sakiyama, E. Oohira, H. Ikeda, and H. Fujisawa, "Pattern recognition and synthesis for sign language translation system," in *Proceedings of the first annual ACM conference on Assistive technologies*. ACM, 1994, pp. 1–8.

[21] H. Sagawa, M. Ohki, T. Sakiyama, E. Oohira, H. Ikeda, and H. Fujisawa, "Pattern recognition and synthesis for a sign language translation system," *Journal of Visual Languages & Computing*, vol. 7, no. 1, pp. 109–127, 1996.

[22] M. Kipp, Q. Nguyen, A. Heloir, and S. Matthes, "Assessing the deaf user perspective on sign language avatars," in *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility*. ACM, 2011, pp. 107–114.

[23] S. Ebling and M. Huenerfauth, "Bridging the gap between sign language machine translation and sign language animation using sequence classification," in *Proceedings of the 6th Workshop on Speech and Language Processing for Assistive Technologies (SLPAT)*, 2015.

[24] L. Xu and W. Gao, "Study on translating chinese into chinese sign language," *Journal of computer science and technology*, vol. 15, no. 5, pp. 485–490, 2000.

[25] H. Kacorri, "Data-driven synthesis and evaluation of syntactic facial expressions in american sign language animation," Ph.D. dissertation, City University of New York, 2016.

[26] H. Brock and K. Nakadai, "Deep jslc: A multimodal corpus collection for data-driven generation of japanese sign language expressions," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, may 2018, p. To appear.

[27] T. Hanke, "Hamnosys-representing sign language data in language resources and language processing contexts," in *LREC*, vol. 4, 2004.

[28] A. C. da Rocha Costa and G. P. Dimuro, "Signwriting-based sign language processing," in *Gesture Workshop*. Springer, 2001, pp. 202–205.

[29] T. Veale, A. Conway, and B. Collins, "The challenges of cross-modal translation: English-to-sign-language translation in the zardoz system," *Machine Translation*, vol. 13, no. 1, pp. 81–106, 1998.

[30] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[31] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*.

[32] M. Cuturi and M. Blondel, "Soft-dtw: a differentiable loss function for time-series," *arXiv preprint arXiv:1703.01541*, 2017.

[33] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[34] O. Koller, S. Zargaran, and H. Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[35] N. C. Camgoz, S. Hadfield, O. Koller, and R. Bowden, "Subunets: End-to-end hand shape and continuous sign language recognition," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.