International Journal of Computer Vision (2021) 129:2113–2135
https://doi.org/10.1007/s11263-021-01457-9

Continuous 3D Multi-Channel Sign Language Production via Progressive Transformers and Mixture Density Networks

Ben Saunders[1] · Necati Cihan Camgoz[1] · Richard Bowden[1]

Received: 16 October 2020 / Accepted: 7 March 2021 / Published online: 7 May 2021

Abstract

Sign languages are multi-channel visual languages, where signers use a continuous 3D space to communicate. Sign language production (SLP), the automatic translation from spoken to sign languages, must embody both the continuous articulation and full morphology of sign to be truly understandable by the Deaf community. Previous deep learning-based SLP works have produced only a concatenation of isolated signs focusing primarily on the manual features, leading to a robotic and non-expressive production. In this work, we propose a novel Progressive Transformer architecture, the first SLP model to translate from spoken language sentences to continuous 3D multi-channel sign pose sequences in an end-to-end manner. Our transformer network architecture introduces a counter decoding that enables variable length continuous sequence generation by tracking the production progress over time and predicting the end of sequence. We present extensive data augmentation techniques to reduce prediction drift, alongside an adversarial training regime and a mixture density network (MDN) formulation to produce realistic and expressive sign pose sequences. We propose a back translation evaluation mechanism for SLP, presenting benchmark quantitative results on the challenging PHOENIX14T dataset and setting baselines for future research. We further provide a user evaluation of our SLP model, to understand the Deaf reception of our sign pose productions.

Keywords Sign language production · 3D Multi-channel sign language · Continuous sequence generation

1 Introduction

Sign languages are visual multi-channel languages and the main medium of communication for the Deaf. Around 5% of the worlds population experience some form of hearing loss (World Health Organisation 2020). In the UK alone, there are an estimated 9 million people who are Deaf or hard of hearing (British Deaf Association 2020). For the Deaf native signer, a spoken language may be a second language, meaning their spoken language skills can vary immensely (Holt 1993). Therefore, sign languages are the preferred form of communication for the Deaf communities.

Sign languages possess different grammatical structure and syntax to spoken languages (Stokoe 1980). As highlighted in Fig. 1, the translation between spoken and sign languages requires a change in order and structure due to their non-monotonic relationship. Sign languages are also 3D visual languages, with position and movement relative to the body playing an important part of communication. In order to convey complex meanings and context, sign languages employ multiple modes of articulation. The manual features of hand shape and motion are combined with the non-manual features of facial expressions, mouthings and upper body posture (Sutton-Spence and Woll 1999).

Sign languages have long been researched by the vision community (Bauer et al. 2000; Starner and Pentland 1997; Tamura and Kawasaki 1988). Previous research has focused on the recognition of sign languages and the subsequent translation to spoken language. Although useful, this is a technology more applicable to allowing the hearing to understand the Deaf, and often not that helpful for the Deaf community. The opposite task of sign language production (SLP) is far more relevant to the Deaf. Automatically translating spoken language into sign language could increase the sign language content available in the predominately hearing-focused world.

To be useful to the Deaf community, SLP must produce sequences of natural, understandable sign akin to a human

Communicated by Manuel J. Marin-Jimenez.

✉ Ben Saunders
b.saunders@surrey.ac.uk

[1] Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, UK

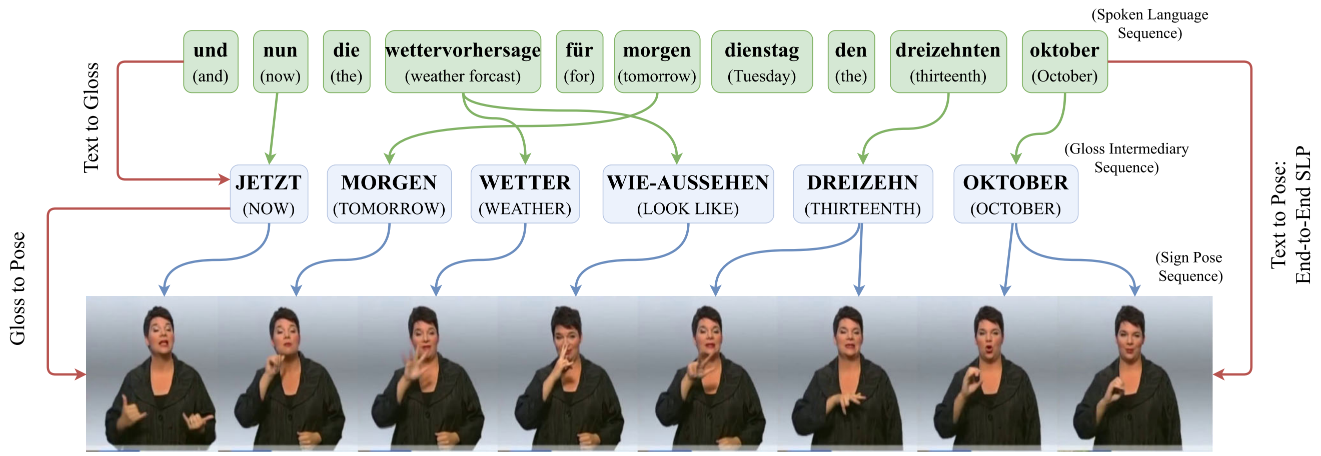

Fig. 1 Sign language production (SLP) example showing corresponding spoken language, gloss representation and sign language sequences. The *Text to Gloss*, *Gloss to Pose* and *Text to Pose* translation tasks are highlighted, where end-to-end SLP is a direct translation from spoken language to sign language, skipping the gloss intermediary. In this manuscript we use text to denote spoken language sequences

translator (Bragg et al. 2019). Previous deep learning-based SLP work has been limited to the production of concatenated isolated signs (Stoll 2020; Zelinka and Kanis 2020), with a focus solely on the manual features. These works also approach the problem in a fragmented Text to Gloss[1] and Gloss to Pose production (Fig. 1, left), where important context can be lost in the gloss bottleneck. However, the production of full sign sequences is a more challenging task, as there is no direct alignment between sign sequences and spoken language sentences. Ignoring non-manual features disregards the contextual and grammatical information required to fully understand the meaning of the produced signs (Valli and Lucas 2000). These works also produce only 2D skeleton data, lacking the depth channel to truly model realistic motion.

In this work, we present a Continuous 3D Multi-Channel Sign Language Production model, the first SLP network to translate from spoken language sentences to continuous 3D multi-channel sign language sequences in an end-to-end manner. This is shown on the right of Fig. 1 as a direct translation from source spoken language, without the need for a gloss intermediary. We propose a *Progressive Transformer* architecture that uses an alternative formulation of transformer decoding for continuous sequences, where there is no pre-defined vocabulary. We introduce a counter decoding technique to predict continuous sequences of variable length by tracking the production progress over time and predicting the end of sequence. Our sign pose productions contain both manual and non-manual features, increasing both the realism and comprehension.

To reduce the prediction drift often seen in continuous sequence production, we present several data augmentation methods. These create a more robust model and reduce the erroneous nature of auto-regressive prediction. Continuous prediction often results in a under-articulated output due to the problem of regression to the mean, and thus we propose the addition of adversarial training. A discriminator model conditioned on source spoken language is introduced to prompt a more realistic and expressive sign production from the progressive transformer. Additionally, due to the multimodal nature of sign languages, we also experiment with a mixture density network (MDN) modelling, utilising the progressive transformer outputs to paramatise a Gaussian mixture model.

To evaluate quantitative performance, we propose a back translation evaluation method for SLP, using a Sign Language Translation (SLT) back-end to translate sign productions back to spoken language. We evaluate on the challenging RWTH-PHOENIX-Weather-2014T (PHOENIX14T) dataset, presenting several benchmark results of both *Gloss to Pose* and *Text to Pose* configurations, to underpin future research. We also provide a user evaluation of our sign productions, to evaluate the comprehension of our SLP model. Finally, we share qualitative results to give the reader further insight into the models performance, producing accurate sign pose sequences of unseen text input.

The contributions of this paper can be summarised as:

- The first SLP model to translate from spoken language to continuous 3D sign pose sequences, enabled by a novel transformer decoding technique.
- An application of conditional adversarial training to SLP, for the production of realistic sign

[1] Glosses are a written representation of sign, defined as minimal lexical items.

- The combination of transformers and mixture density networks to model multimodal continuous sequences.
- Benchmark SLP results on the PHOENIX14T dataset and a new back translation evaluation metric, alongside a comprehensive Deaf user evaluation.

Preliminary versions of this work were presented in Saunders et al. (2020a; 2020b). This extended manuscript includes additional formulation and the introduction of a MDN modelling for expressive sign production. Extensive new quantitative and qualitative evaluation is provided to explore the capabilities of our approach, alongside a user study with Deaf participants to measure the comprehension of our produced sign language sequences.

The rest of this paper is organised as follows: We outline the previous work in SLP and surrounding areas in Sect. 2. Our progressive transformer network and proposed model configurations are presented in Sect. 3. Sect. 4 provides the experimental setup, with quantitative evaluation in Sect. 5 and qualitative evaluation in Sect. 6. Finally, we conclude the paper in Sect. 7 by discussing our findings and future work.

2 Related Work

To understand the sign language computational research landscape, we first outline the recent literature in sign language recognition (SLR) and SLT and then detail previous work in SLP. Sign languages reside at the intersection between vision and language, so we also review recent developments in neural machine translation (NMT). Finally, we provide background on the applications of adversarial training and mixture density networks (MDNs) to sequence tasks, specifically applied to human pose generation.

2.1 Sign Language Recognition and Translation

The goal of vision-based sign language research is to develop systems capable of recognition, translation and production of sign languages (Bragg et al. 2019). There has been prominent sign language computational research for over 30 years (Bauer et al. 2000; Starner and Pentland 1997; Tamura and Kawasaki 1988), with an initial focus on isolated sign recognition (Grobel and Assan 1997; Ozdemir et. al 2016) and a recent expansion to Continuous Sign Language Recognition (CSLR) (Camgoz et al. 2017; Chai et al. 2013; Koller et al. 2015). However, the majority of work has relied on manual feature representations (Cooper et al. 2012) and statistical temporal modelling (Vogler and Metaxas 1999).

Recently, larger sign language datasets have been released, such as RWTH-PHOENIX-Weather- 2014 (PHOENIX14) (Forster et al. 2014), Greek sign language (GSL) (Adaloglou 2019) and the Chinese Sign Language Recognition Dataset (Huang et al. 2018). These have enabled the application of deep learning approaches to CSLR, such as convolutional neural networks (CNNs) (Koller et al. 2016, 2019) and recurrent neural networks (RNNs) (Cui et al. 2017; Koller et al. 2017).

Expanding upon CSLR, Camgoz et al. (2018) introduced the task of SLT, aiming to directly translate sign videos to spoken language sentences. Due to the differing grammar and ordering between sign and spoken language (Stokoe 1980), SLT is a more challenging task than CSLR. The majority of work has utilised NMT networks for SLT (Camgoz 2018; Ko et al. 2019; Orbay and Akarun 2020; Yin 2020), translating directly to spoken language or via a gloss intermediary. Transformer based models are the current state-of-the-art in SLT, jointly learning the recognition and translation tasks (Camgoz et al. 2020b). The inclusion of multi-channel features have also been shown to reduce the dependence on gloss annotation in SLT (Camgoz et al. 2020a).

2.2 Sign Language Production

Previous research into SLP has focused on avatar-based techniques that generate realistic-looking sign production, but rely on pre-recorded phrases that are expensive to create (Ebling and Huenerfauth 2015; Glauert et al. 2006; McDonald et al. 2016; Zwitserlood et al. 2004). Non-manual feature production has been included in avatar generation, such as mouthings (Elliott et al. 2008) and head positions (Cox et al. 2002), but have been viewed as “stiff and emotionless” with an “absense of mouth patterns” (Kipp et al. 2011b). MoCap approaches have successfully produced realistic productions, but are expensive to scale (Pengfei and Huenerfauth 2010). Statistical Machine Translation (SMT) has also been applied to SLP (Kayahan and Gungor 2019; Kouremenos et al. 2018), relying on rules-based processing that can be difficult to encode.

Recently, there has been an increase in deep learning approaches to automatic SLP (Stoll 2020; Xiao et al. 2020; Zelinka and Kanis 2020). Stoll et al. (2020) presented a SLP model that used a combination of NMT and generative adversarial networks (GANs). The authors break the problem into three independent processes trained separately, producing a concatenation of isolated 2D skeleton poses mapped from sign glosses via a look-up table. As seen with other works, this production of isolated signs of a set length and order without realistic transitions results in robotic animations that are poorly received by the Deaf (Bragg et al. 2019). Contrary to Stoll et al. our work focuses on automatic sign production and learning the mapping between text and skeleton pose sequences directly, instead of providing this a priori.

The closest work to this paper is that of Zelinka and Kanis (2020), who use a neural translator to synthesise skeletal pose from text. A single 7-frame sign is produced for each input word, generating sequences with a fixed length and ordering that disregards the natural syntax of sign language. In contrast, our model allows a dynamic length of output sign sequence, learning the length and ordering of corresponding signs from the data, whilst using a progress counter to determine the end of sequence generation. Unlike Zelinka et al. who work on a proprietary dataset, we produce results on the publicly available PHOENIX14T, providing a benchmark for future SLP research.

Previous deep learning-based SLP works produce solely manual features, ignoring the important non-manuals that convey crucial context and meaning. Mouthings, in particular, are vital to the comprehension of most sign languages, differentiating signs that may otherwise be homophones. The expansion to non-manuals is challenging due to the required temporal coherence with manual features and the intricacies of facial movements. We expand production to non-manual features by generating synchronised mouthings and facial movements from a single model, for expressive and natural sign production.

2.3 Neural Machine Translation

NMT is the automatic translation from a source sequence to a target sequence of a differing language, using neural networks. To tackle this sequence-to-sequence task, RNNs were introduced by Cho et al. (2014), which iteratively apply a hidden state computation across each token of the sequence. This was later developed into encoder-decoder architectures (Sutskever et al. 2014), which map both sequences to an intermediate embedding space. Encoder model have the drawback of a fixed sized representation of the source sequence. This problem was overcome by an attention mechanism that facilitated a soft-search over the source sentence for the most useful context (Bahdanau et al. 2015).

Transformer networks were recently proposed by Vaswani et al. (2017), achieving state-of-the-art performance in many NMT tasks. Transformers use self-attention mechanisms to generate representations of entire sequences with global dependencies. Multi-headed attention (MHA) layers are used to model different weighted combinations of each sequence, improving the representational power of the model. A mapping between the source and target sequence representations is created by an encoder-decoder attention, learning the sequence-to-sequence task.

Transformers have achieved impressive results in many classic natural language processing (NLP) tasks such as language modelling (Dai et al. 2019; Zhang et al. 2019) and sentence representation (Devlin et al. 2018), alongside other domains including image captioning (Zhou et al. 2018) and action recognition (Girdhar et al. 2019). Related to this work, transformer networks have been applied to many continuous output tasks such as speech synthesis (Ren et la. 2019b), music production (Huang et al. 2018) and speech recognition (Povey et al. 2018).

Applying sequence-to-sequence methods to continuous output tasks is a relatively underresearched problem. In order to determine sequence length of continuous outputs, previous works have used a fixed output size (Zelinka and Kanis 2020), a binary end-of-sequence (EOS) flag (Graves 2013) or a continuous representation of an EOS token (Mukherjee et al. 2019). We propose a novel counter decoding technique that predicts continuous sequences of variable length by tracking the production progress over time and implicitly learning the end of sequence.

2.4 Adversarial Training

Adversarial training is the inclusion of a discriminator model designed to improve the realism of a generator by critiquing the productions (Goodfellow et al. 2014). GANs, which generate data using adversarial techniques, have produced impressive results when applied to image generation (Isola et al. 2017; Radford et al. 2015; Zhu et al. 2017) and, more recently, video generation tasks (Tulyakov et al. 2018; Vondrick et al. 2016). Conditional GANs (Mirza and Osindero 2014) extended GANs with generation conditioned upon specific data inputs.

GANs have also been applied to natural language tasks (Kevin et al. 2017; Press et al. 2017; Zhang et al. 2016). Specific to NMT, Wu et al. (2017) designed Adversarial-NMT, complimenting the original NMT model with a CNN based adversary, and Yang et al. (2017) proposed a GAN setup with translation conditioned on the input sequence.

Specific to human pose generation, adversarial discriminators have been used for the production of realistic pose sequences (Cai et al. 2018; Chan et al. 2019; Ren et al. 2019a). Ginosar et al. (2019) show that the task of generating skeleton motion suffers from regression to the mean, and adding an adversarial discriminator can improve the realism of gesture production. Lee et al. (2019) use a conditioned discriminator to produce smooth and diverse human dancing motion from music. In this work, we use a conditional discriminator to produce expressive sign pose outputs from source spoken language.

2.5 Mixture Density Networks

Mixture density networks (MDNs) create a multimodal prediction to better model distributions that may not be modelled fully by a single density distribution. MDNs combine a conventional neural network with a mixture density model, modelling an arbitrary conditional distribution via a direct parametrisation (Bishop 1994). The neural network estimates the density components, predicting the weights and statistics of each distribution.

MDNs are often used for continuous sequence generation tasks due to their ability to model sequence uncertainty (Schuster 2000). Graves et al. (2013) combined an MDN with a RNN for continuous handwriting generation, which has been expanded to sketch generation (Ha and Eck 2018; Zhang et al. 2017) and reinforcement learning (Ha and Schmidhuber 2018). MDNs have also been applied to speech synthesis (Wang et al. 2017), future prediction (Makansi et al. 2019) and driving prediction (Hu et al. 2018).

MDNs have also been used for human pose estimation, either to predict multiple hypotheses (Chen and Hee 2019), to better model uncertainty (Prokudin et al. 2018; Varamesh and Tuytelaars 2020) or to deal with occlusions (Ye and Kim 2018). To the best of our knowledge, this work is the first to combine transformers with MDNs for sequence modelling. We employ MDNs to capture the natural variability in sign languages and to model production using multiple distributions.

3 Continuous 3D Sign Language Production

In this section, we introduce our SLP model, which learns to translate spoken language sentences to continuous sign pose sequences. Our objective is to learn the conditional probability $p(Y|X)$ of producing a sequence of signs $Y = (y_1, \ldots, y_U)$ with U frames, given a spoken language sentence $X = (x_1, \ldots, x_T)$ with T words. Glosses could also be used as source input, replacing the spoken language sentence as an intermediary. In this work we represent sign language as a sequence of continuous skeleton poses modelling the 3D coordinates of a signer, of both manual and non-manual features.

Producing a target sign language sequence from a reference spoken language sentence poses several challenges. Firstly, there exists a non-monotic relationship between spoken and sign language, due to the different grammar and syntax in the respective domains (Stokoe 1980). Secondly, the target signs inhabit a continuous vector space, requiring a differing representation to the discrete space of text and disabling the use of classic end of sequence tokens. Finally, there are multiple channels encompassed within sign that must be produced concurrently, such as the manual (hand shape and position) and non-manual features (mouthings and facial expressions) (Pfau et al. 2010).

To address the production of continuous sign sequences, we propose a *Progressive Transformer* model that enables translation from a symbolic to a continuous sequence domain (PT in Fig. 2). We introduce a counter decoding that enables the model to track the progress of sequence generation and implicitly learn sequence length given a source sentence. We also propose several data augmentation techniques that reduce the impact of prediction drift.

To enable the production of expressive sign, we introduce an adversarial training regime for SLP, supplementing the progressive transformer generator with a conditional adversarial discriminator, (Disc in Fig. 2). To enhance the capability to model multimodal distributions, we also propose a MDN formulation of the SLP network. In the remainder of this section we describe each component of the proposed architecture in detail.

3.1 Progressive Transformer

We build upon the classic transformer (Vaswani et al. 2017), a model designed to learn the mapping between symbolic source and target languages. We modify the architecture to deal with continuous output representations such as sign language, alongside introducing a counter decoding technique that enables sequence prediction of variable lengths. Our SLP model tracks the progress of continuous sequence production through time, hence the name *Progressive Transformer*.

In this work, Progressive Transformers translate from the symbolic domains of gloss or spoken language to continuous 3D sign pose sequences. These sequences represent the motion of a signer producing a sign language sentence. The model must produce sign pose outputs that express an accurate translation of the given input sequence and embody a realistic sign pose sequence. Our model consists of an encoder-decoder architecture, where the source sequence is first encoded to a latent representation before being mapped to a target output during decoding in an auto-regressive manner.

3.1.1 Source Embeddings

As per the standard NMT pipeline, we first embed the symbolic source tokens, x_t, via a linear embedding layer (Mikolov et al. 2013). This represent the one-hot-vector in a higher-dimensional space where tokens with similar meanings are closer. This embedding, with weight, W, and bias, b, can be formulated as:

$$w_t = W^x \cdot x_t + b^x \tag{1}$$

where w_t is the vector representation of the source tokens.

As with the original transformer implementation, we apply a temporal encoding layer after the source embedding, to provide temporal information to the network. For the encoder, we apply positional encoding, as:

$$\hat{w}_t = w_t + \text{PositionalEncoding}(t) \tag{2}$$

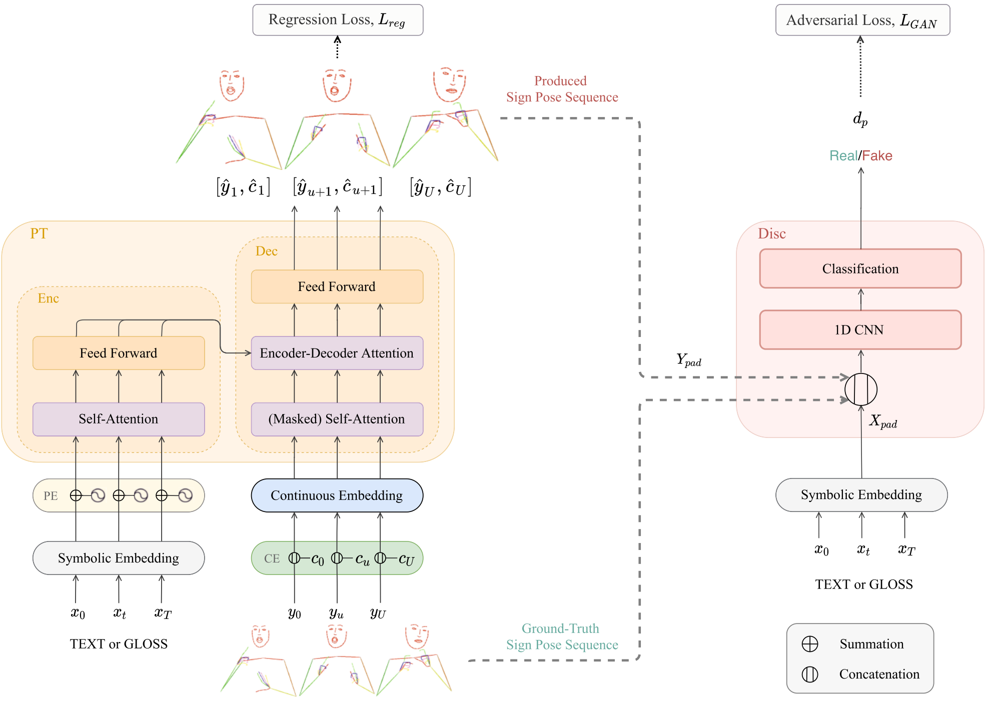

Fig. 2 Architecture details of our *Progressive Transformer* and *Conditional Discriminator* network. The *Progressive Transformer* produces a sign pose sequence, $\hat{y}_{1:U}$, and respective counter values, $\hat{c}_{1:U}$, from source spoken language, $\hat{x}_{1:T}$, in an auto-regressive prediction. The *Conditional Discriminator* takes as input either ground-truth or produced sign pose sequences alongside the respective source spoken language, and predicts a single realism scalar, d_p. The network is trained end-to-end via a weighted combination of regression loss, L_{reg}, and adversarial loss, L_{GAN}. (*PT* progressive transformer, *PE* positional encoding, *CE* counter encoding, *Disc* discriminator)

where PositionalEncoding is a predefined sinusoidal function conditioned on the relative sequence position t (Vaswani et al. 2017).

3.1.2 Target Embeddings

The target sign sequence consists of 3D joint positions of the signer. Due to their continuous nature, we first apply a novel temporal encoding, which we refer to as counter encoding (CE in Fig. 2). The counter, c, holds a value between 0 and 1, representing the frame position relative to the total sequence length. The target joints, y_u, are concatenated with the respective counter value, c_u, formulated as:

$$j_u = [y_u, c_u] \tag{3}$$

where c_u is the counter value for frame u, as a proportion of sequence length, U. At each time-step, counter values, $\hat{c}$, are predicted alongside the skeleton pose, as shown in Fig. 3, with sequence generation concluded once the counter reaches 1. We call this process *Counter Decoding*, determining the progress of sequence generation and providing a way to predict the end of sequence without the use of a tokenised vocabulary.

The counter value provides the model with information relating to the length and speed of each sign pose sequence, determining the sign duration. At inference, we drive the sequence generation by replacing the predicted counter value, $\hat{c}$, with the linear timing information, c^{*}, to produce a stable output sequence.

These counter encoded joints, j_u, are next passed through a linear embedding layer, which can be formulated as:

$$\hat{j}_u = W^y \cdot j_u + b^y \tag{4}$$

where $\hat{j}_u$ is the embedded 3D joint coordinates of each frame, y_u.

3.1.3 Encoder

The progressive transformer encoder, E_{PT}, consists of a stack of L identical layers, each containing 2 sub-layers. Given the temporally encoded source embeddings, $\hat{w}_t$, a MHA sub-layer first generates a weighted contextual representation, performing multiple projections of scaled dot-product attention. This aims to learn the relationship between each token of the sequence and how relevant each time step is in the context of the full sequence. Formally, scaled dot-product attention outputs a vector combination of values, V,

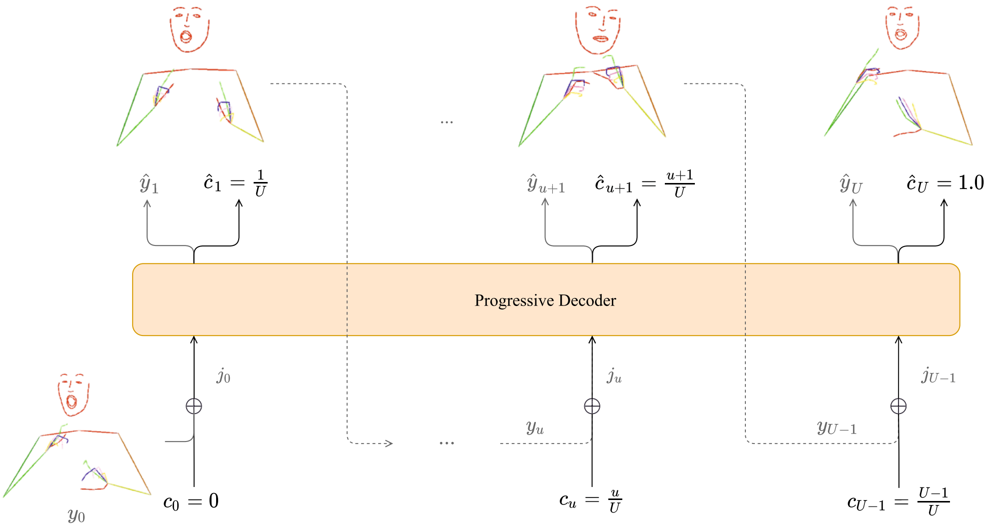

Fig. 3 Counter decoding example, showing the simultaneous auto-regressive prediction of continuous sign pose, $\hat{y}_u$, and counter value, $\hat{c}_u \in \{0 : 1\}$. A counter value of 1, $\hat{c} = 1.0$, denotes end of sequence and decoding is stopped

weighted by the relevant queries, Q, keys, K, and dimensionality, d_k:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{5}$$

MHA uses multiple self-attention heads, h, to generate parallel mappings of the same queries, keys and values, each with varied learnt parameters. This allows different representations of the input sequence to be generated, learning complementary information in different sub-spaces. The outputs of each head are then concatenated together and projected forward via a final linear layer, as:

$$\begin{aligned} &\text{MHA}(Q, K, V) = [head_1, ..., head_h] \cdot W^O, \\ &\text{where} \quad \cdot\, head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \tag{6}$$

and W^O,W_i^Q,W_i^K and W_i^V are weights related to each input variable.

The outputs of MHA are then fed into a second sub-layer of a non-linear feed-forward projection. A residual connection (He et al. 2016) and subsequent layer norm (Ba et al. 2016) is employed around each of the sub-layers, to aid training. The final encoder output can be formulated as:

$$h_t = E_{PT}(\hat{w}_t | \hat{w}_{1:T}) \tag{7}$$

where h_t is the contextual representation of the source sequence.

3.1.4 Decoder

The progressive transformer decoder (D_{PT}) is an auto-regressive model that produces a sign pose frame at each time-step, alongside the previously described counter value. Distinct from symbolic transformers, our decoder produces continuous sequences.

The counter-concatenated joint embeddings, $\hat{j}_u$, are used to represent the sign pose of each frame. Firstly, an initial MHA sub-layer is applied to the joint embeddings, similar to the encoder but with an extra masking operation. The masking of future frames prevents the model from attending to subsequent time steps that are yet to be decoded.

A further MHA mechanism is then used to map the symbolic representations from the encoder to the continuous domain of the decoder. A final feed forward sub-layer follows, with each sub-layer followed by a residual connection and layer normalisation as in the encoder. The output of the progressive decoder can be formulated as:

$$[\hat{y}_u, \hat{c}_u] = D_{PT}(\hat{j}_{1:u-1}, h_{1:T}) \tag{8}$$

where $\hat{y}_u$ corresponds to the 3D joint positions representing the produced sign pose of frame u and $\hat{c}_u$ is the respective counter value. The decoder learns to generate one frame at a time until the predicted counter value, $\hat{c}_u$, reaches 1, determining the end of sequence as seen in Fig. 3. The model is trained using the mean squared error (MSE) loss between the predicted sequence, $\hat{y}_{1:U}$, and the ground truth, $y^*_{1:U}$:

$$L_{MSE} = \frac{1}{U}\sum_{i=1}^{u}(y^*_{1:U} - \hat{y}_{1:U})^2 \tag{9}$$

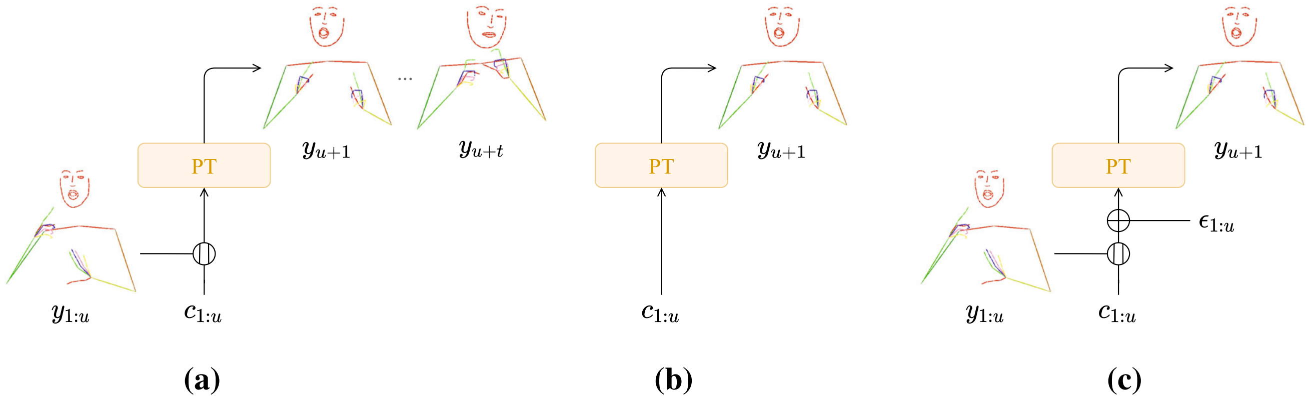

Fig. 4 Data augmentation techniques to reduce prediction drift and create a more robust SLP model. **a** Future prediction is the prediction of multiple future frames. **b** Just counter uses only the counter positions as input. **c** Gaussian noise applies noise to the input skeleton pose. (*PT* progressive transformer)

At inference time, the full sign pose sequence, $\hat{y}_{1:U}$, is produced in an auto-regressive manner, with predicted sign frames used as input to future time steps. Once the predicted counter value reaches 1, decoding is complete and the full sign sequence is produced.

3.2 Data Augmentation

Auto-regressive sequential prediction can often suffer from prediction drift, with erroneous predictions accumulating over time. As transformer models are trained to predict the next time-step using ground truth inputs, they are often not robust to noise in predicted inputs. The impact of drift is heightened for an SLP model due to the continuous nature of skeleton poses. As neighbouring frames differ little in content, a model can learn to just copy the previous ground truth input and receive a small loss penalty.

At inference time, with predictions based off previous outputs, errors are quickly propagated throughout the entire sign sequence production. To overcome the problem of prediction drift, in this section we propose various data augmentation approaches, namely *Future Prediction*, *Just Counter* and *Gaussian Noise*.

3.2.1 Future Prediction

Our first data augmentation method is conditional future prediction, requiring the model to predict more than just the next frame in the sequence. Figure 4a shows an example future prediction of $y_{u+1}, \ldots, y_{u+t}$ from the input $y_{1:u}$. Due to the short time step between neighbouring frames, the movement between frames is small and the model can learn to just predict the previous frame with some noise. Predicting more frames into the future means the movement of sign has to be learnt, rather than simply copying the previous frame. At inference time, only the next frame prediction is considered for production.

3.2.2 Just Counter

Inspired by the memorisation capabilities of transformer models, we next propose a pure memorisation approach to sign production. Contrary to the usual input of full skeleton joint positions, only the counter values are provided as target input. Figure 4b demonstrates the input of $c_{1:u}$ as opposed to $y_{1:u}$. The model must decode the target sign pose sequence solely from the counter positions, having no knowledge of the previous frame positions. This halts the reliance on the ground truth joint embeddings it previously had access to, forcing a deeper understanding of the source spoken language and a more robust production. The network setup is also now identical at both training and inference, with the model having to generalise only to new data rather than new prediction inputs.

3.2.3 Gaussian Noise

Our final augmentation technique is the application of noise to the input sign pose sequences during training, increasing the variety of data. This is shown in Fig. 4c, where the input $y_{1:u}$ is summed with noise $\epsilon_{1:u}$. At each epoch, distribution statistics of each joint are collected, with randomly sampled noise applied to the inputs of the next epoch. The addition of Gaussian noise causes the model to become more robust to prediction input error, as it must learn to correct the augmented inputs back to the target outputs. At inference time, the model is more used to noisy inputs, increasing the ability to adapt to erroneous predictions and correct the sequence generation.

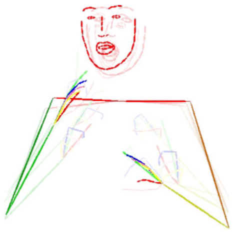

Fig. 5 An average of multiple valid sign poses (blurred) results in an under-articulated production due to the problem of regression to the mean

3.3 Adversarial Training

Sign languages contain naturally varied movements, as each signer produces sign sequences with slightly different articulations and movements. Realistic sign consists of subtle and precise movements of the full body, which can easily be lost when training solely to minimise joint error [e.g. Eq. (9)]. SLP models trained solely for regression can lack pose articulation, suffering from the problem of regression to the mean. Specifically, average hand shapes are produced with a lack of comprehensive motion, due to the high variability of these joints. Figure 5 highlights this problem, as the average of the valid blurred poses results in an under-articulated mean production that does not convey the required meaning.

To address under-articulation, we propose an adversarial training mechanism for SLP. As shown in Fig. 2, we introduce a conditional discriminator, D, alongside the SLP generator, G. We frame SLP as a min-max game between the two networks, with D evaluating the realism of G's productions. We use the previously described progressive transformer architecture as G (Fig. 2 left) to produce sign pose sequences. We build a convolutional network for D (Fig. 6), trained to produce a single scalar that represents realism, given a sign pose sequence and corresponding source input sequence. These models are co-trained in an adversarial manner, which can be formalised as:

$$\begin{aligned}&\min_{G}\max_{D}\ L_{GAN}(G,D)\\&\quad=\mathbb{E}[\log D(Y^{*}\mid X)]+\mathbb{E}[\log(1-D(G(X)\mid X))]\end{aligned}\tag{10}$$

where Y^* is the ground truth sign pose sequence, $y^*_{1:U}$, $G(X)$ equates to the produced sign pose sequence, $\hat{Y} = \hat{y}_{1:U}$, and X is the source spoken language.

3.3.1 Generator

Our generator, G, learns to produce sign pose sequences given a source spoken language sequence, integrating the progressive transformer into a GAN framework. Contrary to the standard GAN implementation, we require sequence generation to be conditioned on a specific source input. Therefore, we remove the traditional noise input (Goodfellow et al. 2014), and generate a sign pose sequence conditioned on the source sequence, taking inspiration from conditional GANs (Mirza and Osindero 2014).

We propose training G using a combination of loss functions, namely regression loss, L_{Reg}, [Eq. (9)] and adversarial loss, L^G_{GAN}, [Eq. (10)]. The total loss function is a weighted combination of these losses, as:

$$L^G = \lambda_{Reg} L_{Reg}(G) + \lambda_{GAN} L^G_{GAN}(G,D) \tag{11}$$

where λ_{Reg} and λ_{GAN} determine the importance of each loss function during training.

3.3.2 Discriminator

We present a conditional adversarial discriminator, D, used to differentiate generated sign pose sequences, $\hat{Y}$, and ground-truth sign pose sequences, Y^*, conditioned on the source spoken language sequence, X. Figure 6 shows an overview of the discriminator architecture.

For each pair of source-target sequences, (X, Y), of either generated or real sign pose, the aim of D is to produce a single scalar, $d_p \in (0, 1)$. This represents the probability that the sign pose sequence originates from the data, Y^*:

$$d_p = P(Y = Y^* \mid X, Y) \in (0, 1) \tag{12}$$

The sequence counter value is removed before being input to the discriminator, in order to critique only the sign content. Due to the variable frame lengths of the sign sequences, we apply padding to transform them to a fixed length, U_{max}, the maximum frame length of target sequences found in the data:

$$Y_{pad} = [Y_{1:U}, \varnothing_{U:U_{max}}] \tag{13}$$

where Y_{pad} is the sign pose sequence padded with zero vectors, $\varnothing$, enabling convolutions upon the now fixed size tensor. In order to condition D on the source spoken language, we first embed the source tokens via a linear embedding layer. Again to deal with variable sequence length, these embeddings are also padded to a fixed length T_{max}, the maximum source sequence length:

$$X_{pad} = [W^X \cdot X_{1:T} + b^X, \varnothing_{T:T_{max}}] \tag{14}$$

where W^X and b^X are the weight and bias of the source embedding respectively and $\varnothing$ is zero padding. As shown in the centre of Fig. 6, the source representation is then concatenated with the padded sign pose sequence, to create the conditioned features, H:

$$H = [Y_{pad}, X_{pad}] \tag{15}$$

N 1D convolutional filters are passed over the sign pose sequence, analysing the local context to determine the temporal continuity of the signing motion. This is more effective

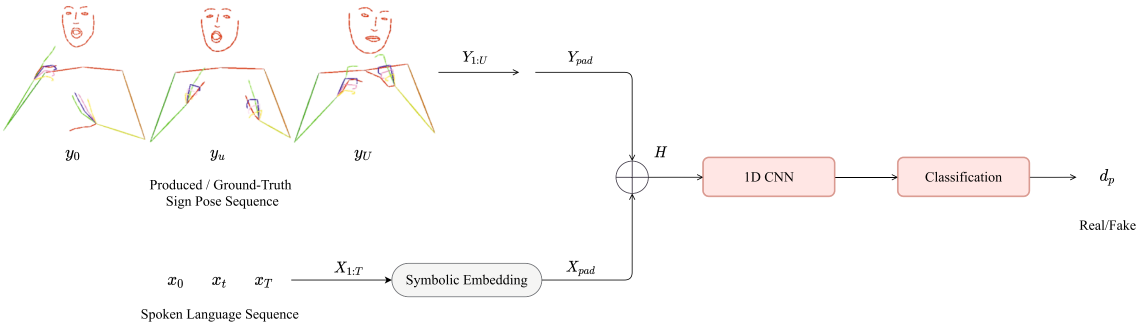

Fig. 6 Architecture details of our conditional discriminator model. Sign pose, $Y_{1:U}$, is concatenated with source text, $X_{1:T}$, and projected to a single scalar, d_p, that represents the realism of the sign pose sequence

than a frame level discriminator at determining realism, as a mean hand shape is a valid pose for a single frame, but not consistently over a large temporal window. Leaky ReLU activation (Maas et al. 2013) is applied after each layer, promoting healthy gradients during training. A final feed-forward linear layer and sigmoid activation projects the combined features down to the single scalar, d_p, representing the probability that the sign pose sequence is real.

We train D to maximise the likelihood of producing $d_p = 1$ for real sign sequences and $d_p = 0$ for generated sequences. This objective can be formalised as maximising Eq. (10), resulting in the loss function $L^D = L^D_{GAN}(G, D)$. At inference time, D is discarded and G is used to produce sign pose sequences in an auto-regressive manner as in Sect. 3.1.

3.4 Mixture Density Networks

The previously-described model architectures generate deterministic productions, with each model predicting a single non-stochastic pose at each time step. A single prediction is unable to model any uncertainty or variation that is found in continuous sequence generation tasks like SLP. The deterministic modelling of sequences can again result in a mean, under-articulated production with no room for expression or variability.

To overcome the issues of deterministic prediction, we propose the use of a mixture density network (MDN) to model the variation found in sign language. As shown in Fig. 7, multiple distributions are used to parameterise the entire prediction subspace, with each mixture component modelling a separate valid movement into the future. This enables prediction of all valid signing motions and their corresponding uncertainty, resulting in a more expressive production.

3.4.1 Formulation

MDNs use a neural network to parameterise a mixture distribution (Bishop 1994). A subset of the network predicts the mixture weights whilst the rest generates the parameters of each of the individual mixture distributions. We use our previously described progressive transformer architecture, but amend the output to model a mixture of Gaussian distributions. Given a source token, x_t, we can model the conditional probability of producing the sign pose frame, y_u, as:

$$p(y_u|x_t) = \sum_{i=1}^{M} \alpha_i(x_t)\phi_i(y_u|x_t) \tag{16}$$

where M is the number of mixture components used in the MDN. $\alpha_i(x_t)$ is the mixture weight of the i^{th} distribution, regarded as a prior probability of the sign pose frame being generated from this mixture component. $\phi_i(y_u|x_t)$ is the conditional density of the sign pose for the i^{th} mixture, which can be expressed as a Gaussian distribution:

$$\phi_i(y_u|x_t) = \frac{1}{\sigma_i(x_t)\sqrt{2\pi}}\, exp^{\frac{\|y_u - \mu_i(x_t)\|^2}{2\sigma_i(x_t)^2}} \tag{17}$$

where $\mu_i(x_t)$ and $\sigma_i(x_t)$ denote the mean and variance of the i^{th} distribution, respectively. The parameters of the MDN are predicted directly by the progressive transformer, as shown in Fig. 7. The mixture coefficients, $\alpha(x_t)$, are passed through a softmax activation function to ensure each lies in the range [0, 1] and sum to 1. An exponential function is applied to the variances, $\sigma(x_t)$, to ensure a positive output.

3.4.2 Optimisation

During training, we minimise the negative log likelihood of the ground truth data coming from our predicted mixture

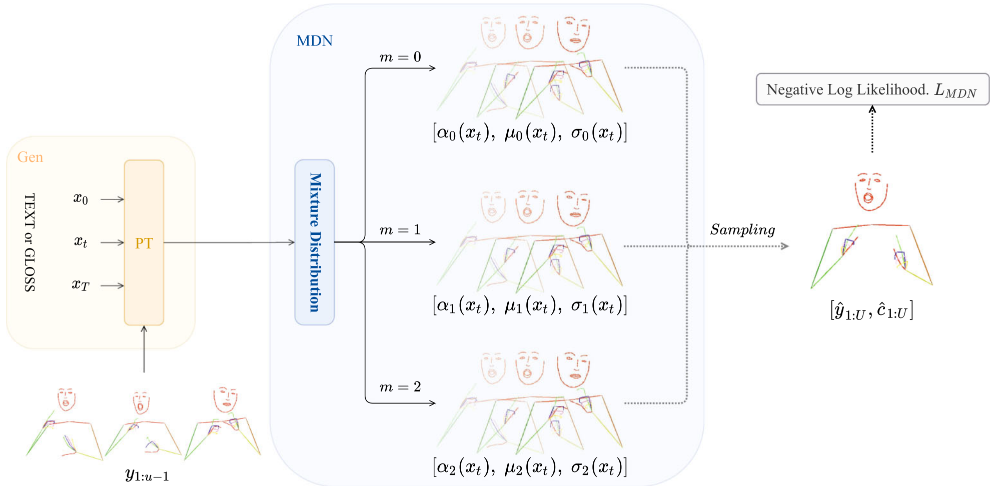
MDN
Gen
TEXT or GLOSS
x_0
x_t
x_T
PT
$y_{1:u-1}$
Mixture Distribution
$m = 0$
$m = 1$
$m = 2$
$[\alpha_0(x_t),\ \mu_0(x_t),\ \sigma_0(x_t)]$
$[\alpha_1(x_t),\ \mu_1(x_t),\ \sigma_1(x_t)]$
$[\alpha_2(x_t),\ \mu_2(x_t),\ \sigma_2(x_t)]$
Sampling
Negative Log Likelihood. L_{MDN}
$[\hat{y}_{1:U}, \hat{c}_{1:U}]$

Fig. 7 An overview of our Mixture Density Network (MDN) network. Multiple mixture distributions, m, are parameterised by the progressive transformer (PT) outputs, taking input source spoken language and previous sign pose frames. An output sign pose is sampled from the mixture distributions, producing an expressive and variable sign language sequence. The network is trained end-to-end with a negative log likelihood, L_{MDN}

distribution. This can be formulated as:

$$L_{MDN} = -\sum_{u=1}^{U} \log\, p(y_u|x_t)$$
$$= -\sum_{u=1}^{U} \log \sum_{i=1}^{M} \alpha_i(x_t)\phi_i(y_u|x_t) \tag{18}$$

where U is the number of frames in the produced sign pose sequence and M is the number of mixture components.

3.4.3 Sampling

At inference time, we sample sign pose productions from the mixture density computed in Eq. (16), as shown in Fig. 7. Firstly, we select the most likely distribution for this source token, x_t, from the mixture weights, $i_{max} = argmax_i\ \alpha_i(x_t)$. From this chosen distribution, we sample the sign pose, predicting $\mu_{i_{max}}(x_t)$ as a valid pose. To ensure there is no jitter in the sign pose predictions, we set $\sigma(x_t) = 0$. This avoids the large variation in small joint positions a large sigma would create, particularly for the hands.

To predict a sequence of multiple time steps, we sample each frame from the mixture density model in an auto-regressive manner as in Sect. 3.1. The sampled sign frames are used as input to future transformer time-steps, to produce the full sign pose sequence, $\hat{y}_{1:U}$.

3.4.4 MDN + Adversarial

The MDN can also be combined with our adversarial training regime outlined in Sect. 3.3. The MDN model is formulated as the adversarial generator pitched against an unchanged conditional discriminator, where a sampled sign pose is used as discriminator input. Again, the final loss function is a weighted combination of the negative log-posterior loss [Eq. (18)] and the adversarial generator loss [Eq. (10)], as:

$$L^G_{MDN} = \lambda_{MDN} L_{MDN}(G) + \lambda_{GAN} L^G_{GAN}(G, D) \tag{19}$$

At inference time, the discriminator model is discarded and a sign pose sequence is sampled from the resulting mixture distribution, as previously explained.

3.5 Sign Pose Sequence Outputs

Each of these model configurations are trained to produce a sign pose sequence, $\hat{y}_{1:U}$, given a source spoken language input, $x_{1:T}$. Animating a video from this skeleton sequence is a trivial task, plotting the joints and connecting the relevant bones, with timing information provided from the progressive transformer counter. These 3D joints can subsequently be used to animate an avatar (Kipp et al. 2011a; McDonald et al. 2016) or condition a GAN (Chan et al. 2019).

Even though the produced sign pose sequence is a valid translation of the given text, it may be signed at a different speed than that found in the reference data. This is not incorrect, as every signer signs with a varied motion and speed, with our model having its own cadence. However, in order to ease the visual comparison with reference sequences, we apply dynamic time warping (DTW) (Berndt and Clifford 1994) to temporally align the produced sign pose sequences. This action does not amend the content of the productions, only the temporal coherence for visualisation.

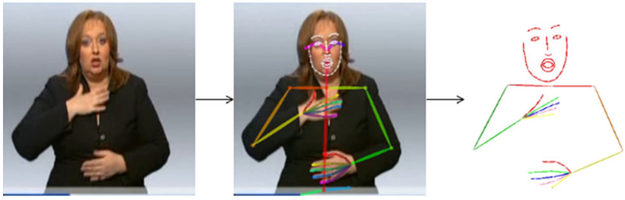

Fig. 8 Skeleton pose extraction, using 2D human pose estimation (Cao et al. 2017) and 2D to 3D mapping (Zelinka and Kanis 2020)

Although our focus has not been on building a real-time system, our current implementation is near real-time and a spoken language sentence can be translated to a sign language video within seconds. However, the nature of translation requires a delay as the context of a whole sentence is needed before it can be translated. As such, the small delay introduced by the automatic system does not present a significant further delay.

4 Experimental Setup

In this section, we outline our experimental setup, detailing the dataset, evaluation metrics and model configuration. We also introduce the back translation evaluation metric and evaluation protocols.

4.1 Dataset

In this work, we use the publicly available PHOENIX14T dataset introduced by Camgoz et al. (2018), a continuous SLT extension of the original PHOENIX14 corpus (Forster et al. 2014), becoming the benchmark for SLT research. This corpus includes parallel German Sign Language—Deutsche Gebärdensprache (DGS) videos and German translation sequences with redefined segmentation boundaries generated using the forced alignment approach of Koller et al. (2016). 8257 videos of 9 different signers are provided, with a vocabulary of 2887 German words and 1066 different sign glosses. We use the original training, validation and testing split as proposed by Camgoz et al. (2018).

We train our SLP network to generate sequences of 3D skeleton pose representing sign language, as shown in Fig. 8. 2D upper body joint and facial landmark positions are first extracted using OpenPose (Cao et al. 2017). We then use the skeletal model estimation improvements presented in Zelinka and Kanis (2020) to lift the 2D upper body joint positions to 3D. Finally, we apply skeleton normalisation similar to Stoll et al. (2020), with face coordinates scaled to a consistent size and centered around the nose joint.

Table 1 Ground-truth back translation results for *Manual*, *Non-Manual* and *Manual* + *Non-Manual* skeleton pose representations

Representation	DEV SET BLEU-4	TEST SET BLEU-4
Manual	11.05	9.97
Non-Manual	8.65	9.18
Manual + Non-Manual	11.44	11.01

4.2 Back Translation Evaluation

The evaluation of a continuous sequence generation model is a difficult task, with previous SLP evaluation metrics of MSE (Zelinka and Kanis 2020) falling short of a true measure of sign understanding. In this work, we propose back-translation as a means of SLP evaluation, translating back from the produced sign pose sequences to spoken language. This provides an automatic measure of how understandable the productions are, and the amount of translation content that is preserved. We find a close correspondence between back translation score and the visual production quality and liken it to the wide use of the inception score for generative models which uses a pre-trained classifier (Salimans et al. 2016). Similarly, recent SLP work has used an SLR discriminator to evaluate isolated skeletons (Xiao et al. 2020), but did not measure the translation performance. Back translation is a relative evaluation metric, best used to compare between similar model configurations. If the chosen SLT model is amended, absolute model performances will likely also change. However, as we have seen in our experimentation, the relative performance comparisons between models remain consistent. This ensures that comparison results between models remains valid.

We use the state-of-the-art SLT system (Camgoz et al. 2020b) as our back translation model, modified to take sign pose sequences as input. We build a sign language transformer model with 1 layer, 2 heads and an embedding size of 128. This is also trained on the PHOENIX14T dataset, ensuring a robust translation from sign to text. We generate spoken language translations of the produced sign pose sequences and compute BLEU and ROUGE scores. We provide BLEU n-grams from 1 to 4 for completeness.

We build multiple SLT models trained with various skeleton pose representations, namely *Manual* (Body), *Non-Manual* (Face) and *Manual* + *Non-Manual*. We evaluate the back translation performance for each configuration, to see how understandable the representation is and the amount of spoken language that can be recovered. As seen in Table 1, the *Manual* + *Non-Manual* configuration achieves the best back translation result, with *Non-Manual* achieving a significantly lower result. This demonstrates that manual and non-manual features contain complementary information when translat-

Table 2 Text to Gloss translation results of our transformer architecture, compared to that of Stoll et al. (2020)

	DEV SET					TEST SET				
Approach	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Stoll et al. (2020)	16.34	22.30	32.47	50.15	48.42	15.26	21.54	32.25	50.67	48.10
Ours	20.23	27.36	38.21	55.65	55.41	19.10	26.24	37.10	55.18	54.55

ing back to spoken language and supports our use of a multi-channel sign pose representation.

As seen in our quantitative experiments in Sect. 5, our sign production sequences can achieve better back translation performance than the original ground truth skeleton data. We believe this is due to a smoothing of the training data during production, as the original data contains artifacts either from 2D pose estimation, the 2D-to-3D mapping or the quality of the data itself. As our model learns to generate a temporally continuous production without these artifacts, our sign pose is significantly smoother than the ground truth. This explains the higher back translation performance from production compared to the ground truth data.

4.3 Evaluation Protocols

With back translation as an evaluation metric, we now set SLP evaluation protocols on the PHOENIX14T dataset. These can be used as measures for ablation studies and benchmarks for future work.

Text to Gloss (T2G): The first evaluation protocol is the symbolic translation between spoken language and sign language representation. This task is a measure of the translation into sign language grammar, an initial task before a pose production. This can be measured with a direct BLEU and ROUGE comparison, without the need for back translation.

Gloss to Pose (G2P): The second evaluation protocol evaluates the SLPs models capability to produce a continuous sign pose sequence from a symbolic gloss representation. This task is a measure of the production capabilities of a network, without requiring translation from spoken language.

Text to Pose (T2P): The final evaluation protocol is full end-to-end translation from a spoken language input to a sign pose sequence. This is the true measure of the performance of an SLP system, consisting of jointly performing translation to sign and a production of the sign sequence. Success on this task enables SLP applications in domains where expensive gloss annotation is not available.

4.4 Model Configuration

In the following experiments, our progressive transformer model is built with 2 layers, 4 heads and an embedding size of 512, unless stated otherwise. All parts of our network are trained with Xavier initialisation from scratch (Glorot and Bengio 2010), Adam optimization with default parameters (Kingma and Ba 2014) and a learning rate of 10^{-3}. We use a plateau learning rate scheduler with a patience of 7 epochs, a decay rate of 0.7 and a minimum learning rate of 2×10^{-4}. Our code is based on Kreutzer et al. 's NMT toolkit, JoeyNMT (2019), and implemented using PyTorch (Paszke et al. 2017).

5 Quantitative Evaluation

In this section, we present a thorough quantitative evaluation of our SLP model, providing results and subsequent discussion. We first conduct experiments using the *Text to Gloss* setup. We then evaluate the *Gloss to Pose* and the end-to-end *Text to Pose* setups. Finally, we provide results of our user study with Deaf participants.

5.1 Text to Gloss Translation

To provide a baseline, our first experiment evaluates the performance of a classic transformer architecture (Vaswani et al. 2017) for the translation of spoken language to sign glosses sequences. We train a vanilla transformer model to predict sign gloss intermediary, with 2 layers, 8 heads and an embedding size of 256. We compare our performance against Stoll et al. (2020), who use an encoder-decoder network with 4 layers of 1000 Gated Recurrent Units (GRUs) as a translation architecture.

Table 2 shows that a transformer model achieves state-of-the-art results, significantly outperforming that of Stoll et al. (2020). This supports our use of the proposed transformer architecture for sign language understanding.

5.2 Gloss to Pose Production

In our next set of experiments, we evaluate our progressive transformer on the Gloss to Pose task outlined in Sect. 4.3. As a baseline, we train a progressive transformer model to translate from gloss to sign pose without augmentation.

5.2.1 Data Augmentation

Our base model suffers from prediction drift, with erroneous predictions accumulating over time. As transformer models

Table 3 Future prediction results on the Gloss to Pose task

		DEV SET					TEST SET				
F_f	F_t	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
0 (Base)	1 (Base)	7.38	9.62	13.81	25.03	26.55	7.13	9.30	13.63	24.86	26.03
0	2	9.52	12.13	16.91	27.98	30.68	9.34	11.99	16.78	28.03	30.29
0	5	**11.30**	**14.17**	19.19	30.45	**33.18**	**10.69**	**13.49**	**18.68**	30.69	31.78
0	10	10.99	13.83	19.02	30.57	32.34	9.93	12.50	17.49	28.94	30.86
0	20	10.08	12.84	17.79	29.30	31.27	9.23	12.02	17.27	29.53	30.11
2	5	10.93	13.85	**19.23**	**31.55**	32.80	10.23	13.13	18.60	**30.87**	**32.38**
5	10	10.32	13.07	18.44	30.95	31.81	9.37	12.12	17.53	30.39	30.52

Bold is used to signify the best performing model. Evaluation upon modifying the prediction frames, F_f to F_t

Table 4 Just counter results on the Gloss to Pose task

	DEV SET					TEST SET				
Configuration	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Base	7.38	9.62	13.81	25.03	26.55	7.13	9.30	13.63	24.86	26.03
Just counter	**12.34**	**15.04**	**21.17**	**32.43**	**35.59**	**12.16**	**15.50**	**21.45**	**33.53**	**34.80**

Bold is used to signify the best performing model. Evaluation against a base architecture that uses full skeleton pose as input

are trained to predict the next time-step, they are often not robust to noise in the target input. Therefore, we experiment with multiple data augmentation techniques introduced in Sect. 3.2; namely *Future Prediction*, *Just Counter* and *Gaussian Noise*.

Future Prediction Our first data augmentation method is conditional future prediction, requiring the model to predict more than just the next frame in the sequence. The model is trained to produce future frames between F_f and F_t. As can be seen in Table 3, prediction of multiple future frames causes an increase in model performance, from a base level of 7.38 BLEU-4 to 11.30 BLEU-4. We believe this is because the model cannot rely on just copying the previous frame to minimise the loss, but is instead required to predict the true motion with future pose predictions.

There exists a trade-off between benefit and complexity from increasing the number of predicted frames. We find the best performance comes from a prediction of 5 frames from the current time step. This is sufficient to encourage forward planning and motion understanding, but without a large averse effect on model complexity.

Just Counter Inspired by the memorisation capabilities of transformer models, we next evaluate a pure memorisation approach. Only the counter values are provided as target input to the model, as opposed to the usual full 3D skeleton joint positions. We show a further performance increase with this approach, considerably increasing the BLEU-4 score as shown in Table 4.

We believe the just counter model helps to allay the effect of drift, as the model must learn to decode the target sign pose solely from the counter position. It cannot rely on the ground truth joint embeddings it previously had access to. This halts the effect of erroneous sign pose prediction, as they are no longer fed back into the model. The setup at training and inference is now identical, requiring the model to only generalise to new data.

Gaussian Noise Our final augmentation evaluation examines the effect of applying noise to the skeleton pose sequences during training. For each joint, randomly sampled noise is applied to the input multiplied by a noise factor, r_n, representing the degree of noise augmentation.

Table 5 shows that Gaussian Noise augmentation achieves strong performance, with $r_n = 5$ giving the best results so far of 12.80 BLEU-4. A small amount of input noise causes the model to become more robust to auto-regressive prediction errors, as it must learn to correct the augmented inputs back to the target outputs. However, an increase of r_n above 5 causes a large degradation, affecting the model training and subsequent testing performance.

Overall, the proposed data augmentation techniques have been shown to significantly improve model performance and are fundamental to the production of understandable sign pose sequences. In the rest of our experiments, we use Gaussian Noise augmentation with $r_n = 5$.

5.2.2 Adversarial Training

We next evaluate our adversarial training regime outlined in Sect. 3.3. During training, a generator, G, and discriminator, D compete in a min-max game where G must create realistic

Table 5 Gaussian noise results on the Gloss to Pose task

	DEV SET					TEST SET				
r_n	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
0 (Base)	7.38	9.62	13.81	25.03	26.55	7.13	9.30	13.63	24.86	26.03
1	9.77	12.41	17.15	28.47	31.09	9.41	12.14	17.36	29.32	31.16
2	10.62	13.13	18.19	29.42	32.54	10.50	13.39	18.76	30.57	32.09
5	**12.80**	**16.03**	**21.60**	33.56	**35.86**	**11.85**	**15.16**	**21.56**	**34.56**	**35.31**
10	12.14	15.26	20.78	32.21	34.77	11.75	15.01	21.26	33.90	34.33
20	12.19	15.54	21.50	**33.69**	35.42	11.56	14.89	20.91	33.44	34.81

Bold is used to signify the best performing model. Evaluation upon modifying the noise rate, r_n

sign pose productions to fool D. During testing, we drop D and use the trained G to produce sign pose sequences given an input source text. For the adversarial experiments, we build our progressive transformer generator with 2 layers, 2 heads and an embedding size of 256. Best performance is achieved when the regression, λ_{Reg}, and adversarial, λ_{GAN}, losses are weighted as $\lambda_{Reg} = 100$ and $\lambda_{GAN} = 0.001$ respectively. This reflects the larger relative scale of the adversarial loss.

We first conduct an experiment with a non-conditional adversarial training regime. Only the sign pose sequence is critiqued, without conditioning upon source input. As shown on the top row of Table 6, this discriminator architecture produces a weak performing generator, of only 12.65 BLEU-4. This is less than the previous augmentation results, showing how an adversary applied solely to produced sign sequences negatively affects performance. The discriminator is prompting realistic production with no regards to source text, affecting the quality of the central translation task.

We next evaluate the conditional adversarial training regime, re-introducing a critique conditioned on source input. We evaluate different discriminator architectures by varying the number of CNN layers, N. This changes the strength of the adversary, which is required to be finely balanced against the generator in the min-max setup. Results are shown in Table 6, where an increase of N from 3 to 6 increases performance to a peak of 13.13 BLEU-4. This shows how a stronger discriminator can enforce a more realistic and expressive production from the generator. However, once N increases further and the discriminator becomes too strong, generator performance is negatively affected.

Overall, our conditional adversarial training regime has demonstrated improved performance over a model trained solely with a regression loss. Even for the test set, the result of 12.76 BLEU-4 is considerably higher than previous performance. This shows that the inclusion of a discriminator model increases the comprehension of sign production when conditioned on source sequence input. We believe this is due to the discriminator pushing the generator towards both a more expressive production and an accurate translation, in order to deceive the adversary. This, in turn, increases the sign content contained in the generated sequence, leading to a more understandable output and higher performance.

5.2.3 Mixture Density Networks

Our final Gloss to Pose evaluation is of the mixture density network (MDN) model configuration outlined in Sect. 3.4. During training, a multimodal distribution is created that best models the data, which is then used to sample from during inference. In this experiment, our progressive transformer model is built with 2 layers, 2 heads and an embedding size of 512.

We evaluate different numbers of mixture components, M, with results shown in Table 7. As shown, initially increasing M allows a multimodal prediction over a larger subspace, better modelling the sequence variation. This is supported by the results, with $M = 4$ achieving the highest validation performance of 13.14 BLEU-4. We find the regression to the mean of a deterministic prediction to be reduced, leading to a more expressive production. The subtleties of sign poses are restored, particularly for the small and variable finger joints. As M increases further, the added model complexity outweighs these benefits, leading to a performance degradation.

Our proposed MDN formulation achieves a higher performance than the previous deterministic approach of the progressive transformer. Comparison against the adversarial configuration shows a slight increase in performance (13.14 and 13.13 BLEU-4 respectively). However, given the back translation evaluation is not perfect, one might consider the performance of the MDN and adversarial models' to be similar, within the error margin of the SLT system. Both methods have a similar result of reducing the regression to the mean found in the original architecture and increasing sign pose articulation.

We additionally evaluate the combination of the MDN loss with the previously described adversarial loss, as explained in Sect. 3.4.4. This creates a network that uses a mixture distribution generator and a conditional discriminator. As in Sect. 5.2.2, we weight the MDN, $\lambda_{MDN} = 100$, and adversarial, $\lambda_{GAN} = 0.001$, losses respectively. As shown at the

Table 6 Adversarial training results on the Gloss to Pose task

		DEV SET					TEST SET				
N	Con.	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
6		12.65	16.09	22.04	**35.95**	36.29	12.05	15.34	21.25	33.37	34.90
3	✓	12.76	15.91	21.54	32.97	36.06	12.16	15.70	22.34	**35.43**	35.71
4	✓	12.70	15.96	21.76	33.69	36.40	12.06	15.46	21.56	33.49	35.55
5	✓	12.42	15.74	21.55	32.94	35.89	12.43	15.83	21.85	33.81	35.66
6	✓	**13.13**	**16.53**	**22.36**	34.13	**36.45**	12.60	16.05	**22.37**	34.67	**36.29**
7	✓	12.54	15.96	21.90	33.62	36.11	**12.76**	**16.15**	22.24	34.36	35.29
8	✓	12.41	15.89	22.02	34.99	35.95	12.38	15.80	22.09	34.60	35.85

Bold is used to signify the best performing model. Evaluation upon inclusion of conditioning on the source input (Con.) and the amount of discriminator layers, *N*

Table 7 Mixture density network results on the Gloss to Pose task

		DEV SET					TEST SET				
M	Adv.	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
1		12.22	15.47	21.15	32.91	35.39	10.88	14.04	19.87	32.75	32.95
2		12.89	16.16	21.80	33.23	36.16	11.60	14.71	20.40	32.18	34.31
4		**13.14**	**16.77**	**22.59**	33.84	**39.06**	11.94	15.22	21.19	33.66	35.19
5		12.75	15.91	21.40	32.67	36.04	11.57	14.77	20.66	32.69	34.48
10		11.48	14.52	19.92	31.62	33.67	10.90	14.02	19.77	32.15	33.39
20		12.59	16.02	22.17	**35.07**	36.28	12.15	15.35	21.34	33.62	**35.47**
30		12.61	15.93	21.72	33.72	36.28	12.11	15.54	21.69	33.30	35.26
50		11.15	14.18	19.66	30.95	33.58	10.56	13.67	19.60	32.62	33.30
4	✓	12.88	16.17	21.83	33.50	35.60	**12.32**	**15.62**	**21.82**	**34.35**	35.36

Bold is used to signify the best performing model. Evaluation upon the mixture components, *M* and the addition of adversarial loss (Adv)

bottom of Table 7, a combination of the MDN and adversarial training actually results in a lower performance than either individually on the dev set, of 12.88 BLEU-4. However, for the test set, this combination results in a slightly better performance than the MDN alone. Both of these configurations aim to alleviate the effect of regression to the mean, but may adversely affect the performance of the other due to their similar goals.

5.3 Text to Pose Production

We next evaluate our models on the Text to Pose task outlined in Sect. 4.3. This is the true end-to-end translation task, direct from a source spoken language sequence without the need for a gloss intermediary.

5.3.1 Model Configurations

We start by evaluating the various model configurations proposed in Sect. 3; namely base architecture, Gaussian noise augmentation, adversarial training and the MDN. The results of different configurations are shown in Table 8.

As with the Gloss to Pose task, Gaussian Noise augmentation increases performance from the base architecture, from 7.30 BLEU-4 to 10.75. We believe this is due to the reduction of the prediction drift as previously explained. The addition of adversarial training again increases performance, to 11.41 BLEU-4. The conditioning of the discriminator is even more important for this task, as the input is spoken language and provides more context for production.

The best Text to Pose performance of 11.54 BLEU-4 comes from the MDN model. As mentioned earlier, the performance of the adversarial and MDN setups' can be seen as equivalent considering the utilized SLT system is not perfect. Due to the increased context given by the source spoken language, there is a larger natural variety in sign production. Therefore, the multimodal modelling of the MDN is further enhanced, as highlighted by the performance gains. The addition of adversarial training on top of an MDN model does not increase performance further, as was seen in the previous evaluations.

Table 8 Results of the Text to Pose task for different model configurations

Configuration	DEV SET					TEST SET				
	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Base	7.30	9.21	12.87	23.15	26.11	6.79	8.74	12.57	23.46	25.02
Gaussian Noise	10.75	13.47	18.41	29.43	32.02	10.08	12.91	18.17	29.96	31.66
Adversarial	11.41	14.26	19.45	**31.02**	**33.59**	10.16	12.98	18.33	29.61	32.03
MDN	**11.54**	**14.48**	**19.63**	30.94	33.40	**11.68**	**14.55**	**19.70**	**31.56**	33.19
MDN + Adv.	11.49	14.36	19.38	30.04	33.92	11.18	14.08	19.35	30.66	**33.43**

Bold is used to signify the best performing model.

Table 9 Results of the *Text to Pose* and *Text to Gloss to Pose* network configurations for the Text to Pose task

Configuration	DEV SET					TEST SET				
	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Text to Pose	**11.54**	**14.48**	**19.63**	**30.94**	**33.40**	11.68	14.55	19.70	31.56	33.19
Text to Gloss to Pose	11.21	14.22	19.46	30.37	32.95	**13.64**	**17.05**	**23.09**	**34.94**	**36.90**

Bold is used to signify the best performing model.

5.3.2 Text to Pose v Text to Gloss to Pose

Our final experiment evaluates two end-to-end network configurations; sign production either direct from text [Text to Pose (T2P)] or via a gloss intermediary [Text to Gloss to Pose (T2G2P)]. These two tasks are outlined in Fig. 1, T2G2P on the left, T2P on the right.

As can be seen from Table 9, the T2P model outperforms the T2G2P for the development set. We believe this is because there is more information available within spoken language compared to a gloss representation, with more tokens per sequence to predict from. Predicting gloss sequences as an intermediary can act as an information bottleneck, as all the information required for production needs to be present in the gloss. Therefore, any contextual information present in the source text can be lost. However, in the test set, we achieve better performance using gloss intermediaries. We believe this is due to the effects of the limited number of training samples and the smaller vocabulary size of glosses on the generalisation capabilities of our networks.

The success of the T2P network shows that our progressive transformer model is powerful enough to complete two sub-tasks; firstly mapping spoken language sequences to a sign representation, then producing an accurate sign pose recreation. This is important for future scaling of the SLP model architecture, as many sign language domains do not have gloss availability.

Furthermore, our final BLEU-4 scores outperform similar end-to-end Sign to Text methods which do not utilise gloss information (Camgoz 2018) (9.94 BLEU-4). Note that this is an unfair direct comparison, but it does provide an indication of model performance and the quality of the produced sign pose sequences.

5.4 User Evaluation

The only true way to evaluate the sign production is in discussion with the Deaf communities, the end users. As our outputs are sign language sequences, we wish to understand how understandable they are to a native Deaf signer. We perform this evaluation with the skeletal output of the model, as we do not wish to confuse the translation ability of the system with the visual aesthetics of an avatar. However, by assessing the skeleton directly, we lose a lot of information that is conveyed in images such as shadow and occlusion. We therefore do a relative comparison between ground-truth and produced sequences, allowing us to assess the productions fairly. Although this work is in its infancy, we understand it is important to get early feedback from the Deaf communities. We believe the Deaf communities should be empowered and be involved in all steps of the development of any technology that is targeting their native languages.

We conducted a user evaluation with native DGS speakers to estimate the comprehension of our produced sign pose sequences. We designed a survey consisting of a comparison of the productions against ground truth data, the *Visual Task*, and a *Translation Task* that evaluates the sign comprehension. We animated our sign pose sequences as explained in Sect. 3.5 and placed the videos in an online survey. The user evaluation was conducted in collaboration with *HFC Human-Factors-Consult GmbH*.

We evaluated with two different model configurations; adversarial training and MDNs, providing users with different sequences from each and randomising the order of the videos. We received 20 Deaf participants who completed the evaluation, both comparing the production quality and testing the sign comprehension.

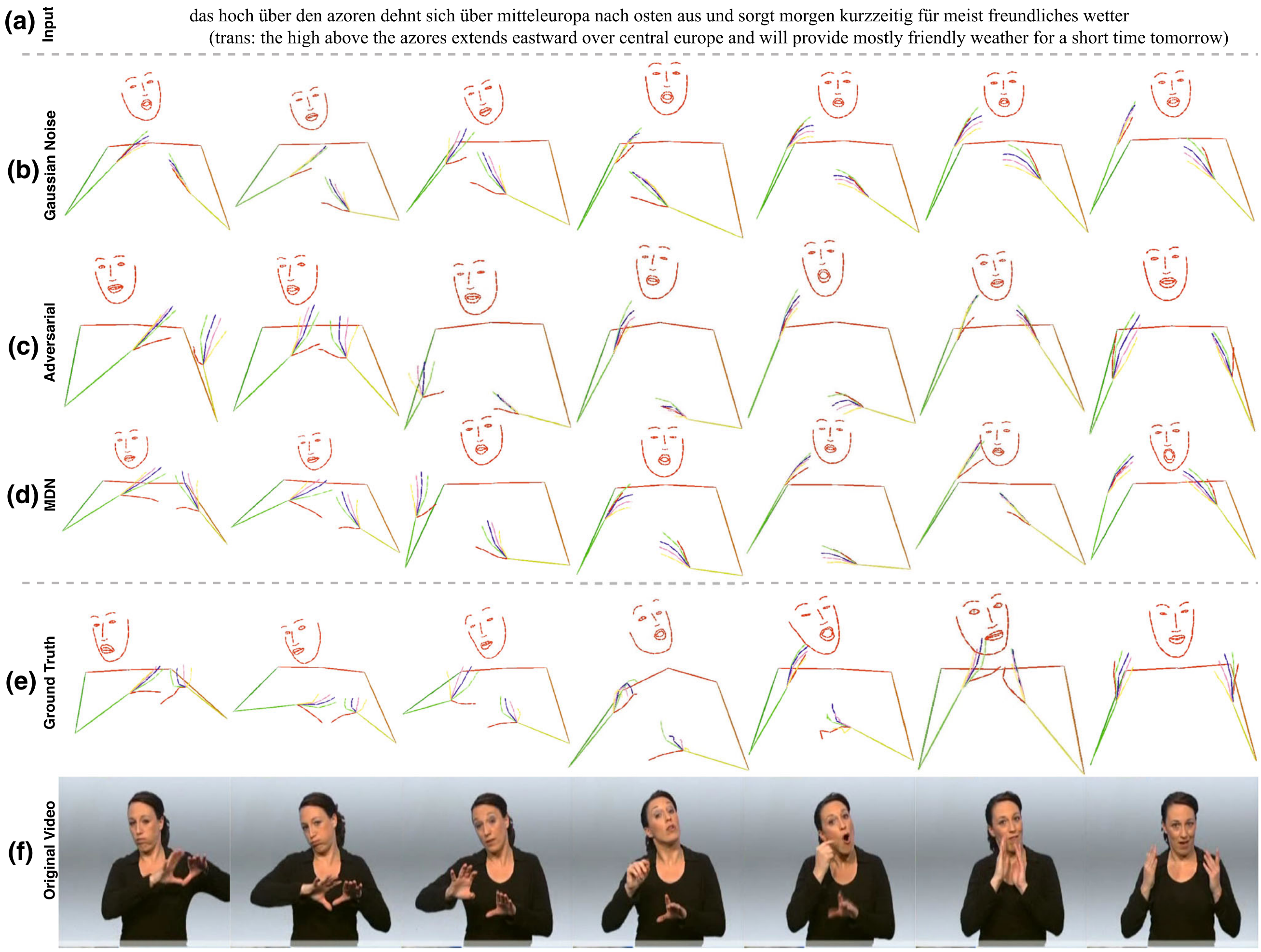
(a) Input
das hoch über den azoren dehnt sich über mitteleuropa nach osten aus und sorgt morgen kurzzeitig für meist freundliches wetter
(trans: the high above the azores extends eastward over central europe and will provide mostly friendly weather for a short time tomorrow)
(b) Gaussian Noise
(c) Adversarial
(d) MDN
(e) Ground Truth
(f) Original Video

Fig. 9 Qualitative evaluation of an example sign pose sequence. The source input is at the top, with the ground truth video frames and poses at the bottom. Middle rows contain produced sign pose sequences of different model configurations

Table 10 User evaluation results of the Visual task, showing the percentage of users who rated the ground truth (GT) or produced sequences (Prod) of a higher visual quality or equal

Configuration	GT (%)	Prod (%)	Equal (%)
Adversarial	14.58	8.33	77.08
MDN	0.00	15.38	84.62

5.4.1 Visual Task

Our first evaluation is a visual task, where a video of a sign production is shown alongside the corresponding ground truth sign sequence. The user is asked to rate both videos, with an implicit comparison between them. The comparison results are shown in Table 10, for both the adversarial and MDN model configurations.

Overall, the user feedback was mainly equal between the produced and ground-truth videos, with slightly more participants preferring the productions. This highlights the quality of the produced sign language videos, often as they are smoothly generated without any visual jitters. On the contrary, the original sequences often suffer from visual jitter, due to the motion blur in the original videos and the artifacts introduced in the 3D pose estimation.

The MDN configuration received higher ratings from the participants than the adversarial setup. 15.38% of users preferred the MDN productions over the ground-truth sequences, compared to 8.33% for the adversarial model. This demonstrates that the participants preferred the visuals of the MDN model. The quantitative back translation results for these models were similar (Sect. 5.2), but the users feedback suggests the MDNs production was of higher quality.

5.4.2 Translation Task

Our second evaluation is a translation task, designed to measure the translation accuracy of the sign productions. An automatic production was shown alongside 4 possible spo-

Table 11 User evaluation results for the Translation task, showing the percentage of participants who chose the correct spoken language translation out of a choice of 4

Configuration	Correct (%)
Adversarial	34.72
MDN	78.57

ken language translations of the sign sequence, where one is the correct sentence. The user is asked to select the most likely translation.

Table 11 shows that, for the adversarial examples, 34.72% of users chose the correct translation, compared to 78.57% for the MDN configuration. This is a drastic difference in the understanding of each of the model configurations, further demonstrating the success of the MDN productions. With the results of both visual and translation tasks, alongside the similar quantitative performance, we can conclude that the proposed MDN configuration generates the most realistic and expressive sign pose production.

6 Qualitative Evaluation

In this section, we report qualitative results for our SLP model. We share snapshot examples of sign pose sequences in Figs. 9 and 11, visually comparing the outputs of the proposed model configurations for the gloss to pose task. The corresponding unseen spoken language sequence is shown as input at the top, alongside example frames from the ground truth video and the produced sign language sequence.

As can be seen from the provided examples, our SLP model produces visually pleasing and realistic looking sign with a close correspondence to the ground truth video. Body motion is smooth and accurate, whilst hand shapes are meaningful if a little under-expressed. Specific to non-manual features, we find a close correspondence to the ground truth video alongside accurate head movement, with a slight under-articulation of mouthings.

For comparisons between model configurations, the Gaussian Noise productions can be seen to be under-expressed, specifically the hand shape and motions of Fig.9b. The adversarial training improves this, resulting in a significantly more expressive production, with larger hand shapes seen in the 6th frame of Fig. 11c. This is due to the discriminator pushing the productions towards a more realistic output. Inclusion of a MDN representation can be seen to provide more accuracy in production, with the sign poses of Fig. 9d visually closer to the ground truth. This is due to the mixture distribution modelling the uncertainty of the continuous sign sequences, removing the mean productions that can be seen in the Gaussian Noise productions.

Visual comparisons between the adversarial and MDN productions reflect the equal quantitative performance of the two (Sect. 5.2), demonstrating two contrasting ways of increasing the sign comprehension. Overall, the problem of regression to the mean is diminished and a more realistic production is achieved, highlighting the importance of the proposed model configurations.

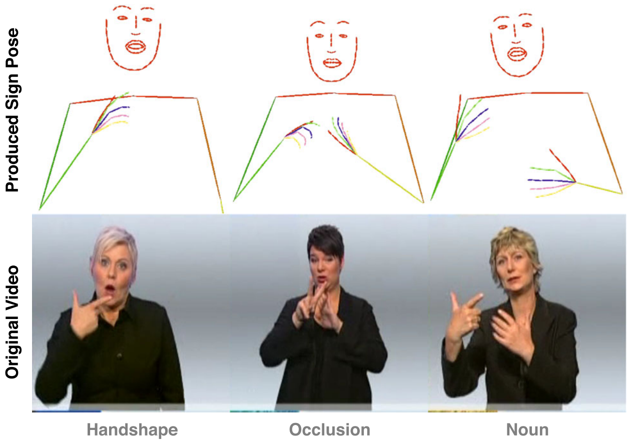

Fig. 10 Example failure sign pose productions. Due to either complex handshape (left), hand occlusion (middle) or proper noun (right)

These examples show that regressing continuous 3D human pose sequences can be successfully achieved using a self-attention based approach. The predicted joint locations for neighbouring frames are closely positioned, showing that the model has learnt the subtle signer movements. Smooth transitions between signs are produced, highlighting a difference from the discrete generation of spoken language.

Figure 10 shows some failure cases of the approach. Complex hand classifiers can be difficult to replicate (left) and hand occlusion affects the quality of training data (middle). We find that the most difficult production occurs with proper nouns and specific entities, due to the lack of grammatical context and examples in the training data (right).

7 Conclusions

In this work, we presented a Continuous 3D Multi-Channel Sign Language Production model, the first SLP model to translate from text to continuous 3D sign pose sequences in an end-to-end manner. To enable this, we proposed a *Progressive Transformer* architecture with an alternative formulation of transformer decoding for variable length continuous sequences. We introduced a counter decoding technique to predict continuous sequences of variable lengths by tracking the production progress over time and predicting the end of sequence.

To reduce the prediction drift that is often seen in continuous sequence production, we presented several data augmentation methods that significantly improve model

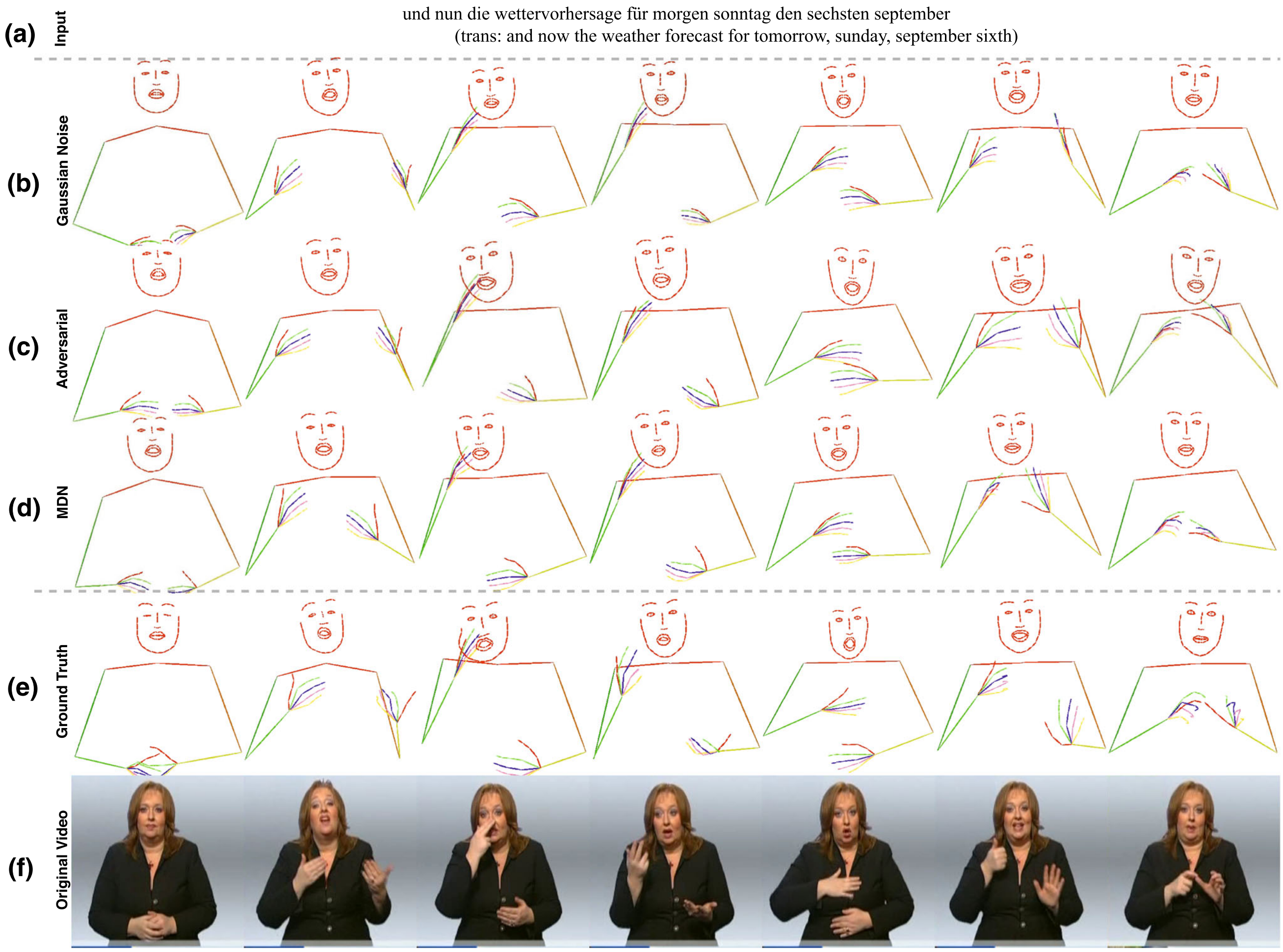
(a)
Input
und nun die wettervorhersage für morgen sonntag den sechsten september
(trans: and now the weather forecast for tomorrow, sunday, september sixth)
(b)
Gaussian Noise
(c)
Adversarial
(d)
MDN
(e)
Ground Truth
(f)
Original Video

Fig. 11 Qualitative evaluation of an example sign pose sequence. The source input is at the top, with the ground truth video frames and poses at the bottom. Middle rows contain produced sign pose sequences of different model configurations

performance. Predicting continuous values often results in under-articulated output, and thus we proposed the addition of adversarial training to the network, introducing a conditional discriminator model to prompt a more realistic and expressive production. We also proposed a mixture density network (MDN) modelling, utilising the progressive transformer outputs to paramatise a mixture Gaussian distribution.

We evaluated our approach on the challenging PHOENIX14T dataset, proposing a back translation evaluation metric for SLP. Our experiments showed the importance of data augmentation techniques to reduce model drift. We improved our model performance with the addition of both an adversarial training regime and a MDN output representation. Furthermore, we have shown that a direct text to pose translation configuration can outperform a gloss intermediary model, meaning SLP models are not limited to domains where expensive gloss annotation is available.

Finally, we conducted a user study of the Deaf's response to our sign productions, understanding the sign comprehension of the proposed model configurations. The results show that our productions, while not perfect, can be further improved by reducing and smoothing noise inherent to the data and approaches. However, they also highlight that the current sign productions still need improvement to be fully understandable by the Deaf. The field of SLP is in its infancy, with a potential for large growth and improvement in the future.

We believe the current 3D skeleton representation affects the comprehension of sign pose sequences. As future work, we would like to increase the realism of sign production by generating photo-realistic signers, using GAN image-to-image translation models (Chan et al. 2019; Zhu et al. 2017; Isola et al. 2017) to expand from the current skeleton representation. Drawing on feedback from the user evaluation, we plan to improve the hand articulation via a hand shape classifier to increase comprehension. An automatic viseme generator could also be included to the pipeline to improve mouthing patterns, producing features in a deterministic manner direct from dictionary data.

Acknowledgements This work received funding from the SNSF Sinergia project 'SMILE' (CRSII2 160811), the European Union's Horizon2020 research and innovation programme under grant agreement no. 762021 'Content4All' and the EPSRC project 'ExTOL' (EP/R03298X/1). This work reflects only the authors view and the Commission is not responsible for any use that may be made of the information it contains.

References

Adaloglou, N., Chatzis, T., Papastratis, I., Stergioulas, A., Papadopoulos, G. T., Zacharopoulou, V., Xydopoulos, G. J., Atzakas, K., Papazachariou, D., & Daras, P. (2019). A comprehensive study on sign language recognition methods. In *IEEE transactions on multimedia*.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer normalization. ArXiv preprint arXiv:1607.06450.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *Proceedings of the international conference on learning representations (ICLR)*.

Bauer, B., Hienz, H., & Kraiss, K.-F. (2000). Video-based continuous sign language recognition using statistical methods. In *Proceedings of 15th international conference on pattern recognition (ICPR)*.

Berndt, D. J., & Clifford, J. (1994). Using dynamic time warping to find patterns in time series. In *AAA1 workshop on knowledge discovery in databases (KDD)*.

Bishop, C. M. (1994). *Mixture density networks*. Technical Report, Citeseer.

Bragg, D., Koller, O., Bellard, M., Berke, L., Boudreault, P., Braffort, A., Caselli, N., Huenerfauth, M., Kacorri, H., Verhoef, T., & Vogler, C. (2019). Sign language recognition, generation, and translation: An interdisciplinary perspective. In *The 21st international ACM SIGACCESS conference on computers and accessibility*.

British Deaf Association (BDA). (2020). UK deaf community. https://bda.org.uk/fast-facts-about-the-deafcommunity/.

Cai, H., Bai, C., Tai, Y. W., & Tang, C. K. (2018). Deep video generation, prediction and completion of human action sequences. In *Proceedings of the European conference on computer vision (ECCV)*.

Camgoz, N. C., Hadfield, S., Koller, O., & Bowden, R. (2017). SubUNets: End-to-end hand shape and continuous sign language recognition. In *Proceedings of the IEEE international conference on computer vision (ICCV)*.

Camgoz, N. C., Hadfield, S., Koller, O., Ney, H., & Bowden, R. (2018). Neural sign language translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Camgoz, N. C., Koller, O., Hadfield, S., & Bowden, R. (2020a). Multi-channel transformers for multi-articulatory sign language translation. In *Assistive computer vision and robotics workshop (ACVR)*.

Camgoz, N. C., Koller, O., Hadfield, S., & Bowden, R. (2020b). Sign language transformers: joint end-toend sign language recognition and translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., & Sheikh, Y. (2017). OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Chai, X., Li, G., Lin, Y., Xu, Z., Tang, Y., Chen, X., & Zhou, M. (2013). Sign language recognition and translation with kinect. In *IEEE international conference on automatic face and gesture recognition (AFGR)*.

Chan, C., Ginosar, S., Zhou, T., & Efros, A. A. (2019). Everybody dance now. In *Proceedings of the IEEE international conference on computer vision (CVPR)*.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Conference on empirical methods in natural language processing (EMNLP)*.

Cooper, H. M., Ong, E. J., Pugeault, N., & Bowden, R. (2012). Sign language recognition using sub-units. *Journal of Machine Learning Research (JMLR)*.

Cox, S., Lincoln, M., Tryggvason, J., Nakisa, M., Wells, M., Tutt, M., & Abbott, S. (2002). TESSA: A system to aid communication with deaf people. In *Proceedings of the ACM international conference on assistive technologies*.

Cui, R., Liu, H., & Zhang, C. (2017). Recurrent Convolutional neural networks for continuous sign language recognition by staged optimization. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J., Le, Q. V., & Salakhutdinov, R. (2019). Transformer-XL: Attentive language models beyond a fixed-length context. In: *International conference on learning representations (ICLR)*.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the annual meeting of the association for computational linguistics (ACL)*.

Ebling, S., & Huenerfauth, M. (2015). Bridging the Gap between Sign Language Machine Translation and Sign Language Animation using Sequence Classification. In *Proceedings of SLPAT 2015: 6th workshop on speech and language processing for assistive technologies*.

Elliott, R., Glauert, J. R., Kennaway, J. R., Marshall, I., & Safar, E. (2008). Linguistic modelling and language-processing technologies for avatar-based sign language presentation. In *Universal access in the information society*.

Forster, J., Schmidt, C., Koller, O., Bellgardt, M., & Ney, H. (2014). Extensions of the sign language recognition and translation corpus RWTH-PHOENIX-Weather. In *Proceedings of the international conference on language resources and evaluation (LREC)*.

Ginosar, S., Bar, A., Kohavi, G., Chan, C., Owens, A., & Malik, J. (2019). Learning individual styles of conversational gesture. In *Proceedings ofthe IEEE conference on computer vision and pattern recognition (CVPR)*.

Girdhar, R., Carreira, J., Doersch, C., & Zisserman, A. (2019). Video action transformer network. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Glauert, J. R. W., Elliott, R., Cox, S. J., Tryggvason, J., & Sheard, M. (2006). VANESSA: A system for communication between deaf and hearing people. In *Technology and disability*.

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the* *international conference on artificial intelligence and statistics (AISTATS)*.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. et al. (2014). Generative adversarial nets. In *Proceedings of the advances in neural information processing systems (NIPS)*.

Graves, A. (2013). Generating sequences with recurrent neural networks. ArXiv preprint arXiv:1308.0850.

Grobel, K., & Assan, M. (1997). Isolated sign language recognition using hidden Markov models. In *IEEE international conference on systems, man, and cybernetics*.

Ha, D., & Eck, D. (2018). A neural representation of sketch drawings. In *International conference on learning representations (ICLR)*.

Ha, D., & Schmidhuber, J. (2018). Recurrent world models facilitate policy evolution. In *Advances in neural information processing systems (NIPS)*.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Holt, J. A. (1993). Stanford achievement test—8th edition: Reading comprehension subgroup results. In *American annals of the deaf*.

Hu, Y., Zhan, W., & Tomizuka, M. (2018). Probabilistic prediction of vehicle semantic intention and motion. In *IEEE intelligent vehicles symposium (IV)*.

Huang, C. Z. A., Vaswani, A., Uszkoreit, J., Shazeer, N., Simon, I., Hawthorne, C., Dai, A. M., Hoffman, M. D., Dinculescu, M., & Eck, D. (2018). Music transformer. In *International conference on learning representations (ICLR)*.

Huang, J., Zhou, W., Zhang, Q., Li, H., & Li, W. (2018). Video-based sign language recognition without temporal segmentation. In *AAAI conference on artificial intelligence (AAAI)*.

Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Kayahan, D., & Gungor, T. (2019). A hybrid translation system from Turkish spoken language to Turkish sign language. In *IEEE international symposium on innovations in intelligent systems and applications (INISTA)*.

Kingma, D. P., & Ba, J. (2014). ADAM: A method for stochastic optimization. In *Proceedings of the international conference on learning representations (ICLR)*.

Kipp, M., Heloir, A., & Nguyen, Q. (2011a). Sign language avatars: Animation and comprehensibility. In *International workshop on intelligent virtual agents (IVA)*.

Kipp, M., Nguyen, Q., Heloir, A., & Matthes, S. (2011b). Assessing the DeafUserPer-spective on sign language avatars. In *The proceedings of the 13th international ACM SIGACCESS conference on computers and accessibility (ASSETS)*.

Ko, S. K., Kim, C. J., Jung, H., & Cho, C. (2019). Neural sign language translation based on human keypoint estimation. In *Applied sciences*.

Koller, O., Camgoz, N. C., Ney, H., & Bowden, R. (2019). Weakly supervised learning with multi-stream CNN-LSTM-HMMs to discover sequential parallelism in sign language videos. In *IEEE transactions on pattern analysis and machine intelligence (PAMI)*.

Koller, O., Forster, J., & Ney, H. (2015). Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. In *Computer vision and image understanding (CVIU)*.

Koller, O., Zargaran, S., & Ney, H. (2017). Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent CNN-HMMs. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Koller, O., Zargaran, O., Ney, H., & Bowden, R. (2016). Deep sign: Hybrid CNN-HMM for continuous sign language recognition. In *Proceedings of the British machine vision conference (BMVC)*.

Kouremenos, D., Ntalianis, K. S., Siolas, G., & Stafylopatis, A. (2018). Statistical machine translation for Greek to Greek sign language using parallel corpora produced via rule-based machine translation. In *IEEE 31st international conference on tools with artificial intelligence (ICTAI)*.

Kreutzer, J., Bastings, J., & Riezler S. (2019). Joey NMT: A minimalist NMT toolkit for novices. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP): System demonstrations.*

Lee, H.-Y., Yang, X., Liu, M. Y., Wang, T. C., Lu, Y. D., Yang, M. H., & Kautz, J. (2019). Dancing to music. In *Advances in neural information processing systems (NIPS).*

Li, C., & Lee, G. H. (2019). Generating multiple hypotheses for 3D human pose estimation with mixture density network. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR).*

Kevin, L., Li, D., He, X., Zhang, Z., & Sun, M. T. (2017). Adversarial ranking for language generation. In *Advances in neural information processing systems (NIPS).*

Pengfei, L., & Huenerfauth, M. (2010). Collecting a motion-capture corpus of American sign language for data-driven generation research. In *Proceedings of the NAACL HLT 2010 workshop on speech and language processing for assistive technologies.*

Maas, A. L., Hannun, A. Y., & Ng, A. Y. (2013). Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the international conference on machine learning (ICML).*

Makansi, O., Ilg, E., Cicek, O., & Brox, T. (2019). Overcoming limitations of mixture density networks: A sampling and fitting framework for multimodal future prediction. In *Proceedings ofthe IEEE conference on computer vision and pattern recognition (CVPR).*

McDonald, J. et al. (2016). Automated technique for real-time production of lifelike animations of American sign language. In *Universal access in the information society (UAIS).*

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems (NIPS).*

Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. ArXiv preprint arXiv:1411.1784.

Mukherjee, S., Ghosh, S., Ghosh, S., Kumar, P., & Roy, P. P. (2019). Predicting video-frames using encoder-convlstm combination. In *IEEE international conference on acoustics, speech and signal processing (ICASSP).*

Orbay, A., & Akarun, L. (2020). Neural sign language translation by learning tokenization. In *IEEE international conference on automatic face and gesture recognition (FG).*

Ozdemir, O., Necati, C. C., & Lale, A. (2016). Isolated sign language recognition using improved dense trajectories. In *Proceedings of the signal processing and communication application conference (SIU).*

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z. (2017). Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*.

Pfau, R., & Quer, J. (2010). *Nonmanuals: their grammatical and prosodic roles*.

Povey, D., Hadian, H., Ghahremani, P., Li, K., & Khudanpur, S. (2018). A time-restricted self-attention layer for ASR. In *IEEE international conference on acoustics, speech and signal processing (ICASSP)*.

Press, O., Bar, A., Bogin, B., Berant, J., & Wolf, L. (2017). Language generation with recurrent generative adversarial networks without pre-training. ArXiv preprint arXiv:1706.01399.

Prokudin, S., Gehler, P., & Nowozin, S. (2018). Deep directional statistics: Pose estimation with uncertainty quantification. In *Proceedings of the European conference on computer vision (ECCV)*.

Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. ArXiv preprint arXiv:1511.06434.

Ren, X., Li, H., Huang, Z., & Chen, Q. (2019). Music-oriented dance video synthesis with pose perceptual loss. ArXiv preprint arXiv:1912.06606.

Ren, Y., Ruan, Y., Tan, X., Qin, T., Zhao, S., Zhao, Z., & Liu, T. Y. (2019). Fastspeech: Fast, robust and controllable text to speech. In *Advances in neural information processing systems (NIPS)*.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., & Chen, X. (2016). Improved techniques for training GANs. In *Advances in neural information processing systems (NIPS)*.

Saunders, B., Camgoz, N. C., & Bowden, R. (2020a). Adversarial training for multi-channel sign language production. In *Proceedings of the British machine vision conference (BMVC)*.

Saunders, B., Camgoz, N. C., & Bowden, R., (2020b). Progressive transformers for end-to-end sign language production. In *Proceedings of the European conference on computer vision (ECCV)*.

Schuster, M. (2000). Better generative models for sequential data problems: bidirectional recurrent mixture density networks. In *Advances in neural information processing systems (NIPS)*.

Starner, T., & Pentland, A., (1997). Real-time American sign language recognition from video using hidden Markov models. In *Motion-based recognition.*

Stokoe, W. C. (1980). Sign language structure. In: *Annual review of anthropology*.

Stoll, S., Camgoz, N. C., Hadfield, S., & Bowden, R. (2020). Text2Sign: Towards sign language production using neural machine translation and generative adversarial networks. In *International journal of computer vision (IJCV)*.

Sutskever, I., Vinyals, O., & Le, Q. V., (2014). Sequence to sequence learning with neural networks. In *Proceedings of the advances in neural information processing systems (NIPS)*.

Sutton-Spence, R., & Woll, B. (1999). *The linguistics of British sign language: An introduction*. Cambridge University Press.

Tamura, S. & Kawasaki, S., (1988). Recognition of sign language motion images. In *Pattern recognition.*

Tulyakov, S., Liu, M. Y., Yang, X., & Kautz, J. (2018). MoCoGAN: Decomposing motion and content for video generation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Valli, C. & Lucas, C., (2000). *Linguistics of American sign language: An introduction*. Gallaudet University Press.

Varamesh, A., & Tuytelaars, T., (2020). Mixture dense regression for object detection and human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems (NIPS)*.

Vogler, C., & Metaxas, D., (1999). Parallel midden Markov models for American sign language recognition. In *Proceedings of the IEEE international conference on computer vision (ICCV).*

Vondrick, C., Pirsiavash, H., & Torralba, A. (2016). Generating videos with scene dynamics. In *Advances in neural information processing systems (NIPS)*.

Wang, X., Takaki, S., & Yamagishi, J., (2017). An autoregressive recurrent mixture density network for parametric speech synthesis. In *IEEE international conference on acoustics, speech and signal processing (ICASSP)*.

World Health Organisation (WHO) (2020). Deafness and hearing loss. https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss.

Wu, L., Xia, Y., Tian, F., Zhao, L., Qin, T., Lai, J., & Liu, T. Y. (2017). Adversarial neural machine translation. In *Proceedings of the Asian conference on machine learning (ACML)*.

Xiao, Q., Qin, M., & Yin, Y., (2020). Skeleton-based Chinese sign language recognition and generation for bidirectional communication between deaf and hearing people. In *Neural networks*.

Yang, Z., Chen, W., Wang, F., & Xu, B. (2017). Improving neural machine translation with conditional sequence generative adversarial nets. In *Proceedings of the conference of the North American chapter of the association for computational linguistics (ACL)*.

Ye, Q., & Kim, T-K. (2018). Occlusion-aware hand pose estimation using hierarchical mixture density network. In *Proceedings of the European conference on computer vision (ECCV)*.

Yin, K. (2020). Attention is all you sign: Sign language translation with transformers. In *ECCV sign language recognition, translation and production workshop*.

Zelinka, J., & Kanis, J. (2020). Neural sign language synthesis: Words are our glosses. In *The IEEE winter conference on applications of computer vision (WACV)*.

Zhang, X.-Y., Yin, F., Zhang, Y. M., Liu, C. L., & Bengio, Y. (2017). Drawing and recognizing Chinese characters with recurrent neural network. In *IEEE transactions on pattern analysis and machine intelligence (PAMI)*.

Zhang, Y., Gan, Z., & Carin, L. (2016). Generating text via adversarial training. In *Neural information processing systems (NIPS) workshop on adversarial training*.

Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., & Liu, Q. (2019). ERNIE: Enhanced language representation with informative entities. In *57th annual meeting of the association for computational linguistics (ACL)*.

Zhou, L., Zhou, Y., Corso, J. J., Socher, R., & Xiong, C. (2018). End-to-end dense video captioning with masked transformer. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Zhu, J.-Y., Park, T., Isola, P., & Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*.

Zwitserlood, I., Verlinden, M., Ros, J., & Van Der Schoot, S. (2004). Synthetic signing for the deaf: Esign. In *Proceedings of the conference and workshop on assistive technologies for vision and hearing impairment (CVHI)*.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.