Mixed SIGNals: Sign Language Production via a Mixture of Motion Primitives

Ben Saunders, Necati Cihan Camgoz, Richard Bowden
University of Surrey
{b.saunders, n.camgoz, r.bowden}@surrey.ac.uk

Abstract

It is common practice to represent spoken languages at their phonetic level. However, for sign languages, this implies breaking motion into its constituent motion primitives. Avatar based Sign Language Production (SLP) has traditionally done just this, building up animation from sequences of hand motions, shapes and facial expressions. However, more recent deep learning based solutions to SLP have tackled the problem using a single network that estimates the full skeletal structure.

*We propose splitting the SLP task into two distinct jointly-trained sub-tasks. The first translation sub-task translates from spoken language to a latent sign language representation, with gloss supervision. Subsequently, the animation sub-task aims to produce expressive sign language sequences that closely resemble the learnt spatio-temporal representation. Using a progressive transformer for the translation sub-task, we propose a novel Mixture of Motion Primitives (*MoMP*) architecture for sign language animation. A set of distinct motion primitives are learnt during training, that can be temporally combined at inference to animate continuous sign language sequences.*

*We evaluate on the challenging RWTH-PHOENIX-Weather-2014**T** (PHOENIX14**T**) dataset, presenting extensive ablation studies and showing that* MoMP *outperforms baselines in user evaluations. We achieve state-of-the-art back translation performance with an 11% improvement over competing results. Importantly, and for the first time, we showcase stronger performance for a full translation pipeline going from spoken language to sign, than from gloss to sign.*

1. Introduction

Sign languages are visual languages used for communication by the Deaf communities. Analogous to phonemes in speech, sign languages can be broken down into cheremes, the smallest distinctive structural units [58]. Cheremes can be represented as motion primitives, a set of manual and non-manual motions[1] that are combined to represent all sign language utterances. Such phonetic representations are typically used by linguists for annotation [29, 62] or in graphical based avatars for sign generation.

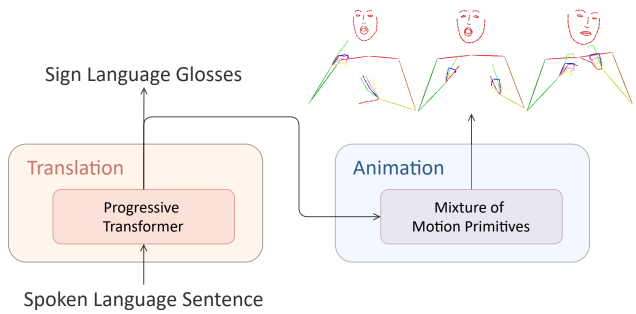

Figure 1: An overview of the proposed SLP sub-tasks of Translation and Animation.

Sign Language Production (SLP), the translation from spoken language sentences to continuous sign language sequences, requires both an accurate translation and expressive animation. Previous work has traditionally tackled these two sub-task as a single task with one network [52, 59, 68], leading to under-expressive production. Although previous SLP models have used gloss[2] as an intermediate representation [60], this creates an information bottleneck that disregards the contextual information available in the original text.

In this paper, we propose to formulate SLP as two distinct but jointly-trained sub-tasks as can be seen in Figure 1: 1) An initial translation from spoken language to a sign language representation, with gloss supervision; 2) Subsequent animation of a visual sign language sequence. This can be viewed as analogous to a text-to-speech pipeline with an initial translation into phonemes and a subsequent vocalisation. However, we do not force a gloss information bottleneck but instead condition learning on gloss, resulting in

[1]Manual features are the hand shape and motion, whereas non-manuals are the mouthings and facial expressions

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

significant performance increases.

We utilise a progressive transformer model as our translation backbone [52]. Sign language representations are learnt per frame using the gloss supervision. This prompts the sub-network to learn meaningful representations for the end goal of sign language production.

To animate expressive sign language sequences, we propose a novel Mixture of Motion Primitives (MoMP) network. Based on a Mixture-of-Expert (MoE) architecture, we learn a combination of distinct motion primitives that are able to produce an infinite number of unique sign language utterances. Due to the continuous nature of sign language, we apply expert blending on a per frame basis, thus enabling different experts to be activated for separate sections of the output sequence.

As the subset of motion primitives is significantly smaller than the full set of signs, the animation sub-task is reduced to a gating network that selects the correct primitive to animate for specific sections of the full sequence. This also enables a scaling of SLP models to larger datasets, with new signs being a novel combination of the learnt primitives. We represent experts as masked transformer encoders, using self-attention to learn unique structural motions. We use a further transformer encoder model for the gating network, thus building, to the best of our knowledge, the first full transformer-based MoE architecture.

We evaluate on the challenging PHOENIX14**T** corpus, performing an extensive ablation study of the proposed network and conducting a user evaluation that highlights the animation quality of the MoMP. Furthermore, we achieve state-of-the-art SLP back translation results and showcase, for the first time, stronger performance for a full translation pipeline that produces sign sequences directly from the source spoken language, than from an intermediate gloss representation.

The contributions of this paper can be summarised as:

- A novel transformer-based MoE architecture, Mixture of Motion Primitives (MoMP), that combines learnt motion primitives at the frame level.
- The first SLP model to separately model the sub-tasks of translation and animation.
- State-of-the-art SLP performance and user evaluation results on the PHOENIX14**T** dataset.
- The first SLP model to achieve higher performance for a full translation pipeline going from spoken language to sign, than from gloss to sign.

The rest of this paper is organised as follows: In Section 2, we review the literature in SLP and MoE. In Section 3, we outline the proposed MoMP network. We present quantitative and qualitative model comparison in Section 4, and finally conclude in Section 5.

2. Related Work

Sign Language Motion Primitives Phonemes are defined as the smallest distinctive structural units of spoken language that can be combined to create an infinite number of meaningful utterances [39, 64]. Cheremes are used as the equivalent representation specific to sign language [5, 14, 58]. This phonetic structure of sign language includes the sublexical parameters of shape, movement and location used to describe the motion and structure of all signs [6, 22]. Motion primitives can be seen as a subset of cheremes, encompassing the gestural motions of both manual and non-manual features. Although the possible motion primitives are much smaller in number than the full set of signs, they can be combined to recreate all unique sign language sequences.

Sign Language Production Computational sign language research has been prominent for the last 30 years [3, 57, 63]. Previous research has focused on isolated sign recognition [1, 28, 45], Continuous Sign Language Recognition (CSLR) [7, 17, 37, 36] and, more recently, the task of Sign Language Translation (SLT) [8, 9, 35, 44]. Camgoz *et al*. [10] proposed a jointly trained CSLR and SLT system, showing a performance increase for both tasks.

Sign Language Production (SLP), the translation from spoken language sentence to sign language sequences, has traditionally been tackled using avatars [2, 15, 18, 20, 24]. Animating sign using avatars helps to separate the translation task from the animation, with an initial manual translation from text into a sign language representation such as HamNoSys [29] or SignWriting [62].

In contrast, more recent works have applied deep learning to SLP [16, 43, 50, 51, 53, 59, 66, 68], with Saunders *et al*. [52] proposing the first SLP model to translate from spoken language sentences to sign language sequences in an end-to-end manner. However, these methods combine both the translation and animation elements into a single pipeline, leading to a lack of expressive animation. Stoll *et al*. [60] use the intermediate representation of gloss but this creates an information bottleneck that all sign must pass through. In this work, we separate the animation from the translation sub-task using a joint supervision of both gloss and skeletal pose. Furthermore, we combine a learnt set of motion primitives that can animate any sign language utterance and use gloss to condition learning rather than form a bottleneck.

Mixture of Experts Mixture-of-Experts (MoEs) are a jointly-trained ensemble of expert systems, each locally specialised for a different subdomain of inputs [31, 32]. A gating network predicts a set of blending coefficients, used to weight the decision of each expert in the final output. We

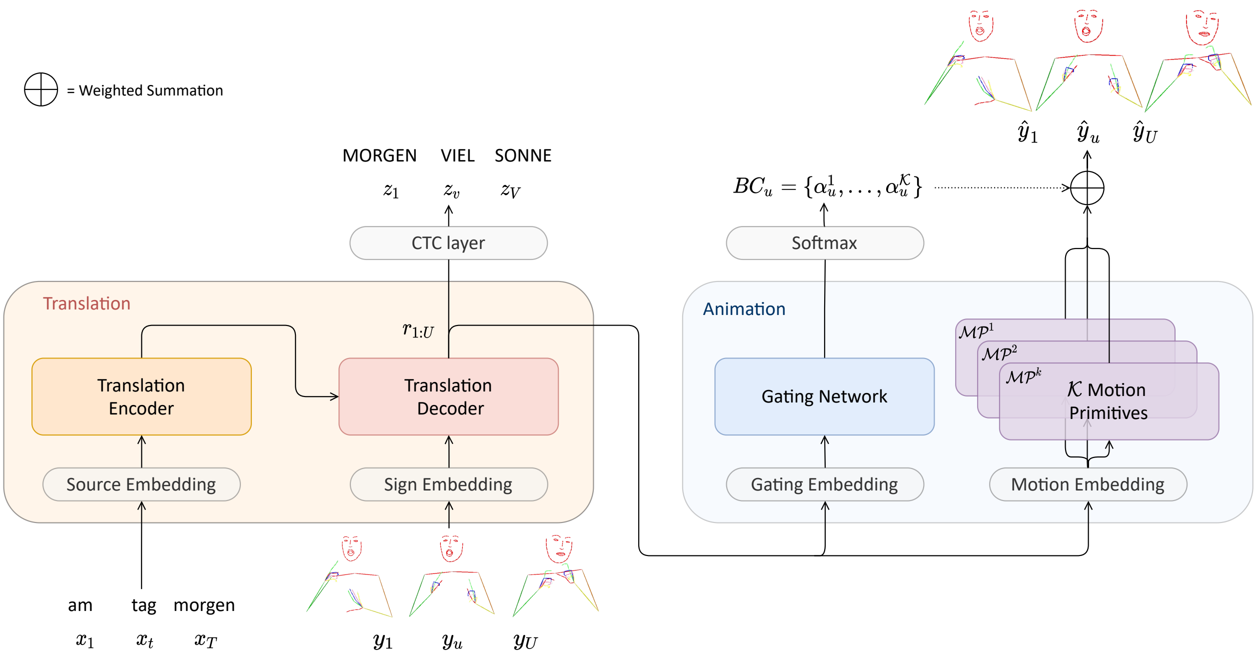

Figure 2: Mixture of Motion Primitives (MoMP) network overview, showing an initial translation sub-task from spoken language, $x_{1:T}$, to sign language representation with gloss supervision, $z_{1:V}$ (left). A subsequent animation sub-task uses a blended mixture of $\mathcal{K}$ learnt motion primitives, $\mathcal{MP}^{\mathcal{K}}$, to produce a continuous sign language sequence, $\hat{y}_{1:U}$, (right).

refer to Gormley *et al*. for a broad survey on MoEs [26].

Recently, MoEs have become popular in Neural Machine Translation (NMT) [23, 47, 48, 67]. Sparsely gated MoEs use a MoE layer to enable extensive scaling of parameters, requiring only a subset of experts to be computed for each sequence [21, 40, 49, 55]. MoEs have also been used to promote diversity in text generation [13, 30, 56] and even to enable multi-task learning [33, 41].

Stoll *et al*. [61] applied a MoEs concept to SLP, producing isolated signs from a small vocabulary of 105 glosses. Our work produces continuous sign sequences directly from spoken language, using distinct motion primitives combined at the output level for 1066 glosses.

Combining transformers with MoEs is motivated by research that suggests transformer networks are overparameterised [42, 65]. Peng *et al*. [47] build experts consisting of multiple transformer heads and Lepikhin *et al*. [40] build large-scale NMT models using transformer MoEs. Our work differs in that we represent each expert as an individual transformer encoder and propose a transformer-based gating network. Additionally, we apply MoEs at the token i.e. frame level as opposed to the sequence level, enabling the modelling of specialised motion primitives.

Probably closest to our work is the approach of Zhang *et al*. [69], who use MoEs to model repetitive gait sequences of quadraped motion. Each expert is trained to be specialised in the production of a certain type of motion. However, we tackle the subtle motions of sign language in the context of translation and, to achieve the required subtleties, perform blending at the output level rather than in the feature space.

3. Methodology

Given a source spoken language sequence, $\mathcal{X} = (x_1, ..., x_T)$ with T words, the objective of an SLP model is to produce a sign language sequence, $\mathcal{Y} = (y_1, ..., y_U)$ with U frames. State-of-the-art SLP works have approached the task using a single end-to-end network with no intermediate representation [52, 68]. This simultaneously tackles the challenging tasks of accurate translation into sign language grammar and expressive animation of sign language motion with a single unified loss function, impacting the networks ability to perform well at either task.

Motivated by this, we propose to split the SLP task into two jointly-trained sub-tasks: 1) An initial translation task from spoken language to sign language representation, with gloss supervision, $\mathcal{Z} = (z_1, ..., z_V)$ with V glosses; 2) A subsequent sign language animation task in the form of skeletal pose sequences. We propose a Mixture of Motion Primitives (MoMP) network that employs a progressive transformer for sign language translation (Left of Figure 2) and a novel MoE architecture for sign language production (Right of Figure 2). In the remainder of this section we describe each component of MoMP in detail.

3.1. Translation: Progressive Transformer

As shown in Figure 2, we utilise a progressive transformer network [52] for the translation sub-task, which learns to translate from spoken language to sign language representation. A transformer encoder learns a representation of the input spoken language sentence, $x_{1:T}$, to pass to the auto-regressive transformer decoder. Given a sign language sequence, $y_{1:U}$, and the respective counter values, the decoder learns a sign language representation on a per frame basis, $\mathcal{R} = r_{1:U}$:

$$r_{u+1} = \text{Translation}(y_u | y_{i:u-1}, \mathcal{X}) \tag{1}$$

We provide the translation sub-network with additional supervision from gloss information during training, prompting the model to learn a meaningful latent temporal representation for the ultimate goal of sign language production. Due to the lack of frame-level gloss annotation, we use a Connectionist Temporal Classification (CTC) layer to provide supervision in a sequence-to-sequence manner [27]. The CTC layer uses the decoded latent representations for each frame, $r_{1:U}$, and computes $p(\mathcal{Z}|\mathcal{R})$ by marginalising over all possible alignments:

$$p(\mathcal{Z}|\mathcal{R}) = \sum_{\pi \in \mathcal{B}} p(\pi|\mathcal{R}) \tag{2}$$

where π is a path and $\mathcal{B}$ are the set of all viable paths corresponding to $\mathcal{Z}$. The translation loss is then calculated as:

$$\mathcal{L}_{\mathcal{T}} = 1 - p(\mathcal{Z}^*|\mathcal{R}) \tag{3}$$

where $\mathcal{Z}^*$ is the ground truth gloss sequence.

3.2. Animation: Mixture of Motion Primitives

To produce expressive sign poses from the translated sign language representation, we learn a Mixture of Motion Primitives (MoMP) network (right of Figure 2) that combines learnt motion primitives at the frame-level using MoEs. MoEs are a common technique for ensemble learning [31] where an ensemble of $\mathcal{K}$ expert systems, $\{\mathcal{MP}^i\}_{i=1}^k$, are jointly trained, each locally specialised across different domains of expertise to produce an output $y_{u+1}^k = \mathcal{MP}^k(r_u)$.

A gating network, GN, learns a set of blending coefficients, $BC_u = \{\alpha_u^1, ..., \alpha_u^k\}$, used to weight the decision of each expert in the final output. Contrary to traditional MoE architectures that apply a unique blend per sequence [47, 55, 56], we generate unique blending coefficients for each frame of the output sequence. This enables distinct motion primitives to be learnt for certain sections of the output sign language sequence.

Gating Network We utilise a transformer encoder with subsequence masking for the gating network, GN, using self-attention to learn the correct expert allocation. We mask future time-steps as in the transformer decoder, to disable a view of the future. Formally, the gating network produces a set of blending coefficients, BC_{u+1}, conditioned on the translated sign language representation, r_u:

$$BC_{u+1} = \{\alpha_{u+1}^1, ..., \alpha_{u+1}^{\mathcal{K}}\} = GN(r_u | r_{1:u-1}) \tag{4}$$

where a softmax operation is applied to the produced blending coefficients to ensure $\alpha_u^k > 0$ and $\sum_{i=0}^k \alpha_u^k = 1$.

Motion Primitives A continuous sign language sequence consists of multiple distinct sections of motion. For example, a hand moving up then subsequently across the body. Our goal is to represent each of these distinct multi-frame motions as separate motion primitives, which can be temporally combined to produce a full sequence of uninterrupted, continuous signing motion. During training, each learnt motion primitive is encouraged to account for separate sections of the data, becoming specialised for specific motions that can be stitched together at inference.

Similar to the gating network, we build motion primitives, $\mathcal{MP}^k$, using transformer encoders with subsequence masking. We use self-attention over the translated sign language representation to learn the desired motion. We avoid conditioning on the source spoken language to ensure this sub-task is solely focused on animation. Formally, the output of each motion primitive per frame is computed as:

$$y_{u+1}^k = \mathcal{MP}^k(r_u) \tag{5}$$

Each output frame is therefore a sign pose, y_{u+1}, produced in an auto-regressive manner by blending motion primitives, y_{u+1}^k, with their respective blending coefficients, α_{u+1}^k:

$$y_{u+1} = \text{Animation}(r_u | r_{1:u-1}) = \sum_{i=1}^{\mathcal{K}} \alpha_{u+1}^k y_{u+1}^k \tag{6}$$

with $\mathcal{K}$ experts and $\sum_{i=1}^{\mathcal{K}} \alpha_{u+1}^k = 1$. As in [52], the respective counter value is also produced for each frame. Once the full sign pose sequence is produced, the animation loss, $\mathcal{L}_{\mathcal{A}}$ is calculated as the Mean Squared Error (MSE) loss between the predicted, $\hat{y}_{1:U}$, and ground truth, $y_{1:U}^*$ sequences:

$$\mathcal{L}_{\mathcal{A}} = \frac{1}{U} \sum_{i=1}^{u} (y_{1:U}^* - \hat{y}_{1:U})^2 \tag{7}$$

We train our network by minimising the overall SLP loss, $\mathcal{L}_{\mathcal{SLP}}$, which is a weighted sum of the CTC based translation loss, $\mathcal{L}_T$, and the joint distance animation loss, $\mathcal{L}_A$, as:

$$\mathcal{L}_{\mathcal{SLP}} = \lambda_T \mathcal{L}_{\mathcal{T}} + \lambda_A \mathcal{L}_{\mathcal{A}} \tag{8}$$

where λ_T and λ_A weight the importance of each loss function during training and are evaluated in section 4.1.

3.3. Training Schedule

Naive end-to-end training of an MoE with backpropagation has been shown to result in a degenerate local minimum where expert weightings are consistent regardless of the input [47, 56]. Therefore, we use a combination of Block Coordinate Descent (BCD) training and expert balancing losses to overcome this phenomenon, as described below.

Block Coordinate Descent We apply a BCD training schedule, as introduced by Peng *et al*. [47], that decomposes updates into two interleaving steps $\mathcal{G}$ and $\mathcal{F}$: The $\mathcal{G}$ step processes a forward pass with blended outputs, fixes the translation sub-network and motion primitives and updates *only* the gating network, GN; The $\mathcal{F}$ step then freezes the gating network and updates the full translation sub-network alongside a single expert for each frame, $E_k(x)$, sampled from the blending coefficient weights. During training, $\mathcal{G}$ steps are required less often than $\mathcal{F}$ steps, with a ratio of 3 $\mathcal{F}$ steps for each $\mathcal{G}$ step achieving best performance.

Specific to MoEs, BCD forces a specialisation of experts for particular sections of sign pose sequences, learning the important motion primitives. Motivated by the comparison to dropout [47], we add a random chance of selecting expert k, with an annealing of the probability during training.

Expert Balancing Losses As seen in previous MoE architectures [19, 55, 56], we find that a small subset of experts tend to be imbalanced and receive higher blending coefficients. This effect is self-reinforcing as the popular experts are trained quicker, receiving further allocation. In addition, as MoMP produces continuous sign pose sequences with blending coefficients applied per frame, we favour gating networks with sparse activations. This avoids a weighted average of two motion primitives that may itself not be valid.

Following Bengio *et al*. [4], we take a soft constraint approach to expert balancing and apply two regularisation terms. The first term is a balancing loss, $\mathcal{L}_{\mathcal{B}}$, that encourages an equal expert share in expectation:

$$\mathcal{L}_{\mathcal{B}} = \sum_{u=1}^{U} \frac{1}{\mathcal{K}} \sum_{k=1}^{\mathcal{K}} (\alpha_u^k - \tau)^2 \tag{9}$$

where τ is the expected balanced load, $\frac{1}{\mathcal{K}}$. The second term is a variance loss, $\mathcal{L}_{\mathcal{V}}$, that encourages a sparse allocation per frame:

$$\mathcal{L}_{\mathcal{V}} = -\sum_{u=1}^{U} \mathrm{var}_k\{\alpha_u^k\} \tag{10}$$

We add these losses only on the $\mathcal{G}$ step of the BCD training, to solely regularise the gating network. We ablate the proposed training schedule in Section 4.1.

3.4. Sign Language Output

Generating a video from the produced skeleton pose sequence is a trivial task, connecting the relevant joints of each frame as seen in Figure 3.

4. Experiments

Implementation Details In our experiments, we build translation sub-networks with two layers ($2L$) two heads ($2H$) and an embedding size of 256 ($256Em$). The architecture for motion primitives and gating network are ($2L, 2H, 128Em$) and ($2L, 4H, 64Em$), respectively. Our proposed architecture contains only 7.8M parameters, compared to 16.3M for the SOTA model [54]. We apply Gaussian noise with a noise rate of 5, as proposed by Saunders *et al*. [52]. All parts of our network are trained with Xavier initialisation [25], Adam optimization [34] with default parameters and a learning rate of 10^{-4} for the gating network and 10^{-3} for the rest. Our code is based on Kreutzer et al.'s NMT toolkit, JoeyNMT [38], and implemented using PyTorch [46].

Dataset We evaluate our approach on the publicly available PHOENIX14**T** dataset introduced by Camgoz et al. [8]. The corpus provides parallel sequences of 8257 German sentences, sign gloss translations and sign pose videos. This is a challenging dataset due to the low video quality. However, more recent sign language datasets are available, which are yet to be utilized for SLP [11]. We train MoMP to generate sign pose sequences of skeleton joint positions. Manual and non-manual features of each video are extracted in 2D using OpenPose [12], with the manuals lifted to 3D using the skeletal model estimation model proposed in [68]. We normalise the skeleton pose as in [52].

Evaluation To compare against the state-of-the-art, we use the back translation evaluation metric [52], which employs a pre-trained SLT model [10] to translate the produced sign pose sequences back to spoken language. BLEU and ROUGE scores are computed against the original input, with BLEU n-grams from 1 to 4 provided for completeness. The SLP evaluation protocols on the PHOENIX14**T** dataset, set by [52], are as follows: *Gloss to Pose (G2P)* is the production of sign pose from gloss intermediary, evaluating the sign production capabilities; *Text to Pose (T2P)* is the production of sign pose directly from spoken language, and is the more difficult end-to-end test of an SLP system.

4.1. Quantitative Evaluation

Number of Motion Primitives We start our experiments on the *Gloss to Pose* task, and evaluate the production capabilities of the animation sub-network. We therefore set the translation loss, $\mathcal{L}_{\mathcal{T}}$, to zero. Our first experiment evaluates

# of Motion Primitives:	DEV SET BLEU-4	DEV SET ROUGE	TEST SET BLEU-4	TEST SET ROUGE
6	12.67	35.17	12.38	35.29
7	12.57	35.90	12.15	35.37
8	**13.32**	**37.58**	**12.67**	**35.61**
9	12.55	36.14	12.31	34.93
10	12.53	35.90	11.99	34.62

Table 1: Impact of different numbers of motion primitives on the performance of MoMP for the *Gloss to Pose* task.

the performance when varying the number of motion primitive experts, $\mathcal{K}$. Although having a larger number of motion primitives allows each to be more specialised, it also makes the models harder to converge and prone to overfitting. To this end, we build MoMP networks using 6 to 10 primitives and evaluated their *Gloss to Pose* performance.

As shown in Table 1, we find that 8 motion primitives performs best, achieving a 13.32 BLEU-4 score on the dev set. This gives a balance between specialisation of experts and training convergence difficulty, as we find that too many experts leads to an overfit. For the rest of our experiments, we constructed our MoMP model with 8 motion primitives.

Ablation Study We next ablate our MoMP network to highlight the importance of each proposed network attribute. Table 2 shows *Gloss to Pose* model performance. We first remove the randomness applied to the BCD training, as described in section 3.3 (MoMP - Rand). Model performance is significantly degraded, resulting in 12.14 BLEU-4 on the dev set. This is due to the removal of any ability for exploration in the expert update, $\mathcal{F}$, step of BCD.

Removing BCD training entirely (MoMP - BCD) can be seen to negatively impact model performance further, to 10.85 BLEU-4. This is due to the combined update of both gating network and expert parameters leading to an unstable MoE model with non-specialised experts, as seen in previous works [47, 56]. We additionally conduct experiments with a simple EM training (MoMP + EM), alternating updates between the gating network, and a non-sampled combination of motion primitives. However, this still resulted in poor performance of 11.84 BLEU-4.

Removing the expert balance loss (MoMP - $\mathcal{L}_{\mathcal{B}}$) results in an unbalanced gating network that activates only a single motion primitive. This means that the model does not take full advantage of the multiple experts available for specialisation, resulting in a poor performance of 12.20 BLEU-4. Removing the variance loss (MoMP - $\mathcal{L}_{\mathcal{V}}$) causes each frame to have a combination of experts rather than a sparse representation. This results in a blended output which regresses to the mean, causing a non-expressive skeletal pose and poor performance of only 11.88 BLEU-4.

Approach:	DEV SET BLEU-4	DEV SET ROUGE	TEST SET BLEU-4	TEST SET ROUGE
MoMP	**13.32**	**37.58**	**12.67**	**35.61**
MoMP - Rand	12.14	35.67	11.93	35.45
MoMP - BCD	10.85	33.64	10.40	32.11
MoMP + EM	11.84	35.16	11.63	34.71
MoMP - $\mathcal{L}_{\mathcal{B}}$	12.20	35.43	11.72	34.60
MoMP - $\mathcal{L}_{\mathcal{V}}$	11.88	35.47	11.45	34.45

Table 2: Ablation study of MoMP performance for the *Gloss to Pose* task.

Translation and Animation In our next set of experiments, we switch to the full *Text to Pose* task. We examine the performance gain from adding the translation loss, $\mathcal{L}_{\mathcal{T}}$, alongside the animation loss, $\mathcal{L}_{\mathcal{A}}$. As a baseline, we trained a MoMP model solely with the animation loss and with no gloss supervision, by setting the translation weight, $\lambda_{\mathcal{T}}$, to zero. We then jointly train for both translation and animation, with various weightings between the losses.

Table 3 shows experiments on the loss weightings, λ_T and λ_A. As can be seen, jointly training MoMP on both translation and animation with equal weightings ($\mathcal{L}_{\mathcal{T}} = \mathcal{L}_{\mathcal{A}} = 1$) significantly improves the back translation performance to 13.72 BLEU-4. This demonstrates the value of explicitly training on both the translation and animation sub-tasks. Increasing the translation loss weighting to $\mathcal{L}_{\mathcal{T}} = 2$ further increased performance to 14.03 BLEU-4, with even larger translation losses degrading performance. We believe this is due to a required balance between gloss supervision and the ultimate SLP performance.

Baseline Comparison We compare the performance of MoMP against 3 baseline SLP models: 1) Progressive transformers [52], which applied the classical transformer architecture to sign language production. 2) Adversarial training [50], which utilised an adversarial discriminator to prompt more expressive productions and 3) Mixture Density Networks (MDNs) [54], which modelled the variation found in sign language using multiple distributions to parameterise the entire prediction subspace.

Loss Weights $\lambda_{\mathcal{T}}$	Loss Weights $\lambda_{\mathcal{A}}$	DEV SET BLEU-4	DEV SET ROUGE	TEST SET BLEU-4	TEST SET ROUGE
0.0	1.0	12.74	36.17	12.16	35.53
1.0	1.0	13.72	37.63	13.18	36.84
2.0	1.0	**14.03**	**37.76**	**13.30**	36.77
5.0	1.0	13.69	37.67	13.12	**37.10**
10.0	1.0	13.51	36.99	12.83	36.53

Table 3: Impact of different translation and animation loss weightings on MoMP *Text to Pose* performance.

Approach:	DEV SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	TEST SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Progressive Transformers [52]	11.93	15.08	20.50	32.40	34.01	10.43	13.51	19.19	31.80	32.02
Adversarial Training [50]	13.16	16.52	22.42	34.09	36.75	12.16	15.31	20.95	32.41	34.19
Mixture Density Networks [54]	13.14	**16.77**	22.59	33.84	**39.06**	11.94	15.22	21.19	33.66	35.19
MoMP (Ours)	**13.32**	16.71	**22.67**	**34.21**	37.58	**12.67**	**16.03**	**22.02**	**33.95**	**35.61**

Table 4: Back translation results on the PHOENIX14**T** dataset for the *Gloss to Pose* task.

Approach:	DEV SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE	TEST SET BLEU-4	BLEU-3	BLEU-2	BLEU-1	ROUGE
Progressive Transformers [52]	11.82	14.80	19.97	31.41	33.18	10.51	13.54	19.04	31.36	32.46
Adversarial Training [50]	12.65	15.61	20.58	31.84	33.68	10.81	13.72	18.99	30.93	32.74
Mixture Density Networks [54]	11.54	14.48	19.63	30.94	33.40	11.68	14.55	19.70	31.56	33.19
MoMP (Ours)	**14.03**	**17.50**	**23.49**	**35.23**	**37.76**	**13.30**	**16.86**	**23.27**	**35.89**	**36.77**

Table 5: Back translation results on the PHOENIX14**T** dataset for the *Text to Pose* task.

Table 4 shows that MoMP achieves state-of-the-art *Gloss to Pose* results of 13.32/12.67 BLEU-4 for the dev and test sets, respectively. This shows the expressive sign language sequences that can be produced from the animation sub-task, highlighting the effect of the learnt motion primitives within the proposed MoMP network.

Text to Pose results are shown in Table 5, with MoMP achieving 14.03/13.30 BLEU-4 for the dev and test sets, an 11%/14% improvement over the state-of-the-art. These results highlight the significant success of detaching the translation and animation sub-tasks for the ultimate task of SLP.

Furthermore, the performance in the *Text to Pose* task is higher than the *Gloss to Pose* task. This is surprising and significant for SLP, as *Gloss to Pose* is often quoted as the simpler task [52]. We believe this is due to the wider context available in spoken language compared to sign glosses. As we are not forcing translation to go via a gloss bottleneck, the model has access to more subtle grammatical cues for sign production. Instead, we are using the gloss information to supervise end-to-end training from spoken language. This is important for scaling SLP to domains that have limited gloss annotations, which can be expensive to obtain.

Perceptual Study We perform a perceptual study of our skeleton pose productions, showing participants pairs of videos produced by MoMP and the state-of-the-art baseline MDN model [54]. Participants were asked to select which video had the best life-like motion, for the overall skeleton and specifically the hands. In total, 24 participants completed the study, of which 13% were signers. Table 6 shows the percentage of participants who preferred our outputs, the baseline outputs or had no preference between them for both the overall skeleton and hands.

	Ours	Baseline [54]	No Pref.
Skeletons	**49%**	33%	18%
Hands	**50%**	36%	14%

Table 6: Perceptual study results, showing the percentage of participants who preferred our outputs, the baseline output or had no preference, for both overall skeleton and hands.

It can be clearly seen that our outputs were preferred by participants compared to the baseline for both the overall skeleton (49%) and specifically the hands (50%), with only 33% (skeleton) and 36% (hands) preferring the baseline. This further suggests that the proposed MoMP network produces expressive and life-like animations from the combinations of learnt motion primitives.

4.2. Qualitative Evaluation

In this section we report qualitative results. Figure 3 shows signs produced from the source spoken language, alongside the original video. The activated expert is shown for each produced skeleton, highlighting the usage of multiple motion primitives over a full sequence. Expert 5 can be seen to produce right handed motions, whereas expert 2 deals with downward motions of both hands.

Motion Primitives To evaluate the importance of each motion primitive at test time, we selectively deactivate a particular primitive and observe the visual results. Practically, we set the pre-softmax blending coefficient to minus infinity, to force $\alpha^k_{1:U} = 0$. This required the model to activate the other experts for these frames.

We observe the output skeleton pose degrades to a mean pose when expert k was intended to be activated, performing a non-expressive motion. We notice a particular affect on the non-manual features that degrade to a significantly worse output. In addition, we find an average performance drop of 0.83 BLEU-4 when disabling a single motion primitive. We believe this phenomenon is due to expert k becoming specialised for the desired motion, meaning all other experts were not trained to perform the motion.

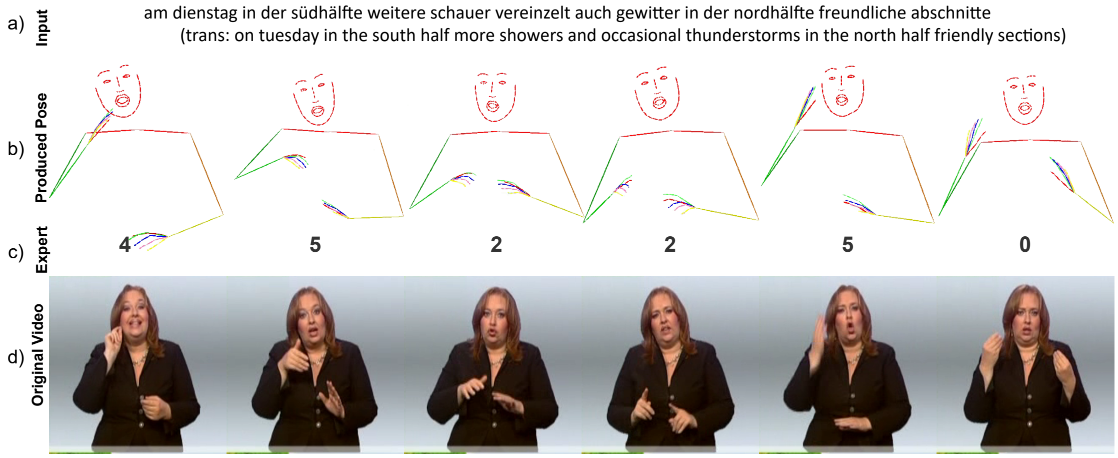

Figure 3: Qualitative results, showing a) Source spoken language, b) Produced sign pose sequence, c) Expert allocation per frame and d) Original video for comparison.

Blending Coefficient Graphs Figure 4 shows some example blending coefficients plotted per sequence. The graphs show the blending coefficients of each weight per frame of a sequence, with each expert plotted as a different colour. As seen, different motion primitives are activated for distinct sections of the sequences, combining unique motions to create a continuous sign language sequence.

The balanced nature of the experts can be seen, as each expert is represented over the sequences. This highlights the effect of the balancing loss, $\mathcal{L}_{\mathcal{B}}$, in ensuring the full repertoire of experts is exploited. In addition, each frame is represented by a single expert, showing the impact of the variance loss, $\mathcal{L}_{\mathcal{V}}$.

5. Conclusion

Even though SLP requires both an accurate translation and expressive production, previous works have combined these tasks into a single end-to-end architecture with one unified loss function [52, 60, 68].

In this paper, we proposed separating the SLP task into two distinct jointly-trained sub-tasks. The first translation sub-task translates from spoken language to sign language representation, with explicit gloss supervision. Subsequently, an animation sub-task produces expressive sign language sequences that closely resemble the gloss representation. Motivated by phonetics, we proposed a Mixture of Motion Primitives (MoMP) architecture, a novel MoE based network that learns to combine distinct motion primitives to produce a continuous sign language sequence.

We evaluated MoMP on the PHOENIX14**T** dataset, with perceptual studies showing that MoMP achieves the best animation quality. We achieved state-of-the-art back translation performance, and reported better SLP performance for direct translation from text, i.e. *Text to Pose*, compared to from gloss intermediaries, i.e. *Gloss to Pose*.

6. Acknowledgements

This project was supported by the EPSRC project ExTOL (EP/R03298X/1), the SNSF project SMILE2 (CRSII5_193686) and the EU project EASIER (ICT-57-2020-101016982). 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. We would also like to thank NVIDIA Corporation for their GPU grant.

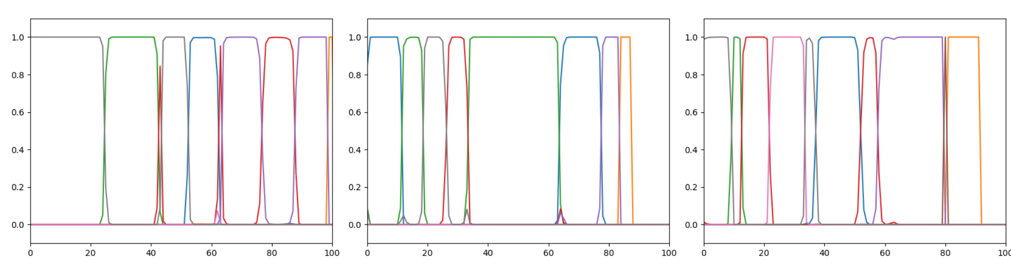

Figure 4: Example expert allocation graphs showing blending coefficient weight per expert against frame.

References

[1] Epameinondas Antonakos, Vassilis Pitsikalis, Isidoros Rodomagoulakis, and Petros Maragos. Unsupervised Classification of Extreme Facial Events using Active Appearance Models Tracking for Sign Language Videos. In *19th IEEE International Conference on Image Processing (ICIP)*, 2012.

[2] J Andrew Bangham, SJ Cox, Ralph Elliott, JRW Glauert, Ian Marshall, Sanja Rankov, and Mark Wells. Virtual Signing: Capture, Animation, Storage and Transmission – an Overview of the ViSiCAST Project. In *Speech and Language Processing for Disabled and Elderly People*, 2000.

[3] Britta Bauer, Hermann Hienz, and K-F Kraiss. Video-Based Continuous Sign Language Recognition using Statistical Methods. In *Proceedings of International Conference on Pattern Recognition (ICPR)*, 2000.

[4] Emmanuel Bengio, Pierre-Luc Bacon, Joelle Pineau, and Doina Precup. Conditional Computation in Neural Networks for Faster Models. *arXiv preprint arXiv:1511.06297*, 2015.

[5] Diane Brentari. *A Prosodic Model of Sign Language Phonology*. Mit Press, 1998.

[6] Diane Brentari. Modality Differences in Sign Language Phonology and Morphophonemics. *Modality and Structure in Signed and Spoken Languages*, 2002.

[7] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. SubUNets: End-to-end Hand Shape and Continuous Sign Language Recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.

[8] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, Hermann Ney, and Richard Bowden. Neural Sign Language Translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[9] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Multi-channel Transformers for Multi-articulatory Sign Language Translation. In *Assistive Computer Vision and Robotics Workshop (ACVR)*, 2020.

[10] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[11] Necati Cihan Camgoz, Ben Saunders, Guillaume Rochette, Marco Giovanelli, Giacomo Inches, Robin Nachtrab-Ribback, and Richard Bowden. Content4All Open Research Sign Language Translation Datasets. In *IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, 2021.

[12] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[13] Jaemin Cho, Minjoon Seo, and Hannaneh Hajishirzi. Mixture Content Selection for Diverse Sequence Generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

[14] David Corina and Wendy Sandler. On the nature of phonological structure in sign language. *Phonology*, 1993.

[15] Stephen Cox, Michael Lincoln, Judy Tryggvason, Melanie Nakisa, Mark Wells, Marcus Tutt, and Sanja Abbott. TESSA, a System to Aid Communication with Deaf People. In *Proceedings of the ACM International Conference on Assistive Technologies*, 2002.

[16] Runpeng Cui, Zhong Cao, Weishen Pan, Changshui Zhang, and Jianqiang Wang. Deep Gesture Video Generation With Learning on Regions of Interest. *IEEE Transactions on Multimedia*, 2019.

[17] Runpeng Cui, Hu Liu, and Changshui Zhang. 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)*, 2017.

[18] Sarah Ebling and Matt Huenerfauth. 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*, 2015.

[19] David Eigen, Marc'Aurelio Ranzato, and Ilya Sutskever. Learning factored representations in a deep mixture of experts. *arXiv preprint arXiv:1312.4314*, 2013.

[20] Ralph Elliott, John RW Glauert, JR Kennaway, Ian Marshall, and Eva Safar. Linguistic Modelling and Language-Processing Technologies for Avatar-based Sign Language Presentation. *Universal Access in the Information Society*, 2008.

[21] William Fedus, Barret Zoph, and Noam Shazeer. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *arXiv preprint arXiv:2101.03961*, 2021.

[22] Jordan Fenlon, Kearsy Cormier, and Diane Brentari. *The Phonology of Sign Languages*. Abigndon/New York: Routledge, 2018.

[23] Ekaterina Garmash and Christof Monz. Ensemble Learning for Multi-Source Neural Machine Translation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016.

[24] JRW Glauert, R Elliott, SJ Cox, J Tryggvason, and M Sheard. VANESSA: A System for Communication between Deaf and Hearing People. *Technology and Disability*, 2006.

[25] Xavier Glorot and Yoshua Bengio. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

[26] Isobel Claire Gormley and Sylvia Frühwirth-Schnatter. Mixtures of Experts Models. In *Handbook of Mixture Analysis*. 2019.

[27] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2006.

[28] Kirsti Grobel and Marcell Assan. Isolated Sign Language Recognition using Hidden Markov Models. In *IEEE International Conference on Systems, Man, and Cybernetics*, 1997.

[29] Thomas Hanke. HamNoSys–Representing Sign Language Data in Language Resources and Language Processing Contexts. In *Workshop on the Representation and Processing of Sign Languages*, 2004.

[30] Xuanli He, Gholamreza Haffari, and Mohammad Norouzi. Sequence to Sequence Mixture Model for Diverse Machine Translation. In *Proceedings of the 22nd Conference on Computational Natural Language Learning (CoNLL)*, 2018.

[31] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. Adaptive Mixtures of Local Experts. *Neural computation*, 1991.

[32] Michael I Jordan and Robert A Jacobs. Hierarchical Mixtures of Experts and the EM Algorithm. *Neural computation*, 1994.

[33] Lukasz Kaiser, Aidan N Gomez, Noam Shazeer, Ashish Vaswani, Niki Parmar, Llion Jones, and Jakob Uszkoreit. One Model to Learn Them All. *arXiv preprint arXiv:1706.05137*, 2017.

[34] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2014.

[35] Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho. Neural Sign Language Translation based on Human Keypoint Estimation. *Applied Sciences*, 2019.

[36] Oscar Koller. Quantitative Survey of the State of the Art in Sign Language Recognition. *arXiv preprint arXiv:2008.09918*, 2020.

[37] Oscar Koller, Necati Cihan Camgoz, Richard Bowden, and Hermann Ney. Weakly Supervised Learning with Multi-Stream CNN-LSTM-HMMs to Discover Sequential Parallelism in Sign Language Videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2019.

[38] Julia Kreutzer, Joost Bastings, and Stefan Riezler. Joey NMT: A Minimalist NMT Toolkit for Novices. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

[39] Roger Lass. *Phonology: An Introduction to Basic Concepts*. Cambridge University Press, 1984.

[40] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling Giant Models with Conditional Computation and Automatic Sharding. *arXiv preprint arXiv:2006.16668*, 2020.

[41] Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H Chi. Modeling Task Relationships in Multi-task Learning with Multi-gate Mixture-of-Experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2018.

[42] Paul Michel, Omer Levy, and Graham Neubig. Are Sixteen Heads Really Better than One? In *Advances in Neural Information Processing Systems (NIPs)*, pages 14014–14024, 2019.

[43] Taro Miyazaki, Yusuke Morita, and Masanori Sano. Machine Translation from Spoken Language to Sign Language using Pre-trained Language Model as Encoder. In *Proceedings of the LREC2020 Workshop on the Representation and Processing of Sign Languages*, 2020.

[44] Alptekin Orbay and Lale Akarun. Neural Sign Language Translation by Learning Tokenization. In *IEEE International Conference on Automatic Face and Gesture Recognition (FG)*, 2020.

[45] Oğulcan Özdemir, Necati Cihan Camgöz, and Lale Akarun. Isolated Sign Language Recognition using Improved Dense Trajectories. In *Proceedings of the Signal Processing and Communication Application Conference (SIU)*, 2016.

[46] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic Differentiation in PyTorch. In *NIPS Autodiff Workshop*, 2017.

[47] Hao Peng, Roy Schwartz, Dianqi Li, and Noah A Smith. A Mixture of $h-1$ Heads is Better than h Heads. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

[48] Prajit Ramachandran and Quoc V Le. Diversity and Depth in Per-Example Routing Models. In *International Conference on Learning Representations (ICLR)*, 2018.

[49] Max Ryabinin and Anton Gusev. Towards Crowdsourced Training of Large Neural Networks using Decentralized Mixture-of-Experts. In *Advances in Neural Information Processing Systems (NeurIPs)*, 2020.

[50] Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Adversarial Training for Multi-Channel Sign Language Production. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2020.

[51] Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Everybody Sign Now: Translating Spoken Language to Photo Realistic Sign Language Video. *arXiv preprint arXiv:2011.09846*, 2020.

[52] Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Progressive Transformers for End-to-End Sign Language Production. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[53] Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. AnonySign: Novel Human Appearance Synthesis for Sign Language Video Anonymisation. In *IEEE International Conference on Automatic Face and Gesture Recognition (FG) (To Appear)*, 2021.

[54] Ben Saunders, Necati Cihan Camgoz, and Richard Bowden. Continuous 3D Multi-Channel Sign Language Production via Progressive Transformers and Mixture Density Networks. In *International Journal of Computer Vision (IJCV)*, 2021.

[55] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *International Conference on Learning Representations (ICLR)*, 2017.

[56] Tianxiao Shen, Myle Ott, Michael Auli, and Marc'Aurelio Ranzato. Mixture Models for Diverse Machine Translation: Tricks of the Trade. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2019.

[57] Thad Starner and Alex Pentland. Real-time American Sign Language Recognition from Video using Hidden Markov Models. *Motion-Based Recognition*, 1997.

[58] William C Stokoe. Sign Language Structure. *Annual Review of Anthropology*, 1980.

[59] Stephanie Stoll, Necati Cihan Camgoz, Simon Hadfield, and Richard Bowden. Sign Language Production using Neural Machine Translation and Generative Adversarial Networks. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[60] Stephanie Stoll, Necati Cihan Camgoz, Simon Hadfield, and Richard Bowden. Text2Sign: Towards Sign Language Production using Neural Machine Translation and Generative Adversarial Networks. *International Journal of Computer Vision (IJCV)*, 2020.

[61] Stephanie Stoll, Simon Hadfield, and Richard Bowden. SignSynth: Data-Driven Sign Language Video Generation. In *Assistive Computer Vision and Robotics Workshop (ACVR)*, 2020.

[62] Valerie Sutton. *Lessons in sign writing: Textbook*. SignWriting, 2014.

[63] Shinichi Tamura and Shingo Kawasaki. Recognition of Sign Language Motion Images. *Pattern Recognition*, 1988.

[64] Nikolai Sergeevich Trubetzkoy. Principles of Phonology. 1969.

[65] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

[66] Qinkun Xiao, Minying Qin, and Yuting Yin. Skeleton-based Chinese Sign Language Recognition and Generation for Bidirectional Communication between Deaf and Hearing People. In *Neural Networks*, 2020.

[67] Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W Cohen. Breaking the Softmax Bottleneck: A High-Rank RNN Language Model. In *International Conference on Learning Representations (ICLR)*, 2018.

[68] Jan Zelinka and Jakub Kanis. Neural Sign Language Synthesis: Words Are Our Glosses. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020.

[69] He Zhang, Sebastian Starke, Taku Komura, and Jun Saito. Mode-Adaptive Neural Networks for Quadruped Motion Control. *ACM Transactions on Graphics (TOG)*, 2018.