# Exploring Visual Representations for Sign Language Translation

**Maram A. Mohamed**
African Masters of Machine Intelligence (AMMI/Senegal)
mmohamed@aimsammi.org

**Xavier Giró**
UPC
xavier.giro@upc.edu

**Laia Tarrés**
UPC
laia.tarres@upc.edu

## Abstract

Sign Language Translation (SLT) task has been addressed in multiple approaches in recent years. In this work we aim to investigate the impact of using different types of visual sign language representation for SLT. For this investigation we use the state-of-the-art in SLT, the Sign Language Transformers model. We compare the translation output performance of two types of body pose estimation models as our skeleton extractor, and 2D CNN features trained on the test dataset. These later perform best, and I3D features outperform the pose estimation-based ones.

## 1 Introduction

Sign Language is the primary communication channel for hearing-impaired people. It consists of manual articulations combined with non-manual elements, such as facial expressions, body poses or mouth motions (Sandler and Lillo-Martin 2006). Together, it results in a very complex natural language. Contrary to common belief, each local community developed and currently uses its own Sign Language, which have been listed to exist 150 different Sign Languages (Eberhard, Simons, and Fennig 2021), although there may exists more which have not been documented yet.

Unfortunately, communication for deaf communities with the hearing ones is very difficult, due to the lack of knowledge on Sign Languages. In this scenario, automatic machine translation for Sign Language appears to be a practical solution and could have significant beneficial impacts.The goal of sign language translation is to either convert written language into a video of sign (production) (Rastgoo et al. 2021) or to extract an equivalent spoken language sentence from a video of someone performing continuous sign (Camgoz, Hadfield, et al. 2018a). Furthermore, a Sign Language has its own linguistic singularities, such as grammar or semantics, as any other language, which are different for each sign. Hence, there is not a one-to-one mapping from the signs to its direct translation, as it is more complex. There exist gloss annotations. For sign languages glosses are the direct transcription of Sign Language that differ from the spoken translation because it contains textual information about the sign and may include words which do not have an equivalent in the spoken language.

In the field of computer vision, much of this latter work has focused on recognising the sequence of sign glosses (Continuous Sign Language Recognition (CSLR)) rather than the full translation to a spoken language equivalent (Sign Language Translation (SLT)). This distinction is important as the grammar of sign and spoken language are very different. These differences include : different word ordering, multiple channels used to convey concurrent information and the use of direction and space to convey the relationships between objects. current state-of-the-art models utilize video recordings containing signers performing Sign Language content to model such tasks. Relaying on the raw video signal is computationally expensive and can lead to overfitting, and non-generalization of the model. In an attempt to overcome this dependence, body pose estimation has been suggested as an alternative being a person-independent, privacy-preserving, and low-dimensional representation source of data, that provides information about the body pose and how the signer changes over time.

The main contributions of this research work can be summarized in the following way: (1) we aim to investigate the impact of using different types of sign language representation for SLT. For this investigation we use the state-of-the-art in SLT, the Sign Language Transformers model(Camgoz, Koller, et al. 2020).(2) We compare the translation output performance of two types of body pose estimation models as our skeleton extractor, as well as the pre-trained 2D Convolution Neural Network (CNN) features provided by(Camgoz, Koller, et al. 2020), following with the performance of I3D features.(3) We adapt the state-of-the-art Sign Language Transformer model to How2Sign and provide baseline results work as starting point for further research on this setup.

This work is organised as follows: In Section 2 we survey the previous studies on SLT and the state-of-the-art model in SLT. In Section 3, we discuss the process of Reproducing and Adapting the Sign Language Transformers. We share our experimental setup with the different types of representations for the input in Section 4. We then report the experimental results of the Sign Language Transformers with different type of features in Section 5 and present new baseline results for the previously defined SLRT model with How2Sign datasets. We conclude the thesis work in Section 7 by discussing our findings and possible future work.

## 2 Related Work

Sign languages have been studied by the computer Vision and NLP community recently (Tamura and Kawasaki 1988), (Starner, Weaver, and Pentland 1998). The end goal of computational sign language research is to build translation and production systems (Cormier et al. 2019), that are capable of translating sign language videos to spoken language sentences and vice versa. Nevertheless, most of the research has mainly focused on Isolated Sign Language Recognition (Joze and Koller 2018), (Yin, Chai, and X. Chen 2016), (Wang, Chai, Hong, et al. 2016), (Camgöz, Kındıroğlu, and Akarun 2016), (Süzgün et al. 2015), (Tornay et al. 2019), working on application specific datasets (Wang, Chai, and X. Chen 2016), (Ebling et al. 2018),(Camgöz, Kındıroğlu, Karabüklü, et al. 2016). More recent work has tackled sign language translation tasks as video based SLT systems and the state-of-the-art work in this approach is Sign Language recognition and Translation in an end-to-end manner(Abadi et al. 2016).

The most important obstacle to vision based SLT research has been the availability of suitable datasets.There are datasets available from linguistic sources (Schembri et al. 2013), (Hanke et al. 2010) and sign language interpretations from broadcasts (Cooper and Bowden 2009). However, the available annotations are either weak (subtitles) or too few to build models which would work on a large domain of discourse. In addition, such datasets lack the human pose information which legacy Sign Language Recognition (SLR) methods heavily relied on.

To address these issues, Camgoz et al. (Camgoz, Hadfield, et al. 2018b) released the first publicly available SLT dataset, PHOENIX14T, which is an extension of the popular RWTH-PHOENIXWeather-2014 (PHOENIX14) CSLR dataset. The authors proposed a system using Convolutional Neural Networks (CNNs) in combination with attention-based NMT methods (Luong, Pham, and Manning 2015), (Bahdanau, Cho, and Bengio 2014) to realize the first end-to-end SLT models. In addition to this, they proposed the current state of the art model for SLT using PHOENIX14T dataset in this work (Bungeroth and Ney 2004). Following this, Ko et al. proposed a similar approach but used body key-point coordinates as input for their translation networks, and evaluated their method on a Korean Sign Language dataset (Ko et al. 2019).Generally, there is no many research works in SLT using keypoints coordinates.

## 2.1 Sign Language Transformers

(Camgoz, Koller, et al. 2020) In their work, the authors introduce a state-of-the-art approach for Sign Language recognition and translation in an end-to-end manner. They present an novel architecture based on Transformers (Cooper and Bowden 2009), which is the current model used to address sequence-to-sequence tasks. In Fig. 1 it can be observed an overview of the architecture of a single Transformer layer, which follows the encoder-decoder classic model. It is designed to generate the English sentence translations from the sign language videos, with intermediate gloss supervision.

On the left side of the image, one can find the encoder model, named SLRT. It correspond to the Sign Language Recognition Transformer. It receives as input the output of the Spatial Embedding layer, which extracts the sign video features from the raw frames. It is summed to the positional encoders (Cooper and Bowden 2009), responsible of adding the temporal coherence of the frame sequence. The SLRT model follows the classic encoder architecture, as can be seen in the figure, with a Self-Attention module, with all operations followed by residual connections and a normalization step. Its goal is to recognize and predict the glosses corresponding to the input sign videos and, more importantly, learning meaning- ful spatio-temporal representations for the further sign language translation. In order to train the encoder, they used the Connectionist Temporal Classification (CTC) sequence- to-sequence learning loss function, instead of cross-entropy loss, as they mention it would require much more precision of gloss annotations, which is not common.

On the left side of the image, one can find the encoder model, named SLRT. It correspond to the Sign Language Recognition Transformer. It receives as input the output of the Spatial Embedding layer, which extracts the sign video features from the raw frames. It is summed to the positional encoders(Vaswani et al. 2017), responsible of adding the temporal coherence of the frame sequence. The SLRT model follows the classic encoder architecture, as can be seen in the figure, with a Self-Attention module, with all operations followed by residual connections and a normalization step. Its goal is to recognize and predict the glosses corresponding to the input sign videos and, more importantly, learning meaningful spatio-temporal representations for the further sign language translation. In order to train the encoder, they used the Connectionist Temporal Classification (CTC) sequence-to-sequence learning loss function, instead of cross-entropy loss, as they mention it would require much more precision of gloss annotations, which is not common.

On the right side of the image, one can find the decoder model, named SLTT. It corresponds to the Sing Language Translation Transformer. It receives as input the output of the Word Embedding layer, which computes a one-hot-vector representation of the English transcriptions into a dense space, and is summed to the positional encoders. The main goal is to generate English sentences from the sign videos representations. The SLTT model follows the classic autoregressive decoder architecture, as can be seen in the figure, with a Masked Self-Attention module and an Encoder-Decoder Attention module, with all operations followed by residual connections and a normalization step. It is important to remark that in the Encoder-Decoder Attention module, the spatio-temporal representations learned from the SLRT are combined with the representations learned from the previous Masked Self-Attention module from the decoder and, there, it learns the mapping between the sign videos and the English transcriptions. In order to train the decoder, they implemented a cross-entropy loss for each word.

The network is trained by minimizing jointly the weighted sum of the recognition and the translation loss multiplied by two hyper-parameters. They trained the model with the PHOENIX2014T (Camgoz, Koller, et al. 2020), a dataset which contains parallel sign language videos, gloss annotations and their written translations, in the weather forecast domain from the German TV. We give further details on the dataset in the following section. Moreover, they specify the following protocols performed to evaluate their model, stated in (Camgoz, Koller, et al. 2020):

- *Sign2Text*, which represents the end goal of translating from a sign video to its spoken transcription, without any intermediary representation.

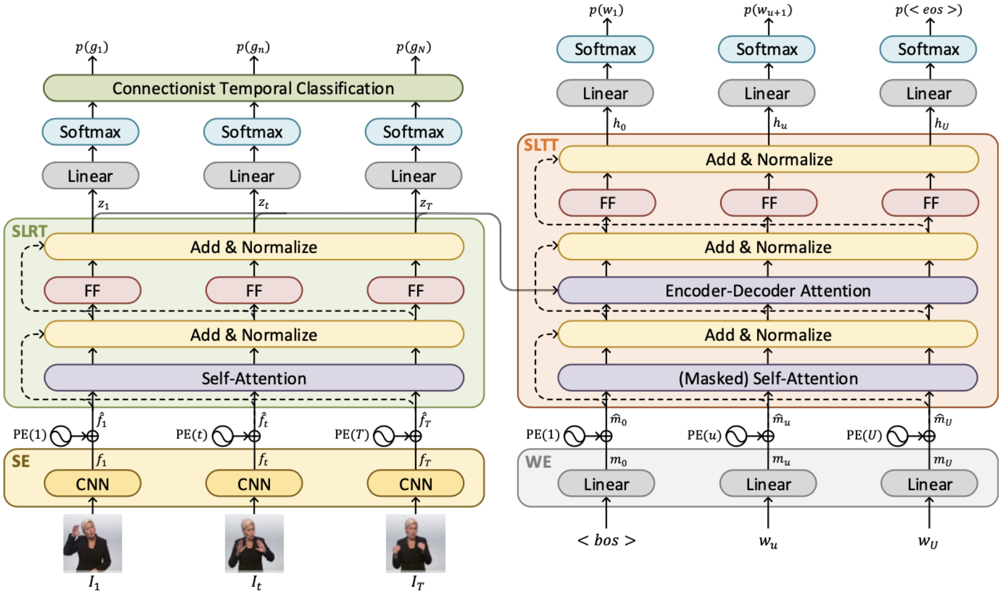

Figure 1: Overview of the architecture of a single Sign Language Transformer layer provided by the authors in their work Camgoz, Koller, et al. 2020

- *Gloss2Text*, a text-to-text translation problem, which consists in translating from the sign glosses to the written sentences.

- *Sign2Gloss2Text*, which represents the current state-of-the-art in Sign Language Transformers. It mainly consists of recognizing the glosses from the sing videos and then using them to predict the written transcriptions.

Additionally, the authors introduced two new evaluation protocols:

- *Sign2Gloss*, which basically consists in sign language recognition from sign videos to glosses.

- *Sign2(Gloss+Text)*, the main contribution of their work, as it represents the joint learning of sign language recognition of glosses and translation to written sentences.

In conclusion, the work presented by Camgoz introduce a novel architecture based on Transformers for learning jointly sign language recognition and translation and provide state-of-the-art results for both tasks, outperforming previous models on the tasks.

## 3 Reproducing and Adapting the Sign Language Transformers

### 3.1 Datasets

The experiments for this work have been conducted using two different datasets namely How2Sign and Phoenix2014T.

#### 3.1.1 Phoenix2014T

This dataset (Camgoz, Koller, et al. 2020) is a large vocabulary, continuous SLT corpus. PHOENIX14T is a translation dataset focused extension of the PHOENIX14 corpus, which has become the primary benchmark for continuous sign language recognition (CSLR) in recent years. PHOENIX14T contains parallel sign language videos, gloss annotations and their translations. The corpus includes unconstrained continuous sign language from 9 different signers with a vocabulary of 1066 different signs. Translations for these videos are provided in German spoken language with a vocabulary of 2887 different words. The dataset version provided by the authors has the text, gloss and sign video already aligned in a structured framework.

### 3.1.2 How2Sign

How2Sign is a Large-scale Multimodal Dataset for Continuous American Sign Language, consisting of a parallel corpus of more than 80 hours of sign language videos and a set of corresponding modalities including speech, English transcripts, and depth.

Nevertheless, the corresponding gloss annotations are not available and, hence, we worked without them. In their place, we used the English text. That is to say, in the recognition task, instead of predicting the gloss annotations, we tried to predict the English transcriptions. This means that the recognition and translation tasks are equivalent in this setup. We have decided to fill the gloss part of the data set with the English sentence because we considered that it was a better temporary solution than filling it with random or arbitrary words, as this way the model could still learn some meaningful representations.

## 3.2 Reproducing the Author's results

Reproducing the author's results was a warm up step for our experiments , Here are some actions we did to reproduce the same results :

## 3.3 Visual Features

The sign information released together with the SLT code does not contain the raw video frames, but the features extracted from a pretrained CNN (Koller, Camgoz, Ney, et al. 2019). This network was pretrained for a sign language recognition task with the PHOENIX2014T dataset, in a CNN+LSTM+HMM configuration.

## 3.4 Code modifications

In order to reproduce the experiment, we had to carry out the following:

First, we prepared the environment for the task. We struggled configuring the versions of the required packages. Therefore, some of the final package versions differed from the specified in the requirements. Second, we modified the data path in the configuration file.

## 3.5 Evaluation Protocol.

To asses the impact of using different visual representations, we use BLEU-4 score(Papineni et al. 2002). BLEU-4 is used for comparing a candidate translation of text to one or more reference translation. For the recognition task we use the Word Error Rate (WER).

## 3.6 Results

After configuring the setup, we trained the SLT model, which took two hours approximately. The obtained results, showed in 1, are similar to the ones provided by the authors (Camgoz, Koller, et al. 2020), in terms of translation (BLEU scores, the higher the better).

| | DEV | | | | | TEST | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Sign2(Gloss+Text) Task | WER | BLEU-1 | BLEU-2 | BLEU-3 | BLEU-4 | WER | BLEU-1 | BLEU-2 | BLEU-3 | BLEU-4 |
| Authors' best results on Recog. | 24.61 | 46.56 | 34.03 | 26.83 | 22.12 | 24.49 | 47.20 | 34.46 | 26.75 | 21.80 |
| Authors' best results on Trans. | 24.98 | 47.26 | 34.40 | 27.05 | 22.38 | 26.16 | 46.61 | 33.73 | 26.19 | 21.32 |
| Our results (Recog. + Trans.) | 48.92 | 44.30 | 31.47 | 24.35 | 19.88 | 48.06 | 44.62 | 31.80 | 24.40 | 19.79 |

Table 1: Comparison between the authors' results and our results obtained with the same setup.

## 4 Sign Language Representation

As mentioned previously, the SLT model was not trained with raw sign videos, but with extracted features that function as Spatial Embeddings (SE). In Fig. 1, this corresponds to the first layer where the video frames are fed to and its output is the input to the Encoder Transformer (SLRT). We conduct our experiments with two different types of data inputs. Firstly, we extract body pose estimation, from the raw videos using OpenPose(Cao, Hidalgo, et al. 2019) and Mediapipe(Lugaresi et al. 2019). Later, we extract R6D angular encodings(Zhou et al. 2019) from Mediapipe. Following, we did another experiment with the pre-trained CNN features provided. Eventually, we used the sign video Embeddings represented as an I3D neural network architecture (Carreira and Zisserman 2017) from (Alvarez, Nieto, and Benet n.d.) to extract the features by(Camgoz, Koller, et al. 2020).

### 4.1 Body Pose estimation

Body pose estimation models have been recently used in different areas of computer vision as an efficient way of representing the human body(Cao, Hidalgo, et al. 2019; Cao, Simon, et al. 2017; Yang et al. 2021). They are robust to different people and background and can be used to preserve the privacy of the person in the video (one of the challenges when working with sign language videos). It is designed to extract 2-dimensional (2D) or 3-dimensional (3D) coordinates that represents the joints of the human body, or also called keypoints (KPs).

The keypoints can be captured via special equipment or directly from video frames. While motion capture equipment can often provide better quality pose estimation, they are still very expensive and intrusive. An alternative to that are pose estimation methods from monocular videos(Cao, Hidalgo, et al. 2019; Xu et al. 2020; Pishchulin et al. 2012; Y. Chen et al. 2017; Güler, Neverova, and Kokkinos 2018). Here we choose the two state-of-the-art body pose estimation models and used them out-of-the-box to extract the keypoints from the raw videos used in our experiments.

**OpenPose (OP)[Cao, Hidalgo, et al. 2019]** is an open-source library for multi-person 2-dimensional keypoints detection. It is designed to estimate a total of 137 keypoints being 70 keypoints from the face, 25 keypoints for the body and 21 keypoints for each hand. Although the 2D representation of the body provided by this method can be used to represent the signer's body, we use the 2D to 3D lifting method proposed by(Zelinka and Kanis 2020) in order to obtain the 3D representation of the sign language videos. This pipeline was used by previous methods as a sign language representation for sign language production(Zelinka and Kanis 2020; Saunders, Camgoz, and Bowden 2020; Viegas et al. 2022).

**Mediapipe v.07 (MP)**(Lugaresi et al. 2019) is a 3D body pose estimation library that provides a total of 543 keypoints, being 468 keypoints for the face, 33 for the body pose and 21 keypoints for each hand.

**6D Rotational Representation (R6D)**(Zhou et al. 2019) Although this 3D Cartesian representation(Keypoints) allows handling occlusions and different camera angles much more effectively, it suffers from sensitivity to scale and length of the speaker's limbs. To obtain a representation that is invariant to changes in scale, we decided to convert the Cartesian coordinates to a 6D rotational representation called R6D. In essence, with a rotational representation such as axis-angle or R6D, we represent a body joint as its rotation with respect to its parent joint.

The conversion process involves a number of steps. Since we compute the rotation of a joint against its parent, we perform the conversion by traversing a kinematic tree. It is necessary to define a root bone, from which to start the traversal. We set the root bone to be the neck, since it is a very non-informative body part of a signer and it is therefore safe to assume it remains fixed. For each triplet of joints, we compute vectors u and v, representing the parent bone and the "rotated" one, respectively. Then, we obtain the axis a^ and angle  by which the rotation from the parent to the child joints is achieved. Vector a^ is the axis-angle representation of the central joint w.r.t to its parent joint. From this, we can easily obtain its rotation matrix and its R6D representation (which corresponds to vectorizing the first two columns of the rotation matrix)

### 4.2 Pre-trained 2D CNN features

We use the pre-trained CNN features(Koller, Camgoz, Bowden, et al. 2019) provided by the authors of(Camgoz, Koller, et al. 2020). This network was used for the sign language recognition task with the PHOENIX2014T dataset (Camgoz, Hadfield, et al. 2018a). In their work, the authors trained a CNN feature extractor and use an LSTM classifier that produced weak and noisy labels, later refined by a multi-stream Hidden-Markov-Model (HMM). The provided feature vectors are of 1024 dimensions.

### 4.3 I3D Features

We used the sign video Embeddings represented as an I3D neural network architecture [3] from [5] to extract the features. I3D features are new Two-Stream Inflated 3D ConvNet (I3D) that is based on 2D ConvNet inflation: filters and pooling kernels of very deep image classification ConvNets are expanded into 3D, making it possible to learn seamless spatio-temporal feature extractors from video while leveraging successful ImageNet architecture designs and even their parameters.

## 5 Experiments

In this section , different experiments with our setups will be discussed. Generally speaking, The experiments have been conducted using PHOENIX2014T dataset and few of them with How2Sign dataset, taking into account the model by Camgoz et al., Sign Language Tranformer.We first go over the implementation details and introduce the evaluation metrics we will be using to measure the performance of our models.

**Method.** We have used the public implementation of SLRT(Camgoz, Koller, et al. 2020) to develop our experiments. The overall model is trained by optimizing a weighted sum of the recognition and the translation loss. This configuration is referred as *Sign2(Gloss+Text)*. We use the same configuration as the authors for our experiments.

**Dataset.** We use the publicly available PHOENIX2014T dataset(Camgoz, Hadfield, et al. 2018a). It comprises a collection of weather forecast videos in German SL, segmented into sentences and accompanied by German transcripts and sign-gloss annotations. The dataset is divided into training, validation, and test set with respectively 7,096, 519, and 642 sentences.

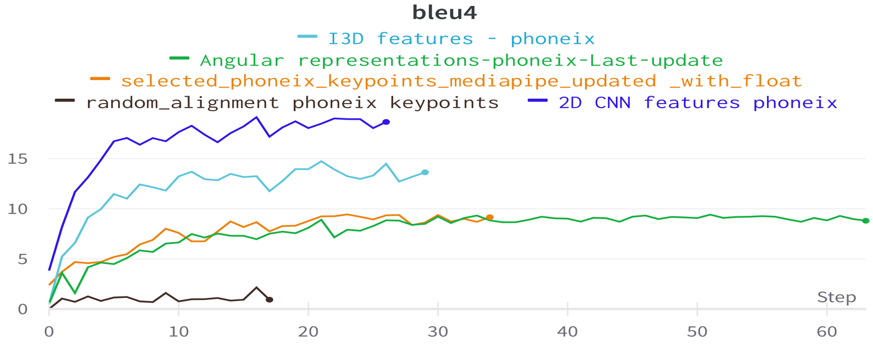

Figure 2: BLEU-4 metric for the validation dataset at each epoch based on different types of input representations.

**Results.** Our contributions are mostly in exploring different alternatives based on keypoints (KPs). We adapted the implementation of SLT(Camgoz, Koller, et al. 2020) to work with keypoints by using a smaller dimension in their input linear layer. Following the approach in(Saunders, Camgoz, and Bowden 2020) when using 3D keypoints features for the PHOENIX2014T dataset, we used 50 KPs instead of the 137 KPs provided by OpenPose because most of the information is contained in the

Table 2: Results for the different types of visual representations

| Features | Validation | Test |
|---|---|---|
| KPs OpenPose (random) | 2.19 | 4.05 |
| KPs OpenPose Cao, Hidalgo, et al. 2019 | 10.50 | 11.00 |
| KPs MediaPipe Lugaresi et al. 2019 | 9.16 | 9.50 |
| KPs MediaPipe Zhou et al. 2019 (R6D) | 8.80 | 9.56 |
| I3D features | 14.74 | 13.63 |
| 2D CNN(Camgoz, Koller, et al. 2020) | 19.88 | 19.79 |

hands and arms. Similarly, for the MediaPipe case we selected the KPs for these body parts, which resulted in 48 keypoints.

As a sanity check, we also trained the model with wrong pairs of OpenPose KPs and spoken English transcriptions. In this set up, the decoder may at most learn an English language model, but will ignore the visual features as they are not related with the translated phrase.

In the case of the CNN features, we used the ones provided by the same authors of SLT(Camgoz, Koller, et al. 2020) and successfully reproduced their results.

We present the results in Table 2. We can see that the 2D CNN features outperforms the keypoints with this model. Compared to CNN features, using keypoints contributes negatively to the results even when we compute the R6D features, which should be more robust than their Cartesian counterparts. Moreover, we took advantage of the I3D features extracted from PHOENIX2014T dataset , as we can see the results obtained were better than the ones from keypoints but still the results with the CNN features were the best.

We hypothesize that there are multiple causes for this poor performance of KP-based features. First, the automatic detection of poses is noisy, which introduces more difficulty to this problem. Second, the skeletal dimension is reduced by almost 10 compared to the CNN features, which even if we are keeping the most important information, some information are lost. Third, when working with keypoints, we know the kinematic tree, but this information is not known by the model, making it specially difficult to learn. Lastly, the translation model has been specifically designed and its hyperparameters optimized to work with the CNN features trained on the PHOENIX dataset, which might cause them to outperform other visual features.

### 5.1 Adapting to How2Sign

We also trained the model with the multimodal American Sign Language dataset, How2Sign. It represents a much more complete and extensive dataset in the Sign Language field than PHOENIX2014T, the one used by the original authors. Moreover, we took advantage of the keypoints computed with the How2Sign video frames. Nevertheless, there are several limitations we have faced during the development of this work that should be addressed, Firstly, the servers we work on have a time limit of 24 hours and, consequently, non of the models we trained actually converged. As the environment is already designed to train from the checkpoints, we plan to continue our research by retraining our current models until they converge. Following that, the annotation glosses are not available, hence the recognition task itself cannot in fact be fulfilled. This represents a major problem, due to the fact that the original authors show that the joint learning of the translation and recognition tasks improves substantially the performance, than learning them separately. Finally, the data was miss-aligned in terms of text and frame pairs and this affect the results with no doubt. As a result the performance of the model on How2Sign was mostly random.

## 6 Conclusions and future work

In this work, we have explored keypoint-based visual representations for Sign Language Translation on the PHOENIX2014T dataset. We conclude that for this specific model and dataset, the 2D CNN features trained for sign recognition in the dataset are the best option. Although we have not been able to successfully train a model with any of the keypoint-based features, we consider it is still an interesting venue that should be further studied. In the future, we aim at exploring the limitations of the keypoint-based features, as we expect that the performance of the model can be improved if the performance of the human keypoint detection algorithm also improves. We will also consider other features based 3D CNNN features(Renz et al. 2021) or optical flow(Moryossef et al. 2020). We also think it is important to evaluate the performance of the model for different sign language datasets as the PHOENIX2014T dataset is focused on the very specific domain of weather forecast.

## 7 Acknowledgement

We acknowledge Particia Cabot from UPC for her help in setting up some of our experiments. Also, initial discussion with Amanda Duarte provided useful insights into how to proceed in this line of research.

## References

Tamura, Shinichi and Shingo Kawasaki (1988). “Recognition of sign language motion images”. In: *Pattern recognition* 21.4, pp. 343–353.

Starner, Thad, Joshua Weaver, and Alex Pentland (1998). “Real-time american sign language recognition using desk and wearable computer based video”. In: *IEEE Transactions on pattern analysis and machine intelligence* 20.12, pp. 1371–1375.

Papineni, Kishore et al. (2002). “BLEU: A Method for Automatic Evaluation of Machine Translation”. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Bungeroth, Jan and Hermann Ney (2004). “Statistical sign language translation”. In: *Workshop on representation and processing of sign languages, LREC*. Vol. 4, pp. 105–108.

Sandler, Wendy and Israel Diane Lillo-Martin (2006). “Sign Language and Linguistic Universals”. In: *Cambridge: Cambridge University Press*.

Cooper, Helen and Richard Bowden (2009). “Learning signs from subtitles: A weakly supervised approach to sign language recognition”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 2568–2574.

Hanke, Thomas et al. (2010). “DGS Corpus & Dicta-Sign: The Hamburg Studio Setup”. In: *4th workshop on the representation and processing of sign languages: corpora and sign language technologies (CSLT 2010), Valletta, Malta*. Vol. 6.

Pishchulin, Leonid et al. (2012). “Articulated people detection and pose estimation: Reshaping the future”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, pp. 3178–3185.

Schembri, Adam et al. (2013). “Building the British sign language corpus”. In: *Language Documentation & Conservation* 7, pp. 136–154.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2014). “Neural machine translation by jointly learning to align and translate”. In: *arXiv preprint arXiv:1409.0473*.

Luong, Minh-Thang, Hieu Pham, and Christopher D Manning (2015). “Effective approaches to attention-based neural machine translation”. In: *arXiv preprint arXiv:1508.04025*.

Süzgün, Muhammed et al. (2015). “Hospisign: an interactive sign language platform for hearing impaired”. In: *Journal of Naval Sciences and Engineering* 11.3, pp. 75–92.

Abadi, Martın et al. (2016). “{TensorFlow}: A System for {Large-Scale} Machine Learning”. In: *12th USENIX symposium on operating systems design and implementation (OSDI 16)*, pp. 265–283.

Camgöz, Necati Cihan, Ahmet Alp Kındıroğlu, and Lale Akarun (2016). “Sign language recognition for assisting the deaf in hospitals”. In: *International Workshop on Human Behavior Understanding*. Springer, pp. 89–101.

Camgöz, Necati Cihan, Ahmet Alp Kındıroğlu, Serpil Karabüklü, et al. (2016). "BosphorusSign: a Turkish sign language recognition corpus in health and finance domains". In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pp. 1383–1388.

Wang, Hanjie, Xiujuan Chai, and Xilin Chen (2016). "Sparse observation (so) alignment for sign language recognition". In: *Neurocomputing* 175, pp. 674–685.

Wang, Hanjie, Xiujuan Chai, Xiaopeng Hong, et al. (2016). "Isolated sign language recognition with grassmann covariance matrices". In: *ACM Transactions on Accessible Computing (TACCESS)* 8.4, pp. 1–21.

Yin, Fang, Xiujuan Chai, and Xilin Chen (2016). "Iterative reference driven metric learning for signer independent isolated sign language recognition". In: *European Conference on Computer Vision*. Springer, pp. 434–450.

Cao, Zhe, Tomas Simon, et al. (2017). "Realtime multi-person 2d pose estimation using part affinity fields". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7291–7299.

Carreira, Joao and Andrew Zisserman (2017). "Quo vadis, action recognition? a new model and the kinetics dataset". In: *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308.

Chen, Yu et al. (2017). "Adversarial posenet: A structure-aware convolutional network for human pose estimation". In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1212–1221.

Vaswani, Ashish et al. (2017). "Attention is all you need". In: *Advances in neural information processing systems* 30.

Camgoz, Necati Cihan, Simon Hadfield, et al. (2018a). "Neural Sign Language Translation". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

– (2018b). "Neural sign language translation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7784–7793.

Ebling, Sarah et al. (2018). "SMILE Swiss German sign language dataset". In: *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC) 2018*. University of Surrey.

Güler, Rıza Alp, Natalia Neverova, and Iasonas Kokkinos (2018). "Densepose: Dense human pose estimation in the wild". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7297–7306.

Joze, Hamid Reza Vaezi and Oscar Koller (2018). "Ms-asl: A large-scale data set and benchmark for understanding american sign language". In: *arXiv preprint arXiv:1812.01053*.

Cao, Zhe, Gines Hidalgo, et al. (2019). "OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields". In: *IEEE transactions on pattern analysis and machine intelligence* 43.1, pp. 172–186.

Cormier, Kearsy et al. (2019). "Extol: Automatic recognition of british sign language using the bsl corpus". In: *Proceedings of 6th Workshop on Sign Language Translation and Avatar Technology (SLTAT) 2019*. University of Surrey.

Ko, Sang-Ki et al. (2019). "Neural sign language translation based on human keypoint estimation". In: *Applied Sciences* 9.13, p. 2683.

Koller, Oscar, Necati Cihan Camgoz, Richard Bowden, et al. (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*.

Koller, Oscar, Necati Cihan Camgoz, Hermann Ney, et al. (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* 42.9, pp. 2306–2320.

Lugaresi, Camillo et al. (2019). "Mediapipe: A framework for building perception pipelines". In: *arXiv preprint arXiv:1906.08172*.

Tornay, Sandrine et al. (2019). "Hmm-based approaches to model multichannel information in sign language inspired from articulatory features-based speech processing". In: *ICASSP 2019-2019*

*IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 2817–2821.

Zhou, Yi et al. (2019). "On the Continuity of Rotation Representations in Neural Networks". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5738–5746. DOI: 10.1109/CVPR.2019.00589.

Camgoz, Necati Cihan, Oscar Koller, et al. (2020). "Sign Language Transformers: Joint End-to-end Sign Language Recognition and Translation". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Moryossef, Amit et al. (2020). "Real-time sign language detection using human pose estimation". In: *European Conference on Computer Vision*. Springer, pp. 237–248.

Saunders, Ben, Necati Cihan Camgoz, and Richard Bowden (2020). *Progressive Transformers for End-to-End Sign Language Production*. arXiv: 2004.14874 [cs.CV].

Xu, Hongyi et al. (2020). "Ghum & ghuml: Generative 3d human shape and articulated pose models". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6184–6193.

Zelinka, Jan and Jakub Kanis (2020). "Neural Sign Language Synthesis: Words Are Our Glosses". In: *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 3384–3392. DOI: 10.1109/WACV45572.2020.9093516.

Eberhard, David M., Gary F. Simons, and Charles D. Fennig (2021). *"Sign language", Ethnologue: Languages of the World*. https://www.ethnologue.com/subgroups/sign-language.

Rastgoo, Razieh et al. (June 2021). "Sign Language Production: A Review". In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 3451–3461.

Renz, Katrin et al. (2021). "Sign Language Segmentation with Temporal Convolutional Networks". In: *ICASSP*.

Yang, Sen et al. (2021). "Transpose: Keypoint localization via transformer". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11802–11812.

Viegas, Carla et al. (Feb. 2022). *Including Facial Expressions in Contextual Embeddings for Sign Language Generation*.

Alvarez, Patricia Cabot, Xavier Giró Nieto, and Laia Tarrés Benet (n.d.). "Sign Language Translation based on Transformers for the How2Sign Dataset". In: ().