Isolated Sign Recognition from RGB Video using Pose Flow and Self-Attention

Mathieu De Coster
IDLab-AIRO - Ghent University - imec
Technologiepark-Zwijnaarde 126, Ghent, Belgium
mathieu.decoster@ugent.be

Mieke Van Herreweghe
Ghent University
Blandijnberg 2, Ghent, Belgium
mieke.vanherreweghe@ugent.be

Joni Dambre
IDLab-AIRO - Ghent University - imec
Technologiepark-Zwijnaarde 126, Ghent, Belgium
joni.dambre@ugent.be

Abstract

Automatic sign language recognition lies at the intersection of natural language processing (NLP) and computer vision. The highly successful transformer architectures, based on multi-head attention, originate from the field of NLP. The Video Transformer Network (VTN) is an adaptation of this concept for tasks that require video understanding, e.g.*, action recognition. However, due to the limited amount of labeled data that is commonly available for training automatic sign (language) recognition, the VTN cannot reach its full potential in this domain. In this work, we reduce the impact of this data limitation by automatically pre-extracting useful information from the sign language videos. In our approach, different types of information are offered to a VTN in a multi-modal setup. It includes per-frame human pose keypoints (extracted by OpenPose) to capture the body movement and hand crops to capture the (evolution of) hand shapes. We evaluate our method on the recently released AUTSL dataset for isolated sign recognition and obtain 92.92% accuracy on the test set using only RGB data. For comparison: the VTN architecture without hand crops and pose flow achieved 82% accuracy. A qualitative inspection of our model hints at further potential of multi-modal multi-head attention in a sign language recognition context.*

1. Introduction

Research into deep learning techniques for computer vision and natural language processing techniques is progressing rapidly. Sign language recognition and translation lie at the intersection of both areas. While a lot of progress has recently been made towards actual sign language translation, it is still a highly complex problem beyond the current state of the art. Most research still focuses on the sub-domains of isolated and continuous sign recognition. This ignores several grammatical aspects of sign languages by simplifying the problem to a classification task or an alignment and classification task, respectively. Current methods devised for recognizing signs in video are unlikely to generalize to actual sign language translation as they are. However, these sub-domains allow the development of feature extraction methods and other information extraction algorithms that can provide input to future sign language translation research.

Several recent works evaluate the use of transformers in sign language translation [6, 7, 35]. The multi-head attention mechanism can also be applied to isolated sign recognition. The Video Transformer Network (VTN), originally proposed by Kozlov *et al*. [20], was used for isolated sign recognition on the corpus of Flemish sign language and achieved promising results (74.7% accuracy on 100 classes [10]), which were mainly limited by the size of the labeled dataset. The VTN architecture consists of a 2D CNN as feature extractor and multi-head attention layers as sequence processing network, as a replacement of a recurrent neural network or variant thereof, *e.g*., a Long Short-Term Memory (LSTM) network.

The work presented in this paper was performed in the context of the ChaLearn 2021 Looking at People Large Scale Signer Independent Isolated SLR CVPR Challenge [26]. The AUTSL dataset [27] used in this competition provides RGB-D data captured using a Kinect camera. Because we believe that the need for any additional hardware (beyond a simple camera) would dramatically reduce the usefulness of sign language translation systems, we consider only the RGB data for this work.

The AUTSL dataset was created for the development of isolated sign recognition algorithms, with varying backgrounds and multiple persons. It consists of 36,302 samples

from 226 sign categories. It is signer-independent: each of the 43 signers occurs only in either training, validation, or test set. This is especially important because a powerful model will pick up particularities about individual persons. When the same person(s) occur in train, validation and test sets, validation and test scores will be overly optimistic due to data leakage. In contrast, AUTSL allows for a realistic evaluation of the scores that can be achieved for entirely unknown persons. Section 3.1 presents more information on this dataset and how we use it to train our models.

In its original form, the VTN achieves a 82% accuracy on the validation set of the AUTSL dataset. The performance of this highly powerful network is largely constrained by the limited amount of labeled data [10]. We propose to overcome this by pre-extracting relevant information and offering only this to the network. As a first step, by using only the two cropped hand images as inputs rather than a full body image, we increase the accuracy by 8.1% (absolute increase). A further improvement is achieved by adding body pose motion information (extracted with OpenPose [8]) to distinguish better between signs with similar hand shapes but different movements. This further increases the accuracy by 1.4% (absolute increase).

Besides their power, the use of attention in transformers also offers increased interpretability of the trained models. Through visual analysis, we can learn what our model is attending to and which important information may still be missing. We believe that such visual analysis of attention in our models can support future interaction with sign language linguists and facilitate the transition to true automatic sign language understanding and translation.

In summary, the contributions of this work are:

- We extract pre-processed multi-modal input from RGB video data to considerably increase the performance of an existing VTN model for isolated sign recognition (AUTSL dataset).
- Our approach introduces pose flow, a method inspired by optical flow, to represent body movements based on pose keypoints.
- We qualitatively analyze our model, showing some interesting properties of multi-head attention when applied to isolated sign recognition.

The source code and weights of our model are publicly available[1].

2. Background

Sign language recognition is the field of research that aims to extract the information that is needed to automatically understand or translate sign language from a continuous input stream. As this work only explores the sub-task of isolated sign recognition, the focus of this section lies on the background in this domain. A broader overview of contributions to sign recognition, including continuous sign recognition, can be found in the works of Bragg *et al*. [4] and Koller [19].

2.1. Isolated sign recognition

Arguably the least intrusive method to recognize sign language possible today is through the use of consumer-grade cameras available in mobile phones. Early research in the domain of isolated sign recognition based on video data, *e.g*., the seminal work of Tamura *et al*. [29], uses computer vision algorithms such as color thresholding to extract features. Grobel *et al*. [12] extract features based on parameters of sign language: hand location, orientation, and shape. Vogler *et al*. extract information such as bending factor of fingers and movements of the hands and use these as independent channels in a recognition system [33]. In the previous decade research shifted towards end-to-end deep learning, encouraged by the success of Convolutional Neural Networks (CNNs) on computer vision problems and Recurrent Neural Networks (RNNs) on sequence processing problems. Promising initial results were achieved in the domain of sign language recognition using end-to-end deep learning. Pigou *et al*. use a 2D CNN for sign recognition on Dutch and Flemish sign language [23]. With the advent of off-the-shelf pre-trained human pose estimation systems such as OpenPose [8], several sign language recognition researchers applied recurrent neural networks using keypoints as input features [18, 17, 10]. However, because movements in sign language can be quick (leading to motion blur), and because there is occlusion between and within the hands, these keypoints can be noisy [10]. Furthermore, recent works have shown that end-to-end models can significantly outperform pose based models [30, 21, 1]. The work that introduces the WLASL dataset for isolated sign recognition compares several deep learning architectures: a 2D CNN followed by an RNN, a 3D CNN, a pose RNN and a pose Temporal Graph Convolutional Network [21]. The pose based networks use OpenPose [8] keypoints as input features. The authors find that the 3D CNN, specifically I3D, outperforms the other networks. Albanie *et al*. further increased the performance on WLASL using transfer learning [1]. They first train the 3D CNN I3D [9] on a dataset extracted from the British sign language corpus [24] and use it for transfer learning to other datasets, including MS-ASL [30] and WLASL [21].

Signs have several phonological components. Stokoe acknowledges hand shape, movement and place of articulation as parameters of signs [28]. Recognizing hand shapes is clearly an important part of a sign language recognition

[1] https://github.com/m-decoster/ChaLearn-2021-LAP

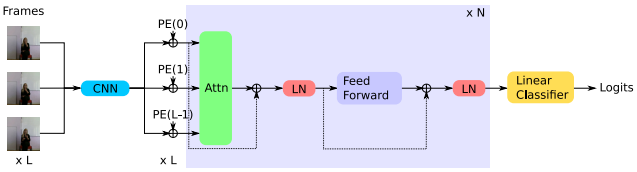

Figure 1. The video transformer consists of a spatial feature extractor (CNN), followed by a self-attention decoder with N layers. $PE(i)$ is the positional encoding for the ith feature vector in the sequence; $Attn$ is multi-head dot-product attention; LN is layer normalization.

system. We see for example that models can learn to focus on the hand regions [27]. However, it is also possible to increase the inductive bias of sign language recognition models by cropping out hand images and optionally using a feature extractor that is pre-trained on a hand shape classification task [5, 6].

The transformer architecture [31] shows very promising results for machine translation. Since its introduction in 2017, the transformer has re-defined the state of the art of natural language processing. It is also being applied in sign language translation [6, 7, 35]. De Coster *et al*., on the other hand, use a 2D CNN and transformer for isolated sign recognition [10]. Specifically, they use the VTN architecture, which comprises a 2D CNN followed by several attention layers [20]. They remove the (cross-attention) decoder from the transformer and only use the (self-attention) encoder layers without masking. Their results are promising. Our work evaluates the VTN on the publicly available AUTSL [27] dataset. We find that the model is able to perform isolated sign recognition with high accuracy on AUTSL and can be improved through pre-processing.

2.2. Video transformer network

Kozlov *et al*. proposed the VTN for the task of action recognition [20]. The VTN comprises a 2D CNN as spatial feature extractor, followed by several attention layers. These attention layers are extracted from the encoder of the original transformer, as they perform self-attention. However, in the context of the VTN, they act as a decoder. Therefore, we further refer to this component of the VTN as the self-attention decoder. This decoder is built from several blocks, with each block containing a residual multi-head attention layer and a residual feed forward network, with layer normalization [2] in between and at the end. As this is a *self*-attention decoder, the multi-head attention is computed from query, key and value originating from the same input sequence. The positional encoding for each element in the sequence is added to the feature vectors before the first multi-head attention layer in the decoder. The VTN uses the fixed positional encoding from the original transformer [31]. This encoding is required to provide information on the ordering of the sequence to the network. The VTN architecture is illustrated in Figure 1.

We now detail dot-product self-attention specifically in the context of the VTN. The inputs to the self-attention network are denoted as X, consisting of the feature vectors extracted by the CNN. X is transformed into the queries Q, keys K and values V through trainable linear transformations $Q = XW^Q$, $K = XW^K$ and $V = XW^V$. Each head i of the n heads computes attention on a subset of the query, key and value,

$$Q^i = QW^i_Q, \tag{1}$$

$$K^i = KW^i_K, \tag{2}$$

$$V^i = VW^i_V, \tag{3}$$

$$A(Q^i, K^i, V^i) = \text{softmax}\left(\frac{Q^i K^{i\top}}{\sqrt{1/d_k}}\right) V^i, \tag{4}$$

where weight matrices W^i_Q, W^i_K and W^i_V are used to transform the query, key and value to a lower-dimensional subspace. Each head operates on a subspace of the input size $d_k = d_{model}/n$. The outputs of all heads are concatenated.

2.3. Pose estimation in sign language recognition

We observe in literature that OpenPose keypoints can be noisy in sign language videos because of occlusions and fast movements [10]. Furthermore purely pose based networks are outperformed by end-to-end systems [10, 21]. We believe, however, that pose estimation can still be useful as an augmentation to raw RGB video data or to compute low-dimensional representations. For example, Albanie *et al*. apply pose distillation (regressing poses from video) as a pre-training step [1]. In this work, we use pose data for pre-processing and to encode movement.

3. Methods

As we use a VTN, we model spatial information using deep CNNs and temporal information using self-attention. We propose several improvements to the VTN. This section details the applied methods. Section 4 discusses our iterative approach to the development of our final model.

3.1. Dataset

We use the balanced AUTSL dataset [27] for our experiments. This dataset consists of 36,302 samples. Each sample corresponds to one of 226 signs, and is performed by one of 43 different persons. The dataset is split in signer independent training, validation, and test sets. The videos are filmed at different locations and from different viewpoints. All samples are provided as separate RGB and depth video files with a spatial resolution of 512 by 512 pixels and a temporal resolution of 30 frames per second (FPS). We only use the RGB data for our experiments in this work.

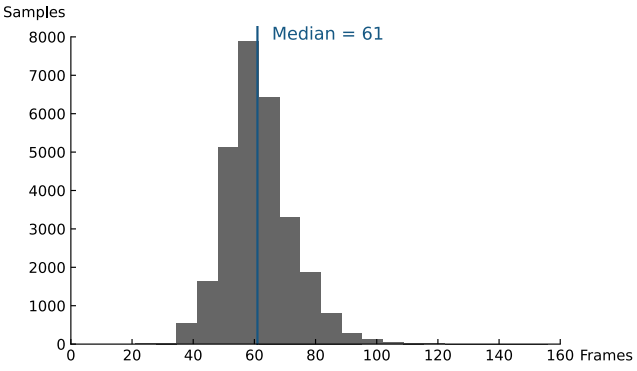

Figure 2. The samples have varying lengths; the median length in the training set is 61 frames (approximately 2 seconds).

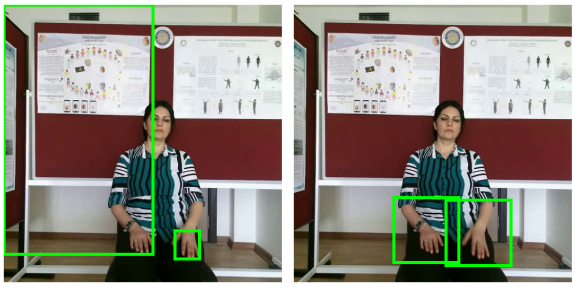

Figure 3. Cropping based on the hand keypoints (on the left) can lead to distorted crops if keypoints are missing. The wrist and elbow keypoints are predicted correctly more often than hand keypoints by OpenPose, and cropping based on these keypoints (on the right) is more robust.

The training set contains 28,142 samples from 31 different signers, the validation set 4,418 samples from 6 different signers and the test set 3,742 samples from 6 different signers.

The samples have varying lengths, with a median of 61 frames: see Figure 2. Every sample has wind-up and wind-down segments in the beginning and end of the video. In sign language conversations, this would not be the case: there would be fluent transitions between signs and at times both hands engage in simultaneous constructions [32]. For the purpose of isolated sign recognition on this dataset, we do not consider the wind-up and wind-down segments. Instead, we decide to select a segment from the middle of the video. We select 16 frames with a stride of 2 frames for an effective temporal receptive field of 32 frames. We decide that this is an appropriate approach after visual inspection and from the validation set accuracy of our models. Note that 32 frames corresponds to slightly more than a second, as the input videos are filmed at 30 FPS.

We find several outliers in terms of sample lengths in the training set. Visual inspection shows us that there are two main causes for the length of these samples. Either the signs are performed slowly and deliberately, or the samples contain repetitions of the sign. We decide not to alter or remove these samples: different people sign with different speeds and repetitions are unlikely to cause issues with training or inference as we perform a temporally centered selection of frames in our experiments.

3.2. Hand cropping

For isolated sign recognition, hand shape, orientation, movement and place of articulation are arguably the most important parameters to recognize. Non-manual components such as mouthings, eye gaze and eyebrow movements are also crucial elements of sign languages [3, 22]. However, we do not consider them here, for the task of isolated sign recognition, because they are less important when merely distinguishing between individual signs.

The videos in the AUTSL dataset have a spatial resolution of 512 by 512 pixels. Considering that our model is trained with inputs of 224 by 224 pixels, non-negligible spatial down-scaling of the inputs occurs. This also means that the spatial resolution of the hand regions is reduced. We can instead crop out hand images in a pre-processing step and use these as main inputs to the model, preserving more spatial information in the areas of the hands. This also allows the model to recognize hand shape and orientation independently from the position of the hands.

We decide to perform this cropping based on OpenPose keypoint information. Specifically, we use the OpenPose BODY-135 model [14] which estimates keypoints for the body, hands, face and feet. We could crop the hands by computing an axis aligned bounding box around the hand keypoints. However, OpenPose hand keypoints can be noisy in sign language data [10]. Instead, we determine a suitable location for the hand crop in the extension of the forearm: based on the position of the elbow and wrist keypoints, similar to the approach taken by Simon *et al.* [25]. There are zero cases in the training set of AUTSL in which all hand keypoints are predicted while the wrist or elbow keypoints are missing. Clearly calculating the crop on these keypoints is a more robust option than using bounding boxes around the hand keypoints. This is also illustrated in Figure 3 which compares both methods.

We determine the size of the crops in a way that is mostly invariant to the distance between the camera and the person and to the physical appearance of that person. While Simon *et al.* choose the crop dimensions based on statistics from the training set, we base it on the distance between the shoulders.

The square crops are defined as

$$b = (c_x, c_y, s), \tag{5}$$

with $c = (c_x, c_y)$ the center of the crop derived from the

wrist w and elbow e,

$$c = w + 0.15(w - e), \tag{6}$$

and s the size of the crop derived from the distance between the left and right shoulder (l and r),

$$s = 1.2 \left\| l - r \right\|_2 . \tag{7}$$

The factor 0.15 for the extension along the direction of the forearm in Equation 6 is the one proposed by Simon *et al.* [25]. The factor 1.2 for the size of the crop in Equation 7 is empirically obtained by us. Without this scaling factor, in some cases the crops would be slightly too small. This happens for example when the hand is oriented vertically or horizontally, or when the arm is stretched towards the camera. This factor slightly increases the crop size to reduce the number of these edge cases.

In rare cases, shoulder, wrist or elbow keypoints can be missing. If this happens, we obtain a hand crop from a neighboring frame, as temporally close as possible. Should we be unable to find such a replacement hand crop, we use full image inputs.

3.3. Pose flow

Cropping out the hands from the original video frames is an effective way to reduce background noise and increase the spatial resolution of the hands in the inputs. However, we lose information about movements in the sign by doing so. The input signal encodes changes in hand shape and orientation, but not movement.

To reintroduce movement as a form of temporal information, we could add the original full frame as an additional input or compute optical flow. There are, however, several drawbacks to these approaches. In both cases, we require an additional branch of our feature extractor, because the high-level features of hand images and optical flow or full body images are not similar. In fact, we found that sharing parameters to compute features for both hand images and full body images reduced the accuracy of our model. Adding an additional branch, however, increases the amount of trainable parameters by a large amount. Furthermore, calculating (dense) optical flow is computationally intensive.

Instead, we extract a movement encoding analogous to optical flow using OpenPose keypoints. For a selection of K keypoints, we compute the angle and magnitude of the vector given by the difference in positions of a keypoint in two consecutive frames. Doing this for all keypoints results in a feature vector that represents the movements of the body using far fewer dimensions than dense optical flow would. We denote the keypoints of a sample of length L frames as $P \in \mathbb{R}^{L \times K \times 2}$, the keypoints of frame $i \in [1, L]$ as $P^{(i)}$ and the keypoint at index k in frame i as $P_k^{(i)}$. For a frame $i > 1$, the motion vector is

$$\mu_k^{(i)} = P_k^{(i)} - P_k^{(i-1)}. \tag{8}$$

The angle of this motion vector with respect to the horizontal axis is given by

$$\theta_k^{(i)} = \arctan2(y, x), \tag{9}$$

for $(x, y) = \mu_k^{(i)}$ the components of the motion vector. The magnitude of the motion vector is its 2-norm:

$$\rho_k^{(i)} = \left\| \mu_k^{(i)} \right\|_2 . \tag{10}$$

The pose flow for the first frame, $i = 1$, is initialized as zero. We therefore obtain a vector $\left(\theta_k^{(i)}, \rho_k^{(i)}\right)$ for every frame i and keypoint k.

Before computing the pose flow, we perform following pre-processing steps to increase robustness against keypoint estimation errors and changes in camera position. We first replace any keypoint that is missing (*i.e.*, mapped to the origin by OpenPose) by looking for a frame in which it was predicted, while minimizing the distance between the original frame and replacement frame. Afterwards, we normalize each pose frame by dividing the keypoints by the length of the neck to account for the distance between the camera and the subject. This ensures that the magnitude of movements is not dependent on this distance as it would be if it were expressed in pixels. We then finally compute the pose flow on these normalized keypoints.

4. Experiments

This section details the three experiments compared in this work. The experiments have several things in common. One is the approach towards temporal sampling of frames from the input videos. The persons always start with their hands in a neutral position, perform the sign and then return to a neutral position. These frames are irrelevant to the classification task. We select 16 frames in the middle of the video with a temporal stride of 2 frames, for an effective temporal receptive field of 32 frames.

For all our experiments, we use the following hyperparameter settings. We use the Adam optimizer [16] with initial learning rate $\lambda = 1\text{e}{-4}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1\text{e}{-8}$. During training, we decrease the learning rate with a factor 10 every 5 epochs. We train until the validation loss does not decrease for 10 epochs, and select the model at the epoch with the lowest loss as our final model. We use the categorical cross-entropy loss function.

For the 2D feature extractor, we choose ResNet-34 [13] pre-trained on ImageNet [11], extracting a 512-dimensional feature vector per frame. We use 4 layers of 8-head attention to process the resulting sequence in the latent space. The size of the embeddings in the self-attention decoder differs per experiment.

4.1. Video transformer network (VTN)

For the first experiment, we apply the VTN to raw RGB inputs. We apply multi-scale cropping [34], random horizontal flipping (with probability 0.5 per sample) and small random changes to brightness, contrast and saturation as data augmentation. Since there is a single image input per frame, the dimensions of the feature vector inputs to the self-attention decoder are $d_{model} = 512$.

4.2. Hand cropping (VTN-HC)

A drawback of the VTN trained with full frame inputs is that the areas covering the hands have a small spatial resolution in the inputs of the network. For this experiment, we crop out the hands based on the wrist positions extracted with OpenPose as described in Section 3.2. The resulting hand crops are passed through the VTN, with parameter sharing in the CNN. As we obtain one 512-dimensional feature vector per hand, the embedding size in the self-attention decoder d_{model} is 1024.

Compared to the VTN, the VTN-HC has considerably more trainable parameters: 50.9 million compared to 28.8 million. This is due to the increase in the embedding size. While we experimented with dimensionality reduction between the feature extractor and decoder, we obtained the best results by keeping all 1024 features.

4.3. Pose flow (VTN-PF)

From error analysis of the VTN-HC experiment we notice confusions between certain classes. For example, classes 224 and 165 are confused, as well as 51 and 22. These pairs of signs have similar hand shapes but different movements. Indeed, by only using hand crops as inputs the network does not have access to information on the motion of the person. Therefore, we wish to add this information again to the model using pose flow.

We can use the pose keypoints to encode movements of the body in a low-dimensional space. This can be done by calculating motion vectors: differences in coordinates between consecutive frames. As a further feature transformation, we calculate the angle and magnitude of each motion vector and use those as features: see Section 3.3. We normalize the angle by dividing by π radians, such that the features are within the range $(-1, 1]$. The magnitude is already normalized because of the normalization of the pose before the computation of pose flow.

We calculate the pose flow for $K = 53$ keypoints: we use the pose keypoints of the upper body and the hands. We do not use the keypoints of the face for pose flow computation as we empirically found no benefit to including them. Note that this observation may not hold for sign language translation.

This pose flow information, which is encoded as a 106-dimensional vector per frame, is concatenated to the feature vectors extracted by the CNN. The resulting feature vector is normalized to have zero mean and unit variance. We use a non-linear transformation (with ReLU activation) such that the input to the self-attention decoder has the same dimensions as in the VTN-HC model, *i.e.*, $d_{model} = 1024$.

The amount of trainable parameters is slightly higher than for the VTN-HC model (52.1 million), because of the non-linear transformation between the feature extractor and self-attention decoder.

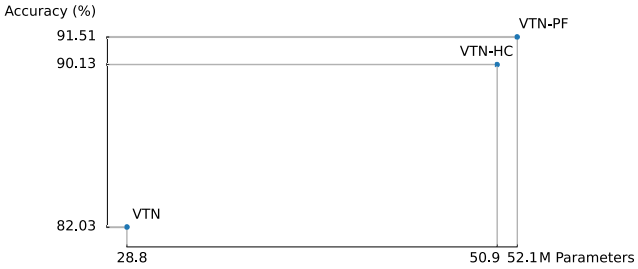

Figure 4. By introducing hand cropping and pose flow into our model, we can increase the accuracy on the validation set. As a side effect, the number of trainable parameters of the model is also increased. Our best model, VTN-PF, achieves 91.51% validation set accuracy.

5. Results and analysis

We compare the three experiments on the validation set of AUTSL. The vanilla VTN, with full frame inputs, has the smallest number of parameters but also obtains the lowest accuracy of our experiments, 82.03%. By cropping out the hands as a pre-processing step, we can increase the accuracy to 90.13% (VTN-HC). This is due to several reasons. Firstly, background noise is reduced and the model can focus entirely on the regions around the hands. Secondly, the spatial resolution of the hands is higher when they are cropped in pre-processing. Finally, the model can learn hand shape and orientation independently from hand position.

This last reason is also a drawback of our hand cropping approach: we remove information about the position (and therefore movement) of the hands. The VTN-PF model obtains this information from pose flow. This reduces errors in cases where two signs are hard to distinguish without movement information, *e.g.*, classes 224 and 165 and classes 51 and 22. Using this model, we can obtain a validation set accuracy of 91.51%.

Figure 4 shows a comparison between the accuracy and the amount of trainable parameters of these three experiments.

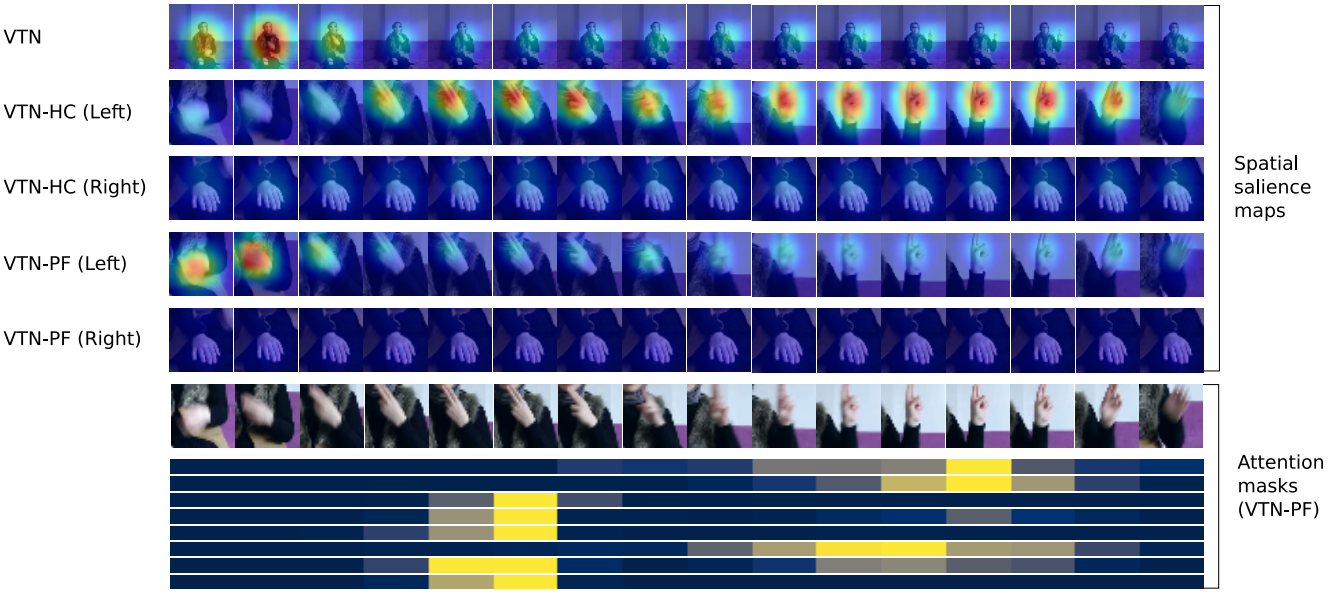

Figure 5. The top five rows in this figure show the salience heatmaps for sample 120 of signer 11 for our three models. Each row is normalized (along the time axis). On the bottom the attention masks are illustrated for all eight heads in the final self-attention layer of the VTN-PF model. These attention masks are averaged to show the amount of attention given to each frame by all frames in the sequence.

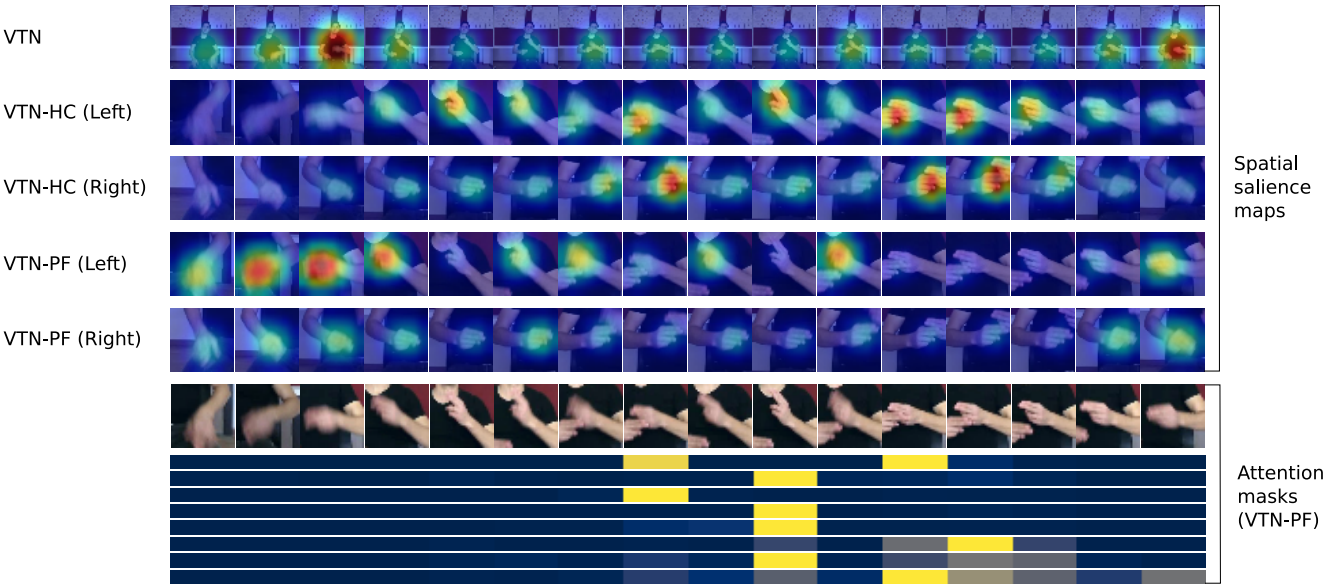

Figure 6. Salience heatmaps and attention masks for sample 57 of signer 1, similar to Figure 5.

5.1. Qualitative analysis of the models

We qualitatively inspect our models by selecting two random samples from the validation set and visualizing the spatial information extraction by the ResNet-34 CNN as well as the attention masks in the self-attention decoder. This inspection is visualized in Figures 5 and 6.

The top five rows of each figure show salience heatmaps, computed from the feature maps of the final layer of the ResNet-34 feature extractor in the VTN, VTN-HC and VTN-PF models. These heatmaps have been normalized along the entire sequence such that we see the most important features both spatially and temporally. For the VTN (first row), we notice that the model focuses on the entire person during the beginning and end frames and on the hand regions during the other frames. The second and third row visualize the salience maps of the VTN-HC model, for the

left and right hand, respectively. We notice that the network focuses on hand shape. Temporally, the VTN-HC model focuses on the frames in which the hand shape is specific for the sign. Our observations are similar for the VTN-PF model, though the specific frames that are focused on are different. Furthermore, because of the added pose flow features, less movement information needs to be encoded in the spatial representation. The non-dominant hand is clearly not important to the models in Figure 5. In a two handed sign (Figure 6), we see activations in both hands.

Below the salience maps, we visualize the attention masks for every head in the final self-attention layer of the VTN-PF model. We take the attention mask and average it along one dimension to visualize the amount every frame is attended to by all frames in the sequence. We notice quite some redundancy between the heads, suggesting that we could prune the network to reduce the model size. We also observe that only a small subset of frames is attended to. This is the subset of frames from which we can determine the sign: the wind-up and wind-down frames are not attended to. The multi-head attention appears to be able to determine the temporal region(s) of interest of signs in this task. We further observe that the attention masks and spatial feature maps do not correspond temporally in the VTN-PF model. This is likely because the attention is being performed to the motion in those frames (encoded by pose flow), while the hand shape is extracted from different frames as it is mostly the same throughout the sequence.

We draw two conclusions based on these visualizations. Firstly, we propose always using cropping as a pre-processing step. In our case, we cropped around the hands, but for sign language translation, we would also crop around the face, like done by Camgoz *et al*. [6]. Secondly, our qualitative analysis suggests that multi-head attention extracts temporal regions of interest from signs. These promising results as well as the results in the sign language translation domain [7, 6, 35] encourage further investigation of the applications and inner workings of multi-head attention in these domains.

5.2. Test set accuracy

This work has described our iterative approach towards the development of our best model, the Video Transformer Network with hand cropping and pose flow (VTN-PF). We evaluate this model on the balanced test set of AUTSL. We achieve an accuracy of 92.92%.

The baseline proposed by Sincan *et al*. [27] obtains a test set accuracy of 49.22%. It is based on a CNN with feature pooling, followed by a bi-directional LSTM (BiLSTM) with temporal attention. We also compare our method to the three winners of the ChaLearn 2021 Looking at People Large Scale Signer Independent Isolated SLR CVPR Challenge [26]. This comparison can be seen in Table 1.

Table 1. Comparison of our method with the three winners of the ChaLearn 2021 Looking at People Large Scale Signer Independent Isolated SLR CVPR Challenge [26] and the baseline.

Method	**Test set accuracy**
SAM-SLR [15]	98.42%
USTC-SLR	97.62%
wenbinwuee	96.55%
VTN-PF (Ours)	92.92%
Baseline [27]	49.22%

The VTN-PF model clearly outperforms the baseline, but other methods are more powerful. Jiang *et al*. (first place) use an ensemble of models trained using optical flow, RGB and keypoint data, which they name Skeleton Aware Multi-modal SLR framework (SAM-SLR) [15]. For future work, it would be interesting to investigate using our model in such an ensemble. At the time of submission, no information on the approaches used by the second (USTC-SLR) and third place entries (wenbinwuee) was available.

6. Conclusion

We apply the VTN architecture, comprising a 2D CNN and self-attention decoder, on the AUTSL isolated sign recognition dataset. We obtain promising initial results with the vanilla VTN, but a simple improvement can increase the classification accuracy by a large amount. Instead of using full frames of the video as input, which include irrelevant information and possibly background noise, we crop out images of the hands as inputs for the network. This yields an increase in accuracy of 8.1%. As a further improvement of 1.4% based on error analysis, we include pose flow inputs. Pose flow is similar to optical flow, but computed on pose coordinates obtained from OpenPose rather than pixels. These pose flow inputs allow our model to better distinguish between signs which have similar hand shapes but different movements. Finally, we perform a qualitative analysis of our model by visualizing and interpreting both spatial salience maps and attention masks in our best performing model. This analysis provides insight into the workings of multi-head attention in a sign language recognition context.

Our final model, the Video Transformer Network with hand cropping and pose flow (VTN-PF), achieves 92.92% accuracy on the balanced test set of AUTSL.

Acknowledgements

Mathieu De Coster's research is funded by the Research Foundation Flanders (FWO Vlaanderen): file number 77410. This research was conducted under the SignON project, funded by the European Union's Horizon 2020 Programme, with Grant Agreement No. 101017255.

References

[1] Samuel Albanie, Gül Varol, Liliane Momeni, Triantafyllos Afouras, Joon Son Chung, Neil Fox, and Andrew Zisserman. BSL-1K: Scaling up co-articulated sign language recognition using mouthing cues. In *European Conference on Computer Vision*, 2020.

[2] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[3] Richard Bank, Onno A Crasborn, and Roeland Van Hout. Variation in mouth actions with manual signs in Sign Language of the Netherlands (NGT). *Sign Language & Linguistics*, 14(2):248–270, 2011.

[4] Danielle Bragg, Oscar Koller, Mary Bellard, Larwan Berke, Patrick Boudreault, Annelies Braffort, Naomi Caselli, Matt Huenerfauth, Hernisa Kacorri, Tessa Verhoef, et al. Sign language recognition, generation, and translation: An interdisciplinary perspective. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*, pages 16–31, 2019.

[5] Necati Cihan Camgoz, Simon Hadfield, Oscar Koller, and Richard Bowden. Subunets: End-to-end hand shape and continuous sign language recognition. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3075–3084. IEEE, 2017.

[6] Necati Cihan Camgoz, Oscar Koller, Simon Hadfield, and Richard Bowden. Multi-channel transformers for multi-articulatory sign language translation. In *European Conference on Computer Vision*, pages 301–319. Springer, 2020.

[7] 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/CVF Conference on Computer Vision and Pattern Recognition*, pages 10023–10033, 2020.

[8] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):172–186, 2019.

[9] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[10] Mathieu De Coster, Mieke Van Herreweghe, and Joni Dambre. Sign language recognition with transformer networks. In *12th International Conference on Language Resources and Evaluation*, 2020.

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[12] Kirsti Grobel and Marcell Assan. Isolated sign language recognition using hidden Markov models. In *1997 IEEE International Conference on Systems, Man, and Cybernetics. Computational Cybernetics and Simulation*, volume 1, pages 162–167. IEEE, 1997.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[14] Gines Hidalgo, Yaadhav Raaj, Haroon Idrees, Donglai Xiang, Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Single-network whole-body pose estimation. In *ICCV*, 2019.

[15] Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li, and Yun Fu. Skeleton aware multi-modal sign language recognition. *arXiv preprint arXiv:2103.08833*, 2021.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[17] Sang-Ki Ko, Chang Jo Kim, Hyedong Jung, and Choongsang Cho. Neural sign language translation based on human keypoint estimation. *Applied Sciences*, 9(13):2683, 2019.

[18] Sang-Ki Ko, Jae Gi Son, and Hyedong Jung. Sign language recognition with recurrent neural network using human keypoint detection. In *Proceedings of the 2018 Conference on Research in Adaptive and Convergent Systems*, pages 326–328, 2018.

[19] Oscar Koller. Quantitative survey of the state of the art in sign language recognition. *arXiv preprint arXiv:2008.09918*, 2020.

[20] Alexander Kozlov, Vadim Andronov, and Yana Gritsenko. Lightweight network architecture for real-time action recognition. In *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, pages 2074–2080, 2020.

[21] Dongxu Li, Cristian Rodriguez, Xin Yu, and Hongdong Li. Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[22] Irene Murtagh. Towards a linguistically motivated Irish Sign Language conversational avatar. *The ITB Journal*, 12(1):4, 2011.

[23] Lionel Pigou, Mieke Van Herreweghe, and Joni Dambre. Sign classification in sign language corpora with deep neural networks. In *International Conference on Language Resources and Evaluation (LREC), Workshop, Proceedings*, pages 175–178, 2016.

[24] Adam Schembri, Jordan Fenlon, Ramas Rentelis, Sally Reynolds, and Kearsy Cormier. Building the British sign language corpus. *Language Documentation & Conservation*, 7:136–154, 2013.

[25] Tomas Simon, Hanbyul Joo, Iain Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1145–1153, 2017.

[26] Ozge Mercanoglu Sincan, Julio C. S. Jacques Junior, Sergio Escalera, and Hacer Yalim Keles. Chalearn LAP large scale signer independent isolated sign language recognition challenge: Design, results and future research. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2021.

[27] Ozge Mercanoglu Sincan and Hacer Yalim Keles. AUTSL: A Large Scale Multi-Modal Turkish Sign Language Dataset

and Baseline Methods. *IEEE Access*, 8:181340–181355, 2020.

[28] William Stokoe. Jt.(1960). sign language structure: An outline of the visual communication systems of the american deaf. *Studies in Linguistics, Occasional Papers*, 8, 1960.

[29] Shinichi Tamura and Shingo Kawasaki. Recognition of sign language motion images. *Pattern recognition*, 21(4):343–353, 1988.

[30] Hamid Vaezi Joze and Oscar Koller. MS-ASL: A large-scale data set and benchmark for understanding American Sign Language. In *The British Machine Vision Conference (BMVC)*, September 2019.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[32] Myriam Vermeerbergen, Lorraine Leeson, and Onno Alex Crasborn. *Simultaneity in signed languages: Form and function*, volume 281. John Benjamins Publishing, 2007.

[33] Christian Vogler and Dimitris Metaxas. Handshapes and movements: Multiple-channel american sign language recognition. In *International Gesture Workshop*, pages 247–258. Springer, 2003.

[34] Limin Wang, Yuanjun Xiong, Zhe Wang, and Yu Qiao. Towards good practices for very deep two-stream ConvNets. *arXiv preprint arXiv:1507.02159*, 2015.

[35] Kayo Yin and Jesse Read. Better sign language translation with STMC-transformer. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5975–5989, 2020.