September 2023

USING NEURAL MACHINE TRANSLATION TO TRANSLATE BETWEEN DIFFERENT SIGN LANGUAGE FORMS DURING A VIDEO CONFERENCE

Anusha Kopparam

Ananthi Jairaj

Vinay Kumar Abburi

Donald M Allen, PhD

USING NEURAL MACHINE TRANSLATION TO TRANSLATE BETWEEN DIFFERENT SIGN LANGUAGE FORMS DURING A VIDEO CONFERENCE

AUTHORS:
Anusha Kopparam
Ananthi Jairaj
Vinay Kumar Abburi
Donald M Allen, PhD

ABSTRACT

Techniques are presented herein that support the efficient conversion of signs from one form of sign language to another while considering the cultural context (e.g., dialect, etc.) of a sign language. Aspects of the presented techniques support the conversion between different sign language forms through a neural machine translation (NMT)-based architecture. Further aspects of the techniques may encompass a contextual frame sampler (which may employ a sign language image database to filter out noise and which may sample frames from an input sign language video), an image normalizer (that may accept as input a sampled image frame and produce as output a skeletal structure of that frame), a translation layer (which may contain a NMT-based model and which may comprise feature extraction, feature conversion, and feature generation capabilities), and a video generator (which may stitch together the generated translated sign language output frames into a video). Under still further aspects of the techniques, such a conversion capability may be available during a video conference.

DETAILED DESCRIPTION

The process of learning sign language is similar to that of learning to speak a new language. However, ‘sign language’ is not one universal language. Currently, there are over 300 different forms of sign language in use.

Like a spoken language, a person typically learns the particular form of sign language that is dominant in their environment. One example is American Sign Language (ASL), several samples of which are presented in Figure 1, below.

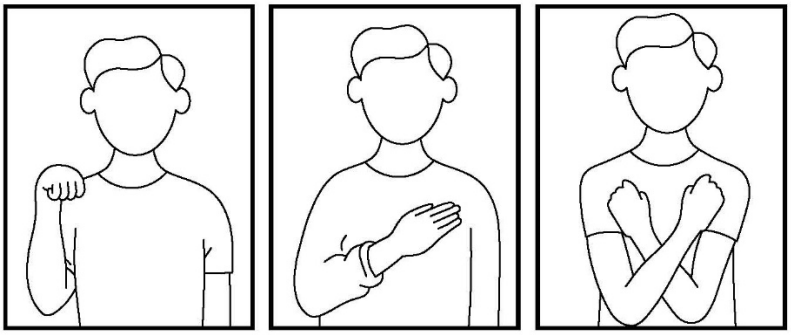

Figure 1: Sample Signs in ASL

However, a person who understands and uses ASL may not be able to understand or sign in British Sign Language (BSL), even though the associated spoken and written language are both based on the English language.

The differences between sign language forms are greater when those forms do not share the same base language. As a result, signers who sign using different languages cannot understand each other using their primary method of communication (i.e., signing). For example, during a meeting such individuals will either need to rely on captions, which may distract them from the main content of the meeting, or have a person translate from one sign language to another to facilitate communication between the two signers. However, it is frequently difficult to find a sign language interpreter and finding a person who knows more than one sign language is problematic.

The above-described challenge is particularly apparent during a video conference. Video conferencing is a powerful tool that can help people collaborate and communicate more effectively. However, current video conferencing systems are not fully accessible to deaf people, thus making a deaf person's participation in a video conference a challenge. This is because most video conferencing software does not provide real-time captioning or a translation of sign language. As a result, deaf people can miss important information or be unable to participate in conversations.

Before presenting a detailed discussion of the techniques presented herein, which address the above-described challenge, it will be helpful to briefly review a number of existing solutions (that attempt, but fail, to address elements of that challenge), aspects of which are leveraged in unique ways by the presented techniques.

First, for more than half a decade, neural machine translation (NMT) techniques have been used in natural language processing settings for the automated translation of language. Importantly, the encode-decoder model of such a facility learns a correspondence between an input and an output language to effectively translate text from one language to another.

Second, with computer vision gesture recognition has been applied to solve the problem of translating between a sign language and spoken or written languages. One such approach is Sign Language Recognition (SLR), where the approach recognizes a sequence of continuous signs but fails to understand the rich grammar and sentence structure of a sign language (since the approach translates one sign at a time to a spoken or written language). Such an approach affects the quality of a translation and does not produce any meaningful interpretation of what a signer is conveying.

Third, sign language translation (SLT) has recently been proposed to address the above-described problem. This approach converts a video containing a series of sign language images (which may be referred to herein as a sign video) to an equivalent spoken language sentence. The first proposal of SLT encompassed NMT techniques to learn a correspondence between the grammar of a signed language and a spoken or written language.

All of the above-described solutions are designed to translate between a sign video and written or spoken language sentences. However, none of the solutions discuss translating directly between sign languages. Also, while an NMT-based architecture has been employed in the language domain, there is no research applying the same to images. The presented techniques, which will be described and illustrated below, support adapting an NMT-based architecture to images to effectively learn an image vocabulary in support of a translation of signs between sign language forms.

And fourth, a prefix tree is a data structure that may be used to store strings. Each node in the tree represents a character, and the children of a node represent the possible next characters in a string. The leaves of such a tree represent strings that have been completely matched. Prefix trees have long been applied for text; the presented techniques support a new way of using such a data structure to efficiently store image ‘chunks’ (i.e., image words) for memory savings and a faster lookup.

Turning now to the techniques presented herein, those techniques support translating directly between different sign language forms during a virtual meeting, enabling signers using any language to converse and communicate with each other with no external dependencies. Through the use of the presented techniques, a video conferencing system may be made more inclusive for deaf people so that such an individual may collaborate effectively no matter what sign language they employ.

The presented techniques comprise a number of novel elements. A first element encompasses a new way of storing an image vocabulary (i.e., image chunks) using a prefix tree. A second element encompasses training a sign language translation model on each pair of a set of sign language videos (e.g., one video going from ASL to BSL and another video going from BSL to ASL), the results of which may be used to generate a translated video in near real-time using a prefix tree with no intermediate step.

A third element encompass the first application of a NMT mechanism to images. In brief, input image frames may be chunked and then passed through such a NMT model to generate output image frames one chunk at a time. Such an approach significantly improves translation performance and reduces the memory requirements for storing frames from high-resolution sign videos.

As will be described in detail below, the presented techniques encompass an architecture that comprises a series of interconnected modules. Figure 2, below, presents elements of an exemplary system (highlighting various of those modules) that is possible according to the presented techniques and which is reflective of the above discussion.

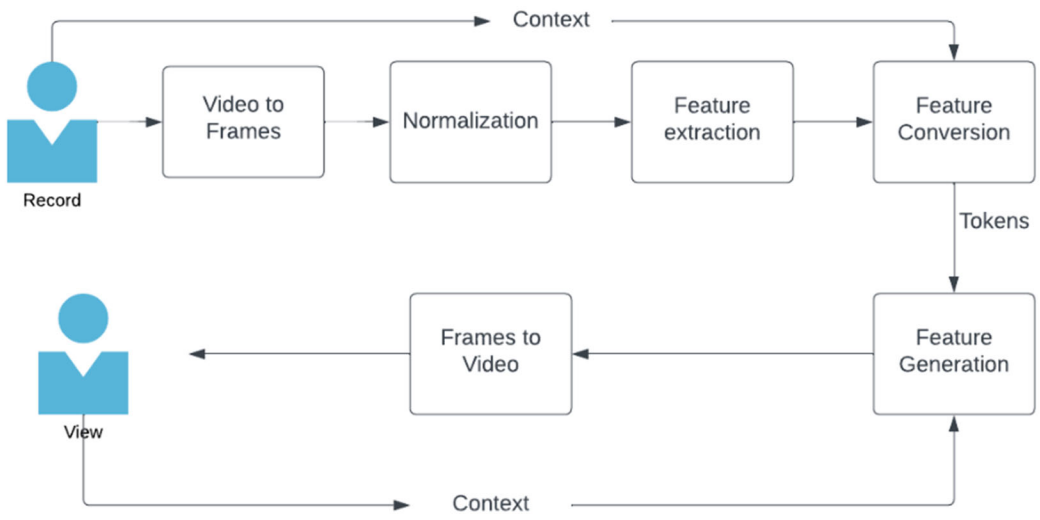

Figure 2: Exemplary System

Aspects of the presented techniques may convert a given sign language into an intermediate sign language (such as, for example, International Sign) by considering metadata such as a dialect. Further aspects of the techniques encompass a lightweight model, a minimization of bandwidth and memory consumption, and a minimization of model retraining effort.

As shown in Figure 2, above, a system according to the presented techniques may consist of a series of interconnected modules that translate a sign video from one sign language to another.

A first module encompasses a contextual frame sampler (CFS) which may employ a sign language image database to filter out noise and which may sample frames from an input sign language video. A second module encompasses an image normalizer that may accept as input a sampled image frame (as described above with the first module) and produce as output a skeletal structure of that frame.

A third module encompasses a translation layer, which may comprise feature extraction, feature conversion, and feature generation capabilities. This layer contains an NMT-based model whose encoder and decoder may be trained on different sign language videos. Once such training is complete, only the decoder is then used for inferencing or actual translation.

A fourth module encompasses a video generator which may stitch together the generated translated sign language output frames into a video.

Turning to the first module (which, as described previously, encompasses a CFS), this module employs the dialect that is being signed as the context to sample the frames from the input video stream. Such an activity involves taking each frame of the video as an input and, using the dialect, calculating the probability that the frame contains a sensible sign. A sign language image database may be used as a source of truth to filter out noise. The above-described model may be trained by maximizing the logarithmic likelihood of correctly predicting the output frame, measured as a frame having a sensible sign given the input image frame from the image database and the dialect.

Mathematically, the above-described process may be represented by the equation that is shown in Equation 1, below:

$$\max \log\left(P\left(\frac{Y}{X,d}\right)\right)$$

$P\left(\frac{Y}{X,d}\right)$ is likelihood of output image Y having a sensible sign given input image frame X and dialect d.

Equation 1: Training Objective for CFS

and visually, the above-described process may be expressed through the arrangement that is presented in Figure 3, below.

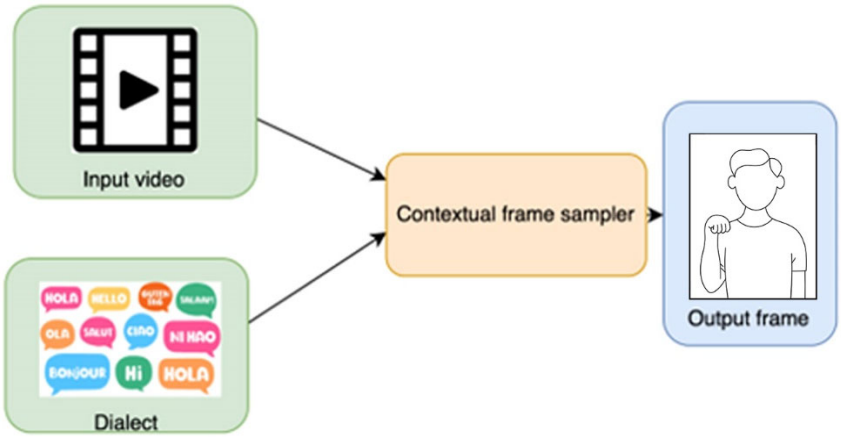

Figure 3: Exemplary CFS

The second module within the presented techniques (which, as described previously, encompasses an image normalizer) improves the accuracy of a translation from

one sign language form to another. During the generation of frames in a requested sign language, it is important that the input frames be of a superior quality. The quality of an input frame may be poor for any number of reasons, including a signer having a busy background, low bandwidth resulting in a low-resolution image, etc. Figure 4, below, presents an example of a low-quality image frame.

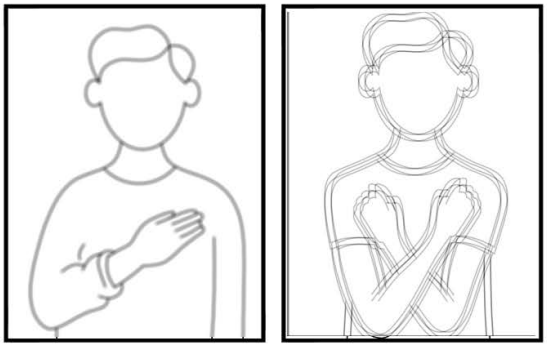

Figure 4: Exemplary Poor Image

To address the above-described problem, under the presented techniques an input image may be normalized to a skeletal image. The concept of body pose detection may be combined with a state-of-the-art model to convert the input image into a skeletal image. Such a skeletal image may be of the same size and height as the person in the input frame. Since each person may sign in a slightly different (e.g., personalized) fashion, to remove any anomalies or deviation in such signs a normalizer may map a signer's skeletal image to a pre-exiting skeletal image in an image database. This ensures that all of the frames containing the same signs look the same. Figure 5, below, depicts elements of such a process where a sample input image is converted into a skeletal image using body pose detection.

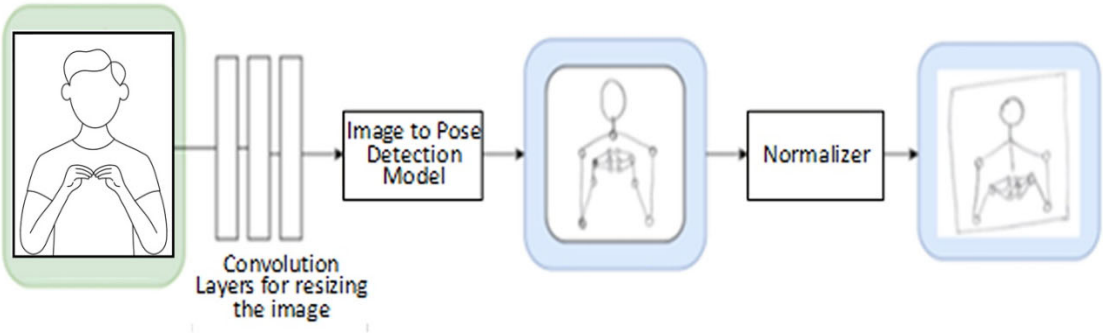

Figure 5: Exemplary Skeletal Image

Turning next to the translation layer (which, as described previously, contain an NMT-based model and which may comprise feature extraction, feature conversion, and feature generation capabilities), an initial activity encompasses a training phase of the model (e.g., an NMT-based architecture) for learning a correspondence between input and output image frames.

As an initial matter, it is important to note that when one language is translated to another, a simple word-to-word translation is not effective. For example, Figure 6, below, depicts a word-to-word correspondence between English and German for a candidate sentence.

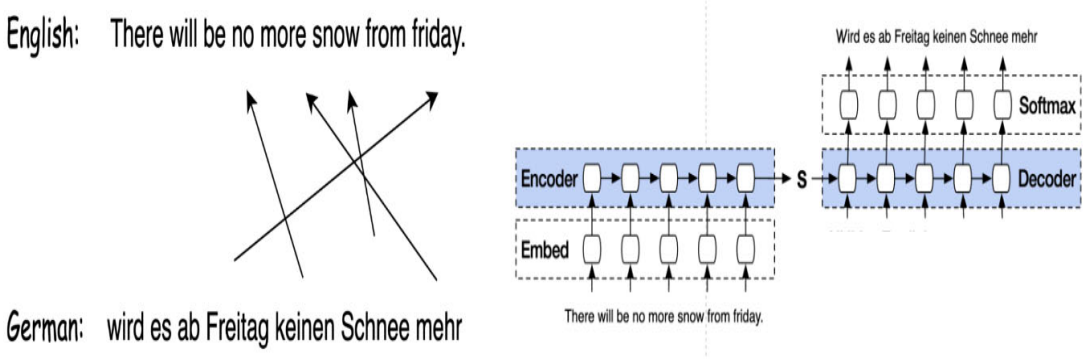

Figure 6: Word-to-Word Correspondence

As shown in Figure 6, above, words that appear at the beginning of the sentence in German are translated or mapped to words that appear at the end in the corresponding (translated) English sentence.

In contrast to the above-described approach, an NMT-based model learns to map words from one language to another in their correct position. The presented techniques

leverage such a capability of an NMT-based model and adapt such a model to work with images.

During a training phase, two sign video clips (Xv and Yv) conveying the same message may be employed. The sign video clip Xv contains signs in the input sign language while the sign video clip Yv contains signs in the output sign language. Using a CFS (as described above), image frames from both Xv and Yv may be sampled and then normalized.

Then, a normalized input and output frame [X, Y], of size h x w, may be sliced or chunked into N image patches, each having a size n x n, where N is given by the equation that is presented in Equation 2, below.

$$N = \frac{h \cdot w}{n \cdot n}$$

Equation 2: Input Chunking Mechanism

Next, the chunked input frames may be converted to embeddings and then passed through the encoder model to generate a latent vector which may be used as an input to the decoder. The decoder may then generate chunks of output frames, which may be used to calculate an error by applying a cross entropy loss for each chunk of an output frame. Such a loss may be back-propagated to teach the model the correct correspondence between input image patches or chunks and output image patches or chunks.

It is important to note that during the initial iterations of a training cycle, the model may generate random patches or chunks resulting in a higher loss value.

Figure 7, below, presents elements of an architecture of an NMT-based model (as applied to images) that is possible according to the presented techniques and which is reflective of the above discussion.

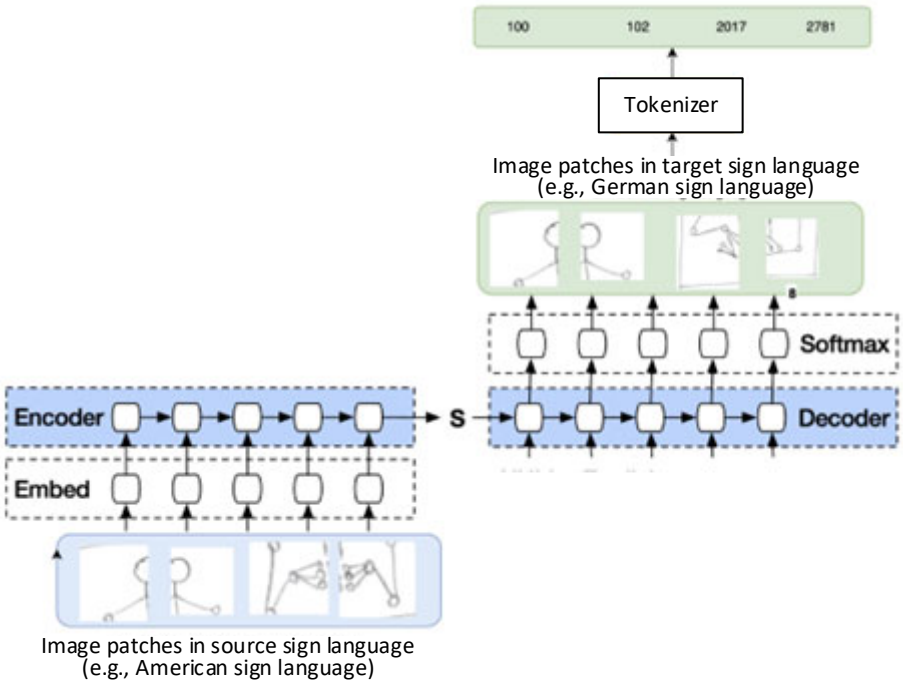

Figure 7: Architecture of NMT-based Model

After the above-described model is trained, it may then be used to translate one form of sign language to another, as explained below in connection with an inference phase. In support of that explanation, Figure 8, below, presents elements of an exemplary inference engine architecture.

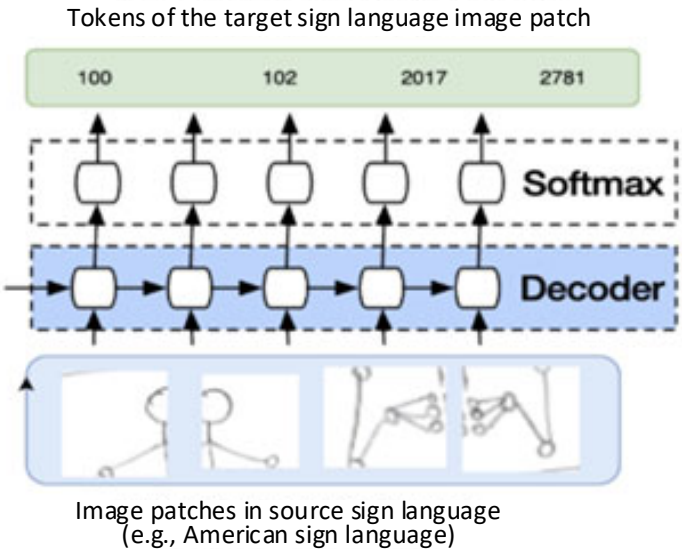

Figure 8: Inference Engine Architecture

An inference phase leverages a trained decoder (as previously described) and the architecture that is depicted in Figure 8, above. During an inferencing process, from a live video feed of a user a CFS selects the specific image frames that contain meaningful source sign language signs that are to be translated; through an image normalizer each image frame is converted to a normalized skeletal image of the user signing; the skeletal image is chunked according to Equation 2, above; and the source skeletal image chunks are fed into the decoder which outputs the appropriate tokens. The tokens may then be used to perform a lookup operation into a prefix tree of images to construct the final output frame which contains the sign in the requested target sign language.

It is important to note that all of the images in the previously described sign language image database may be converted to image patches of a predetermined size using Equation 2, above. Further, an image dictionary may be created with some number of image patches and an index value assigned to each image patch. Figure 9, below, presents a tokenized view of an exemplary image dictionary.

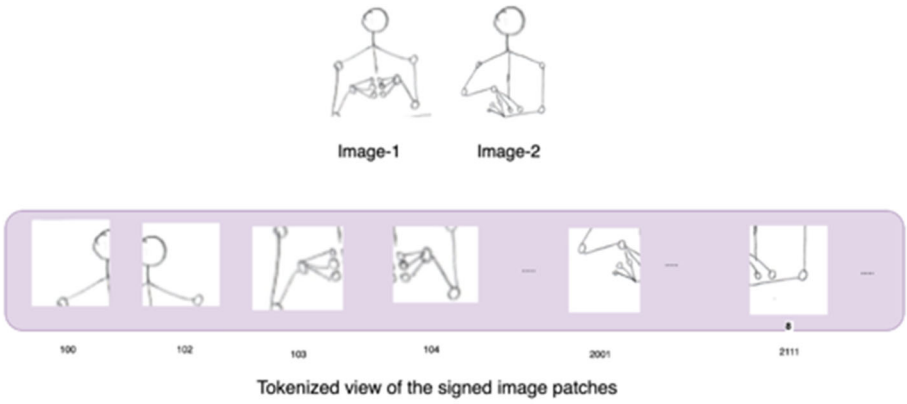

Figure 9: Exemplary Image Dictionary

As described previously, under the presented techniques a prefix tree may be employed for images. In support of the generation of such a tree, an input image may be split into some number of image patches and those patches may then be tokenized using a tokenizer. Both the image patches and the tokens may serve as input to a prefix tree building algorithm. Such an algorithm may output a prefix tree with each node of the tree storing an image patch and its token.

The above-described algorithm may begin with a root node R and then repeat, in a loop, a processing activity until all of the images (X_1, X_2, X_3, ... X_n) in the image dictionary

are covered. During such processing each image may be converted into some number of image patches and each of the image patches may be encoded into vectors (represented by x_1, x_2, x_3, …. x_n) and then tokenized. Each of the image vectors (x_1, x_2, x_3, …. x_n) may then be processed and added to the tree.

During the above-described activity a similarity score may be computed between an image vector x_i and the image patches that are stored in the children of the current root R. If an image patch is found with a similarity score that is greater than a threshold T, processing may move on to the next image patch x_{i+1}. Otherwise, a new child node may be added to the current root R (to store the image patch x_i and its token) and the current root R may be updated to x_i.

Figure 10, below, depicts elements of the construction of a prefix tree, according to the above-described algorithm, for exemplary images.

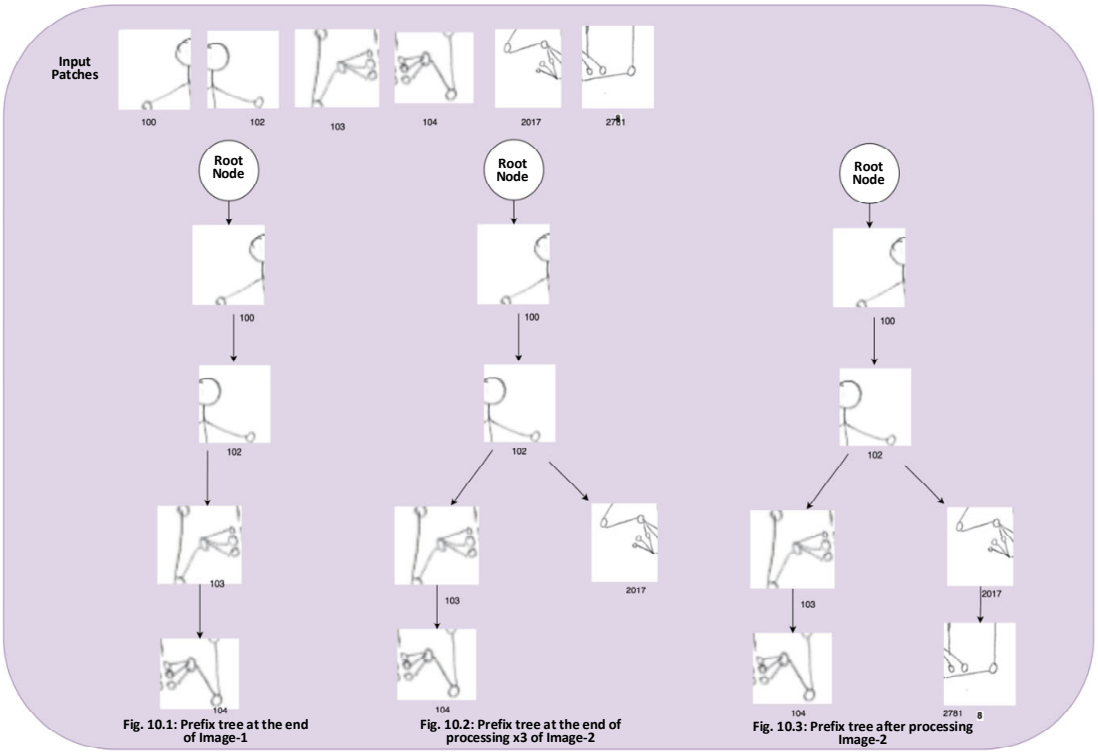

Figure 10: Prefix Tree Construction of Images

The fourth, and final, module within the presented techniques (which, as described previously, encompasses video generation) assembles the above-described chunks to create the output skeletal frames having the translated sign. Using those skeletal image frames, a

video may be generated using known “do as I do” motion transfer mechanisms. Using an input signer’s image as a base style image, the style transfer paradigm under such a mechanism’s generative adversarial network (GAN) may be applied to the target to generate a skeletal image. As a result of such a transfer, the generated skeletal image may be converted into a photo-realistic image of the input signer signing in the target sign language. Figure 11, below, depicts elements of such a video generation process.

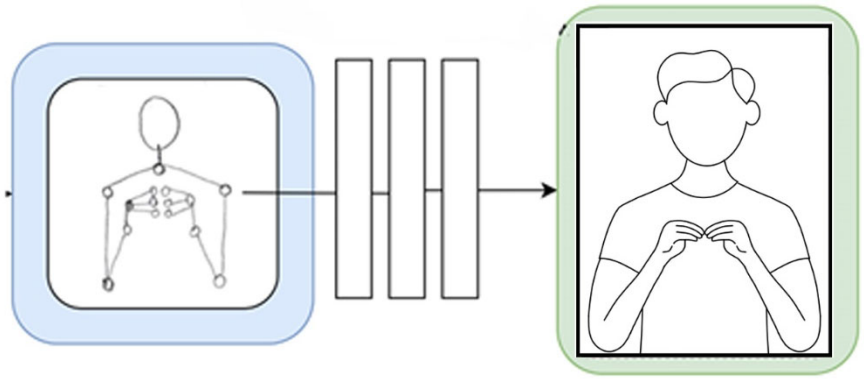

Figure 11: Video Generation

In summary, techniques have been presented herein that support the efficient conversion of signs from one form of sign language to another while considering the cultural context (e.g., dialect, etc.) of a sign language. Aspects of the presented techniques support the conversion between different sign language forms through an NMT-based architecture. Further aspects of the techniques may encompass a contextual frame sampler (which may employ a sign language image database to filter out noise and which may sample frames from an input sign language video), an image normalizer (that may accept as input a sampled image frame and produce as output a skeletal structure of that frame), a translation layer (which may contain a NMT-based model and which may comprise feature extraction, feature conversion, and feature generation capabilities), and a video generator (which may stitch together the generated translated sign language output frames into a video). Under still further aspects of the techniques, such a conversion capability may be available during a video conference.