DHQ: Digital Humanities Quarterly
2021
Volume 15 Number 1

Towards a User-Friendly Tool for Automated Sign Annotation: Identification and Annotation of Time Slots, Number of Hands, and Handshape

Manolis Fragkiadakis <m_dot_fragkiadakis_at_hum_dot_leidenuniv_dot_nl>, Leiden University
Victoria Nyst <v_dot_a_dot_s_dot_nyst_at_hum_dot_leidenuniv_dot_nl>, Leiden University
Peter van der Putten <p_dot_w_dot_h_dot_van_dot_der_dot_putten_at_liacs_dot_leidenuniv_dot_nl>, Leiden University

Abstract

The annotation process of sign language corpora in terms of glosses, is a highly labor-intensive task, but a condition for a reliable quantitative analysis. During the annotation process the researcher typically defines the precise time slot in which a sign occurs and then enters the appropriate gloss for the sign. The aim of this project is to develop a set of tools to assist the annotation of the signs and their formal features in a video irrespectively of its content and quality. Recent advances in the field of deep learning have led to the development of accurate and fast pose estimation frameworks. In this study, such a framework (namely OpenPose) has been used to develop three different methods and tools to facilitate the annotation process. The first tool estimates the span of a sign sequence and creates empty slots in an annotation file. The second tool detects whether a sign is one- or two-handed. The last tool recognizes the different handshapes presented in a video sample. All tools can be easily re-trained to fit the needs of the researcher.

Introduction

While the majority of the studies in the field of digital humanities have been mostly text oriented, the evolution in computing power and technology has resulted in a shift towards multimedia-oriented studies. Recently, advances in computer vision have started to find practical applications in study domains outside of computer and data science. Video is one of the most important time-based media as it has the ability to carry large amount of digital information in a condensed form, and hence it serves as a rich medium to capture various forms of cultural expression. Automated processing and annotation of large numbers of videos is now becoming feasible due to the evolution of computer vision and machine learning. 1

In sign language linguistics, a transition took place from paper-based materials to large video corpora to facilitate the study of the languages in question. Sign language corpora are mainly composed of video data. The primary goal of these video corpora is to study sign language functioning. 2

The processing of sign languages usually involves requires a form of textual representation [Dreuw and Ney 2008], most notably glosses for annotation. Sign language glosses are words from a spoken language. Uniquely identifying glosses by definition refer to a specific sign. Such ID glosses are an essential element for the quantitative analysis of a sign language corpus [Johnston 2010]. Typically, sign language linguists add glosses and other annotations to the video recordings with the use of a software tool (namely ELAN). ELAN allows researchers to add time-aligned annotations to a video. However, this task requires a lot of time and can be prone to errors. 3

New advances in computer vision open up additional ways of studying videos containing sign language data, extracting formal representations of linguistic phenomena, and implementing these in computer applications, such as automatic recognition, generation, and translation. Using computer vision and machine learning enables quick and new ways of 4

processing large sets of video data, which in turns makes it possible to address research questions that were not feasible before.

This study is the first part of a project aiming at the creation of tools to automatize part of the annotation process of sign language video data. This paper presents the methodologies, tools and implementations of three functionalities: the detection of 1) manual activation 2) the number of hands involved and 3) the handshape distribution on sign language corpora. 5

Recent developments in sign language recognition illustrate the advantages of machine and deep learning for tasks related to recognition and classification [Agha et al. 2018] [Cao et al. 2017] [Pigou et al. 2015]. Nevertheless, current approaches are restricted in various ways, limiting their applicability in current sign language research. For example, training deep learning networks requires a vast amount of data as well as adequate computational power. These networks are usually trained in one sign language and they do not generalize well in other sign languages. 6

Additionally, current approaches in sign language automatic annotation need manual annotation of the hands and body joints for the training of the recognizer models [Aitpayev et al. 2016] [Dreuw and Ney 2008]. Moreover, the application of color and motion detection algorithms [Kumer 2017], as feature extraction methods, can be susceptible to errors and possibly skin color bias. Finally, several hand tracking models only work on a particular type of recordings, for example, a signer wearing gloves, or recordings made with Microsoft's Kinect [Pigou et al. 2015]. As a result, these models are not usable for the majority of the existing sign language corpora which have been recorded with standard RGB cameras. 7

Our methods have been developed and tested on two West African sign language corpora containing natural conditions with non-Caucasian signers. While most studies in the sign language recognition field have mainly concerned signers with light skin tones, little research has been conducted using darker skin tones. With the emergence of corpora compiled in African countries under challenging real-world conditions, and their contribution to the overall sign language community, it is of utmost importance to test how methods perform in such a domain. Alleviating biases and increasing diversity should be a top priority of any computer assisted study. 8

In this study, a pre-trained deep learning pose estimation library developed by Cao et al. [Cao et al. 2017] named OpenPose has been used to extract body and finger key-points. OpenPose has been trained and evaluated on two large datasets for multi-person pose estimation. The MPII human multi-person dataset [Andriluka et al. 2014] and the COCO 2016 keypoints challenge dataset [Lin 2014] contain images of people of different age groups and ethnicities in diverse scenarios. As a result, OpenPose does not have a bias toward skin color. Additionally, its easy-to-use implementation makes it an ideal framework to be used by linguists with limited coding experience. 9

The combination of the aforementioned pose estimation framework as well as the machine and deep learning architectures tested in this study, provides a robust approach towards automatic annotation. Current models and tools can be used in any sign language or gestural corpus independently of its quality, length and number of people in the video. These tools have been developed as python modules that can run automatically in a video and produce the relevant annotation files requiring minimal effort from the user. More generally, as large parts of our cultures nowadays are captured in video, our study serves as a case example of how intelligent machine learning techniques can serve digital humanities researchers by extracting semantics from large video collections. 10

This article is structured as follows: Section 2 introduces the developments on the sign language recognition and automatic annotation fields. Section 3 describes the materials used in this study and the methodologies developed and applied for each tool separately. Section 4 presents the results for each experimental setup and tool. Section 5 contains the discussion and future work while Section 6 presents our conclusions. Finally, Appendix A presents the architecture and technical details of the Long-Short-Term-Memory Network trained for this study. 11

2. Literature review

In this section we present the studies conducted on the sign language recognition and automatic annotation field 12

developed with depth sensors as well as standard RGB cameras. Additionally, we describe the developments of the human pose estimation field and we introduce the OpenPose framework that will be used in this article.

2.1 Sign Language Recognition and Automatic Annotation

The primary goal of sign language recognition is to develop methods and algorithms to accurately identify a series of produced signs and to discern their meaning. The majority of studies have focused on recognizing those features and methods that can properly identify a sign out of a given set of possible signs. However, such methods can only be used on a particular set of signs and, thus, a specific sign language, which makes it harder to study the relationships between and evolution of various sign languages. 13

An additional motivation behind Sign Language Recognition (SLR) is to build automatic sign language to speech or text translation systems to assist the communication between the deaf and hearing community [Fang et al. 2004]. Moreover, SLR plays an important role in developing gesture-based human–computer interaction systems [Kelly et al. 2009]. Sign language linguists have used such systems to facilitate the annotation process of sign language corpora in order to discern the different signs in a video recording and further study the linguistic phenomena presented. 14

There are numerous studies dealing with the automated recognition of sign languages as clearly presented by Cooper et al. [Cooper and Bowden 2007], in their review study on the state-of-the-art in sign language recognition. However, the experiments presented in most of these studies are either hard to replicate, or they pose limitations as far as their applicability is concerned. For instance, most of these studies use depth sensors, most notably MS Kinect, to capture 3D images of the environment [Aitpayev et al. 2016] [Pigou et al. 2015] [Zhang et al. 2013]. As a result, using the frameworks developed in these studies requires a machine with similar features as the one used for testing and most probably will only work for that sign language on which they have been trained. 15

Recently, computer vision techniques have been applied to sign language recognition to overcome the aforementioned limitations. Roussos et al. [Roussos et al. 2012] created a skin color probabilistic model to detect and track the hands of a signer on a video, while Cooper et al. [Cooper and Bowden 2007] use this model to segment the hands and apply a classifier based on Markov models. However, systems based on skin color [Buehler et al. 2009] [Cooper and Bowden 2007] [Farhadi et al. 2007] [Starner et al. 1998] are prone to errors and have difficulties on tracking the hands and the signer's features against cluttered backgrounds and in noisy conditions in general. Also, they do not work in videos with multiple signers. 16

2.2 Human Pose Estimation

Human pose estimation has been extensively studied due to its numerous applications on a number of different fields [Moselund et al. 2006]. Due to low computational complexity during inference, pictorial structures have been commonly used [Felzenszwalb and Huttenlocher 2005] [Ramanan et al. 2007] [Sivic et al. 2006] to model human pose. Recently, studies have focused on improving the appearance models used in these structures by modelling the individual body parts [Eichner and Ferrari 2009] [Eichner et al. 2012] [Johnson and Everingham 2009] [Sapp et al. 2010]. Felzenszwalb and Huttenlocher [Felzenszwalb and Huttenlocher 2005], relying on the pictorial structure framework recommended a deformable part-based model. Additionally, Yang and Ramanan [Yang and Ramanan 2011] showed that a tree-structured model using a combination of deformable parts can be used in order to achieve accurate pose estimation. Furthermore, Charles et al. [Charles et al. 2014] showed that human body joint positions can be predicted using a random forest regressor based on a co-segmentation process over all video frames. 17

In general, most of the vision-based approaches developed for sign language recognition tasks utilizing pose estimation, have used the RWTH-PHOENIX-Weather data set [Forster 2012] to validate their models. This data set consists of weather forecast airings from the German public tv-station PHOENIX along with transcribed gloss annotations. However, it is a question to what extent such systems tested in this data set can be replicated with real-life conditions in the corpora. It is often the case that sign language and gestural corpora, especially the ones filmed outside of studio conditions, have bad quality, low brightness and often contain more than one person in the frame. These characteristics create an additional challenge to the tracking and prediction task. 18

2.2.1 OpenPose

OpenPose is a real-time, open-source library for academic purposes for multi-person 2D pose estimation. It can detect body, foot, hand and facial keypoints [Cao et al. 2017]. Following a bottom-up approach (from an entire image as input to full body poses as output), it outperforms similar 2D body pose estimation libraries. 19

A major advantage of the library is that it achieves high accuracy and performance regardless of the number of people in the image. Its high accuracy is performed by using a non-parametric representation of 2D vector fields. These fields encode the position and orientation of body parts over the image domain and their degree of association in order to learn to relate them to each individual. 20

OpenPose is able to run on different operating systems and multiple hardware architectures. Additionally, it provides tools for visualization and output file generation. The output can be multiple json files containing all the pixel x, y coordinates of the body, hand and face joints. In this study the DEMO version on a CPU-only mode has been used to train our models. This choice was made in order to ensure that reproducibility can be easily achieved without the need for powerful computers from the linguist's side. 21

3. Materials and Methods

This section describes the datasets used in our study as well as the pre-processing stage using OpenPose to extract the body joints' pixel coordinates. Furthermore, we introduce the methods applied in the development of each tool. Special consideration is given on the handshape recognition module as an additional normalization part has been developed. 22

3.1 Data

A data set of 7,805 frames in total (approximately 4 minutes) labeled as signing or not signing has been compiled for the first part of the study. The dimensions of the frames were 352 by 288 pixels and were extracted from the Adamorobe and Berbey Sign Language corpora [Nyst 2012] [Nyst et al. 2012]. These corpora portray an additional challenge as the signers have been filmed in and around their homes, in natural conditions, outside of a studio, with strongly varying brightness and background noise. Furthermore, they may contain signing from one and two people at the same time. As a result, they can be considered as one of the hardest corpora to perform classification tasks. It is arguable that if the methods developed in this study can perform reasonably well on corpora of such poor conditions, then they can be applied to any sign language corpus under better settings. 23

Additional videos from YouTube with higher quality have been selected for testing purposes too. For the first task of this study, the original data set was split into a training and testing set of 6,150 and 1,655 frames respectively and the labels were one hot encoded (i.e. signing as 1 and not-signing as 0). 24

After a successful training of the first prediction model, the tool was applied on a different part of the corpora. The predicted signing sequences were manually labeled as one- or two-handed signs. Together with randomly selected not-signing sequences (as predicted by the first tool), they formed a second data set. The size of this data set was slightly larger than the previous one: 10,120 frames in total. 25

3.2 Pre-processing

Using OpenPose, the pixel coordinates of the hands, elbows, shoulders and head were extracted from each frame. In the case of the handshape recognition module, the fingers joints coordinates were additionally extracted. We avoided using the finger extraction module of OpenPose on the first two parts of the study as that would have increased the computational time significantly. The positions of the rest of the body joints were disregarded as most of the time they were out of the frame bounds. Although the quality of the frames was poor, it created an advantage for the pose estimation framework, reducing the computational time to a reasonable level. 26

3.3 Tool 1: Manual Activation

The first tool is a temporal segmentation method to predict the begin and end frames of a sign sequence in a video sample. Thus, it is important to compare the performance of multiple different machine learning algorithms consistently. Four classification methods were used, namely: Support Vector Machines (SVM), Random Forests (RF), Artificial Neural Networks (ANN) and Extreme Gradient Boosting (XGBoost). The majority of these algorithms have been extensively used in machine learning studies as well as in sign language applications [Agha et al. 2018]. Performance was measured using the Area Under the Receiver Operating Characteristics (AUC) to validate each model. The AUC is a performance measurement specifically designed for binary (i.e. two class) classification problems. In general, it expresses how well a model is capable of distinguishing between classes, for example whether someone is signing in a video fragment or not. A model that makes random predictions will have an AUC of 0.5, a perfect model will have an AUC of 1. AUC stands for “Area under the Receiver Operating Characteristic (ROC) Curve”, the curve of True Positive Rate (probability of detection) versus False Positive Rate (probability of false alarms). It is better than just accuracy, i.e. percentage of correct predictions by the model, because it is not dependent on the relative amount of positives, i.e. percentage of total videos with signs in our case. We searched for the optimal setting of the various classification method parameters by exhaustive testing of the possible parameter settings and testing the performance on a validation set (“grid search”, searching the “grid” of possible parameter values). 27

3.4 Tool 2: Number of Hands

The second tool’s goal is to predict not only if a person is signing or not, but also to identify the number of hands involved (one- or two-handed). We hypothesized that this task is more complex than before, thus we considered it as a time-series problem. By using a sliding window technique, the original data set was parsed to form new training sets, where different possible frame intervals (1,2,3,5 and 10) were tested. Furthermore, similar (to some extend) classification methods with Tool 1 have been used [1]. 28

Moreover, recent studies in the sign language recognition field suggest that the use of Long-Short-Term-Memory (LSTM) networks can yield accurate results. LSTM is an artificial recurrent neural network (RNN) architecture used in the field of deep learning. Unlike standard feedforward neural networks (like the one tested in Tool 1) LSTM has feedback connections. It can not only process single data points, but also understand patterns in entire sequences of data, by combining its internal state resulting from previous input with a new input data item. In our case, instead of predicting whether a specific pose belongs into a class, we investigate whether a sequence of poses can be used for the same purpose. In this part of the study an LSTM network with different layer units as well as sliding window intervals has also been tested and compared with the above traditional machine learning classifiers. The overall architecture and technical details of the LSTM network can be found in Appendix A. 29

3.5 Tool 3: Handshape

The handshape recognition module was considered a so-called unsupervised learning problem as no ground truth information regarding this feature was available prior to the experiment, i.e. in contrast to the previous two problems we did not know what classes (handshapes) to detect. Such an unsupervised learning method can be useful in other newly compiled sign language or gestural corpora where there is no information regarding the different handshapes presented by the signers in the video. Additionally, an unsupervised learning method can be useful in other newly compiled sign language or gestural corpora where there is no information regarding the different handshapes presented by the signers in the video. We approached this as a clustering task: can we find groups of signs that were similar. Two different clustering methods have been tested: K-means and DBSCAN. The first clustering method was chosen for its simplicity as well as its fast implementation on the Python library that was utilized (namely scikit-learn). However, as the complexity of the data is unknown and it is case sensitive, it was decided to employ Density-Based Spatial Clustering of Applications with Noise (DBSCAN) as an alternative option. Given a set of points in some space, DBSCAN groups together points that are closely packed together, marking as outliers the ones that lie alone in low-density regions. This clustering method is one of the most common clustering algorithms. 30

Determining the optimal number of clusters (i.e. total number of expected handshapes) is a crucial issue in clustering methods such as K-means, which requires the user to specify the number of clusters *k* to be generated. The definition 31

of clusters is done so that the total within-cluster sum of square (WSS) is minimized, hence, in this study the *elbow method* was utilized to estimate the number of clusters.

3.5.1 Hand Normalization

Since the output of OpenPose contains the raw x, y pixel positions for the different finger joints, it is important to normalize them before applying the clustering method. To do so, the angle of the vector between the elbow and the wrist of the right hand is calculated. Subsequently, the coordinates of the finger joints positions are rotated to be in parallel on the horizontal axis and normalized so that their averaged location is at the origin. Figure 1 shows the output of the overall normalization process. All experiments were conducted using one machine with a hexa-core processor (Intel Core i7-3930K) and 4GB RAM. The models are implemented using the Python libraries scikit-learn [Pedregosa et al. 2011] and Keras [Chollet 2015] for their fast and easy implementation.

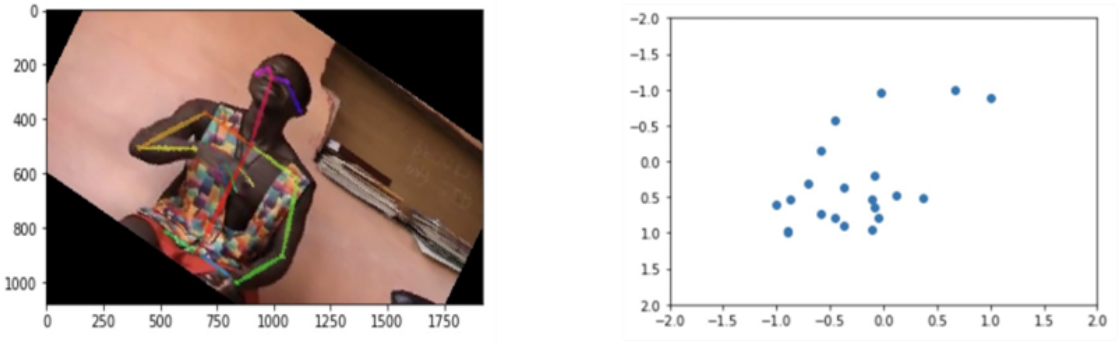


Figure 1. Signer's hand normalization is done based on the angle between the horizontal axis and the vector of the elbow and wrist coordinates. The finger joints are rotated according to that angle in order to be in parallel to the horizontal axis and scaled so that their average location is at the origin.

4. Results

The results section consists of three parts, the first part (Section 4.1) discusses the results of the analysis regarding the manual activation prediction. Section 4.2 discusses the results regarding the classification of one- and two-handed signing sequences. Last but not least, Section 4.3 presents the result regarding the handshape distribution using different clustering methods.

4.1 Tool 1: Manual Activation

All classifiers performed adequately well, apart from the Support Vector Machines (AUC: 0.80) (Table 1). Extreme Gradient Boosting (XGBoost) showed the highest AUC score at 0.92[2]. Figure 2 presents the ROC curve after a 10-fold cross-validation. The Artificial Neural Network was found to perform sufficiently well (AUC: 0.88). By exploring the importance of each feature on the prediction of the model we observe that the y and x pixel coordinates of the dominant (i.e. right) hand are on the top two positions (Table 2).

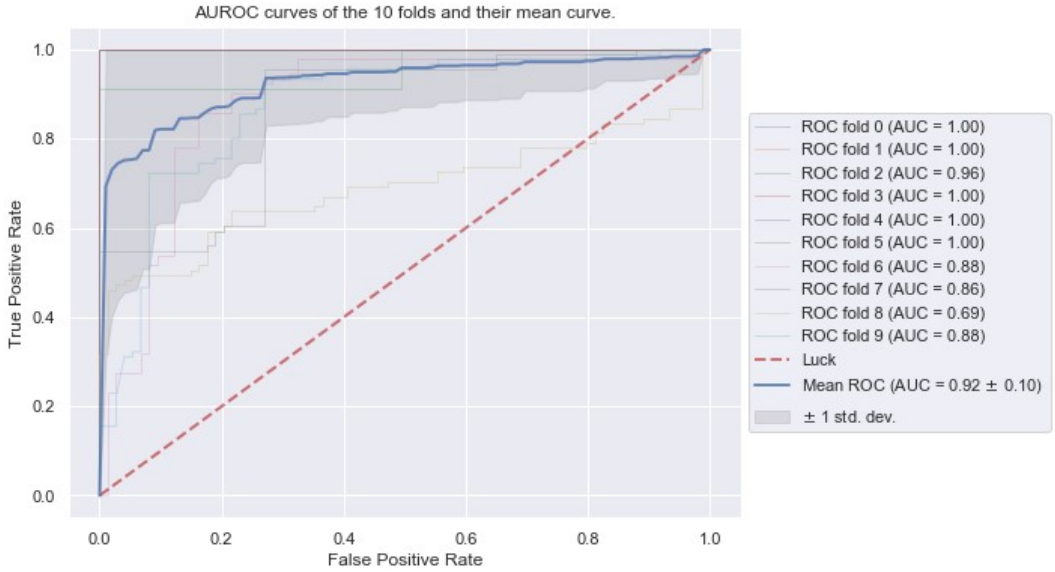


Figure 2. 10-fold cross validation of Extreme Gradient Boosting (XGBoost) classifier for the prediction of manual activation.

The fact that the Artificial Neural Network turned out to be a less efficient approach than the XGBoost can be accounted to the small training data set. Typically, Neural Networks require a lot more training data than traditional machine learning algorithms. Additionally, designing a network that correctly encodes a domain specific problem is challenging. In most cases, competent architectures are only reached when a whole research community is working on those problems, without short-term time constraints. Fine-tuning such a network would require time and effort that reach beyond the scope of this study. 35

To account for multiple people signing in one frame, an extra module was added. This module creates bounding boxes around each person recognized by OpenPose, normalizes the positions of the body joints and runs the classifier. This process makes it possible to classify sign occurrences for multiple people irrespective of their positions in a frame (Figure 4). 36

Once all the frames have been classified, the "cleaning up" and annotation phase starts. A sign occurrence is annotated only if at least 12 consecutive frames have been classified as "signing". That way we account for the false positive errors. This sets the stage for the annotation step. Using the PyMpi python library [Lubbers and Torreira 2013] the classifications are translated into annotations that can be imported directly to ELAN, a standard audio and video annotation tool [Sloetjes and Wittenburg 2008]. Figure 3 shows the result of the overall outcome. 37

Classifier	AUC score
Artificial Neural Network (ANN)	0.88
Random Forest (RF)	0.87
Support Vector Machines (SVM)	0.78
Extreme Gradient Boosting (XGBoost)	0.92

Table 1. AUC scores of all the classifiers tested for manual activation prediction

Weight	Feature
0.1410	Right wrist y
0.1281	Right wrist x
0.0928	Left wrist y
0.0917	Left wrist x
0.0717	Nose x
0.0658	Left shoulder x
0.0623	Left elbow y
0.0588	Right elbow y
0.0552	Nose y
0.0517	Left shoulder y
0.0482	Left elbow x
0.0482	Right elbow x
0.0482	Right shoulder y
0.0364	Right shoulder x

Table 2. Importance of each feature during manual activation as predicted by the Extreme Gradient Boosting classifier

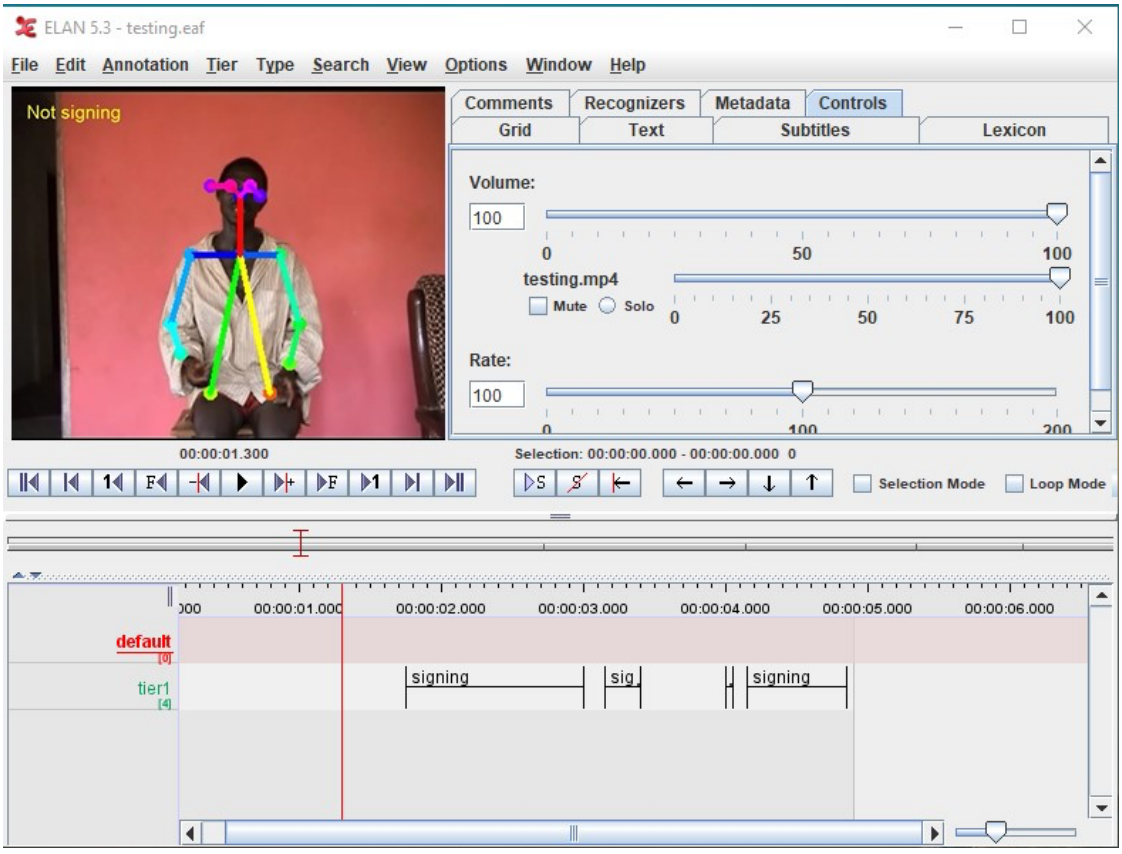


Figure 3. Final output of the manual activation tool as seen in ELAN. Signing sequences have been given a 'signing' gloss for readability. This attribute can be easily changed to produce empty glosses.

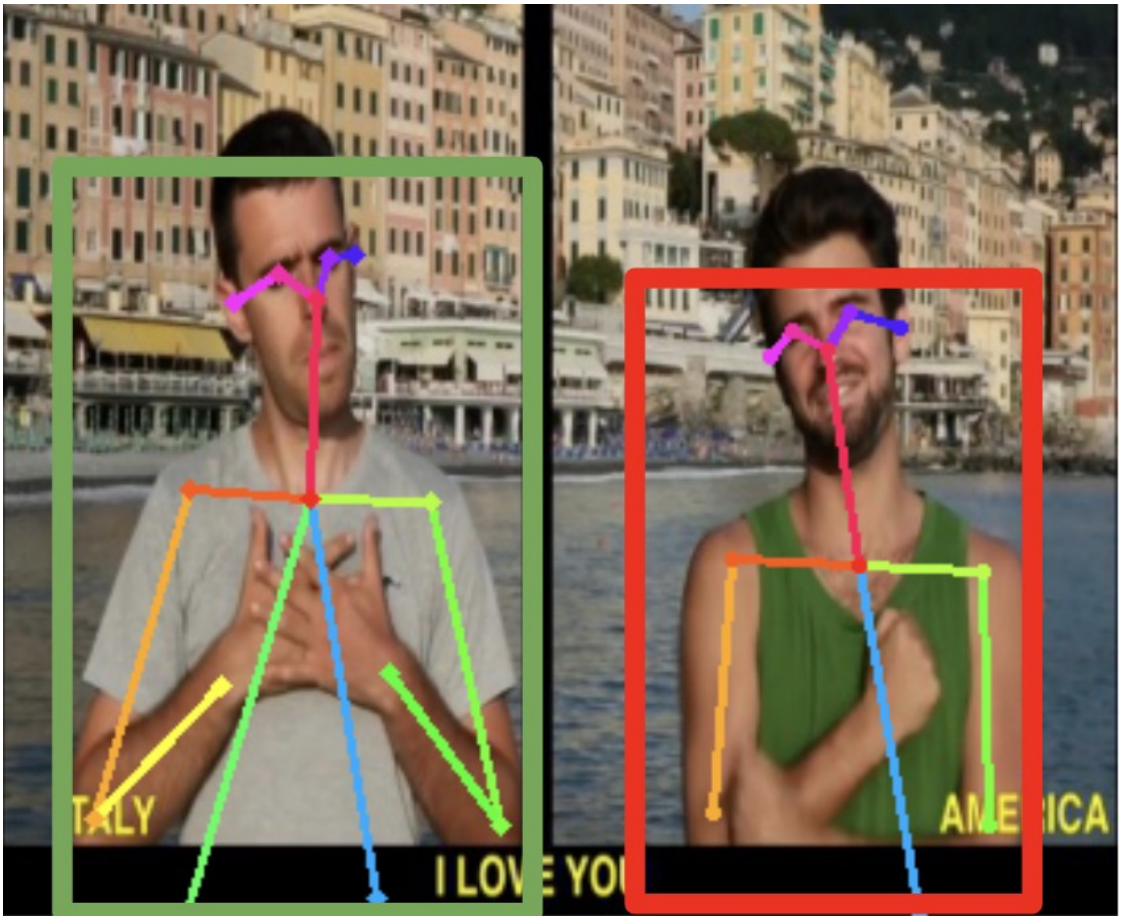


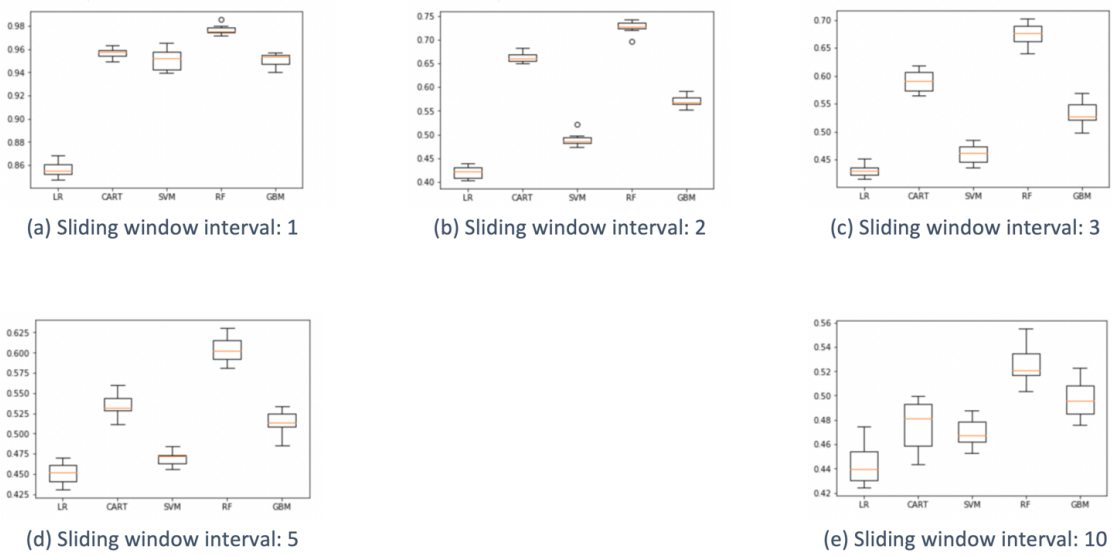
Figure 4. Bounding boxes are calculated in order to normalize the body joint coordinates for each signer. After this process the normalized coordinates are passed to the XGBoost classifier.[3]

4.2 Tool 2: Number of Hands

The second tool is responsible for not only recognizing whether a person in a video is signing but also if the sign is one or two-handed. We have previously hypothesized that this is a more complex task than the previous binary classification. Results on the accuracy of all the classifiers suggest that it is not as intricate as initially thought of; the higher the sliding window interval, the lower the accuracy of the model. As seen in Figure 5 of all classifiers tested, Random forest had the highest accuracy at the sliding window interval of 1 frame at a time. Similarly to the previous experiment, a frame-to-frame prediction can produce the highest results. 38

Furthermore, the results regarding the Long-Short-Term-Memory networks (Figure 6) suggest that the highest accuracy can be achieved at a sliding window interval of 56 frames and at a hidden layer size of 8 units. However, such a high window interval contains more than one sign, as the average length of a sign is approximately 14 frames. This discrepancy can be caused due to the architectural properties of the LSTM network. The average length of the signs is too small for the network to converge. The LSTM units needed more timesteps in order to prevent overfitting to the data. This property in addition to the small dataset used to train the network caused this anomaly. 39

Although the tool performs well on predicting whether a sign is one- or two-handed (using a Random Forest classifier) there are cases were the output is not as expected. In particular, cases where there is a two-handed symmetrical sign produced, the tool fails to accurately predict the correct class. It is likely that such signs were under-presented in our data set, thus resulting in poor classification. 40



(a) Sliding window interval: 1 (b) Sliding window interval: 2 (c) Sliding window interval: 3

(d) Sliding window interval: 5 (e) Sliding window interval: 10

Figure 5. Accuracy of different classifiers in various sliding window intervals in order to predict whether a sequence contains a one- or a two- handed sign or not signing at all. At a sliding window interval of 1 (a), Random Forest shows the highest accuracy.

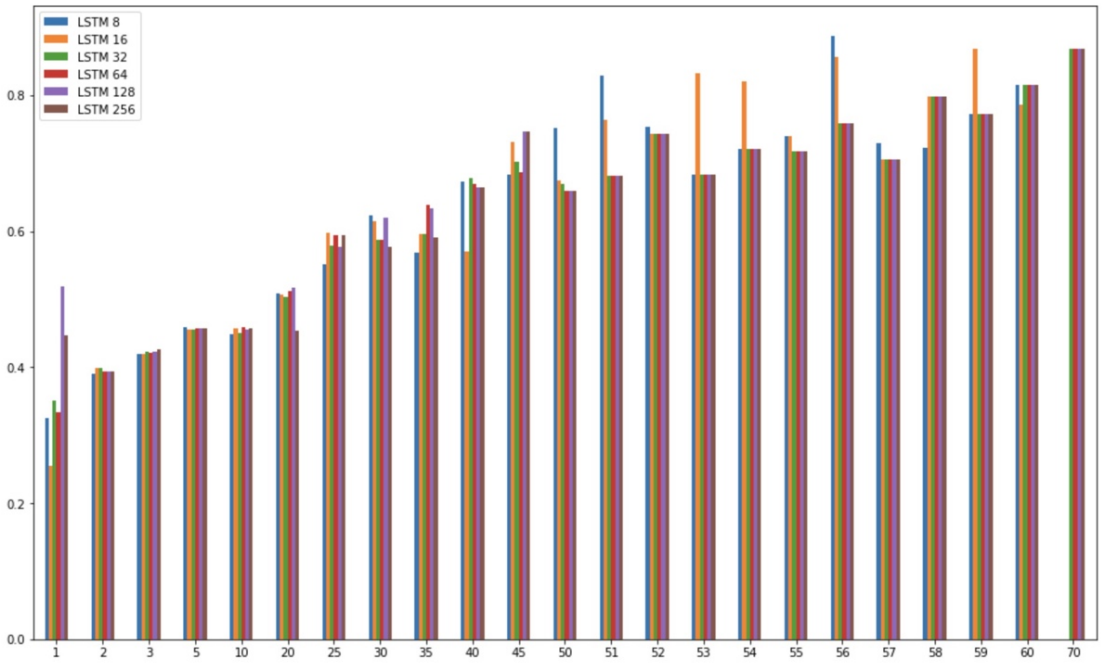


Figure 6. Accuracy of Long-Short-Term-Memory networks on the test set with different sliding window intervals (x) and hidden layer sizes (8,16,32,64,256).

4.3 Tool 3: Handshape

In order to understand the distribution of the different handshapes presented in a video, Principal Component Analysis (PCA) was utilized on all the normalized finger joint coordinates for all the frames at once (Figure 7a). This process allows us to reduce the dimensionality of the data while retaining as much as possible of the variance in the dataset. Each multidimensional array of the extracted finger joints positions, for each frame, has been reduced to a single x,y coordinate. The result already suggests that there are regions dense enough to be considered different clusters. The utilized elbow method suggested that at k=5 the highest classification could be achieved (Figure 7b). On the video sample used in our study that number seemed to reflect the proper amount of discerned handshapes. However, as OpenPose captures all the finger configurations in each frame it is at the linguist's discretion to decide on when a

handshape is significantly different from another. Additionally, experiments to optimize the hyperparameters (eta, min samples and leaf size) for the DBSCAN failed to create an accurate clustering (Figure 7c). Subsequently, the module creates annotation slots for the different handshapes in the video and adds an overlay containing the number of the predicted cluster on each frame.

However, special consideration must be given to the overall handshape recognition module. Although the hand normalization process prepares the finger joints adequately enough to be used in the clustering methods, it fails to account for hands perpendicular to the camera's point of view. Additionally, handshapes that are similar to each other but are rotated towards or outwards of the signer's body will most probably clustered differently. Some of these limitations can be solved by manually editing the cluster numbers prior to the annotation process. 42

In its current form, this method can already be used to either fully annotate the handshapes in a video sample or be used in different samples and treated as weakly annotated data in order to be used in other handshape classifiers similarly to Koller's et al. study [Koller et al. 2016]. 43

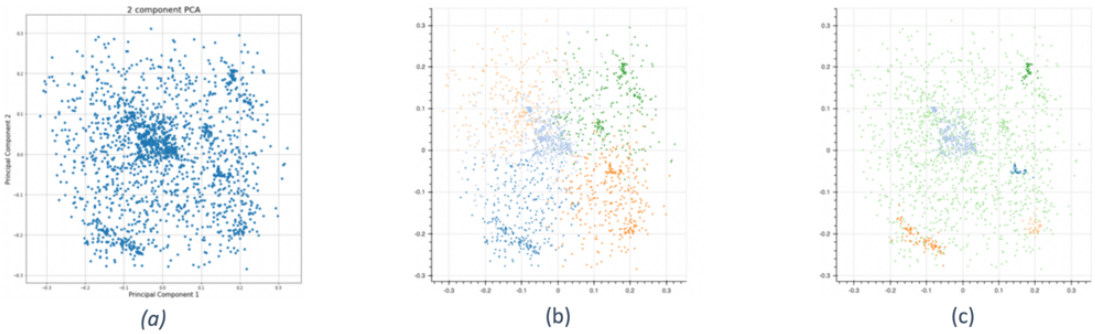


Figure 7. Visualizations produced by the handshape recognition module. Principal component analysis (a) can be used to reduce the dimensionality of the finger joints coordinates. Two clustering methods, namely K-means (b) and DBSCAN (c), can be used to detect the different handshapes presented in a video sample.

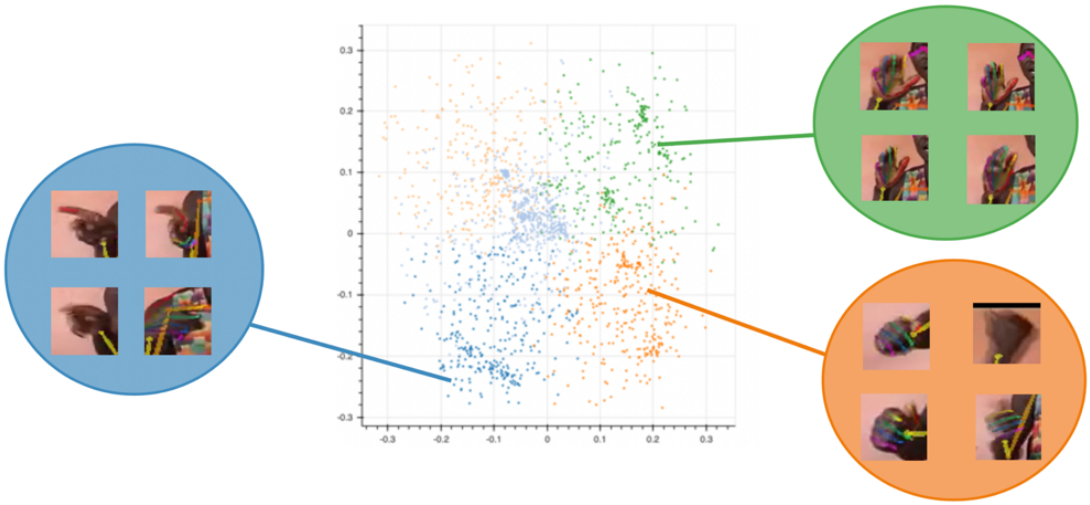


Figure 8. Different handshapes recognized by the handshape recognition module using K-means.

5. Discussion

In this study we have presented three different tools that can be used to assist the annotation process of sign language corpora. The first tool proved to be robust on the task of classification of manual activation even when the corpora are noisy, of poor quality and most importantly containing more than one signer. This eliminates the preprocessing stage that many sign language corpora have to endure where either dedicated cameras per signer are utilized or manually 44

cropping the original video. As a result, a more natural filming process can be applied. One limitation regarding our methodology is that at its current state is not possible to account for individual sign temporal classification. Reaching such level would require to fuse additional information into the training sets which in most cases might be language specific. However, it is possible to get a per sign prediction when the "number of hands involved" feature changes.

45 The most striking observation to emerge from our methodology is that there is no necessity of having massive training sets for the classification of low-level features (such as manual activation and number of hands involved). In contrast to earlier studies using neural networks for sign language recognition [Aitpayev et al. 2016] [Pigou et al. 2015] [Zhang et al. 2013], we used a proportionally smaller dataset. Additionally, this is the first time to our knowledge where corpora outside of studio conditions have been used to train and most importantly test models and tools for sign language automatic annotation. Furthermore, such findings can be applied in other studies as well. It is a common misconception that only large data sets can be used for analysis. Such a trend, although true for deep learning purposes, can be daunting for digital humanities researchers without in depth data science knowledge. In our study, we have shown that even with a small and noisy dataset of visual materials, researchers can use machine learning algorithms to effectively extract meaningful information. Our testing in West African Sign Language corpora showed that such frameworks can work effectively in different skin color participants lifting possible bias by previously developed algorithms.

46 There are few limitations regarding our methodologies, particularly with respect to the handshape distribution module. Low quality video and consequently framerate seem to affect the robustness of OpenPose. As a result, finger joint prediction can be noisy and of low confidence. Additionally, we observed that finger joints could not be predicted when the elbow was not visible in the frame, and thus, losing that information. In our study we treated all predicted joints equally but it is necessary for future research to include the prediction confidence interval as an additional variable. Furthermore, on the current output from OpenPose it is difficult to extract the palm orientation attribute meaning that differently rotated handshapes might result in the same cluster. Future research will concentrate on fixing that issue as well as creating an additional tool for the annotation of this feature.

47 In the sign language domain, researchers can use our tools to recognize the times of interest and basic phonological features on newly compiled corpora. Additionally, such extracted features can be further used to measure variation on different sign languages or signers, for example, to measure the distribution of one- and two-handed signs or particular handshapes. Moreover, other machine or deep learning experiments can benefit from our tools by using them to extract only the meaningful information from the corpora during the data gathering process, thus reducing possible noise in the datasets. Our tools can also be used towards automatic gloss suggestion. A future model can search only the signing sequences predicted by our tool rather than "scanning" the whole video corpus, and consequently making it more efficient.

48 Outside the sign language domain, the results have further strengthened our confidence that pre-trained frameworks can be used to help extract meaningful information from audio-visual materials. In particular, OpenPose can be a useful asset when human activity needs to be tracked and recognized in a video without the need of special hardware setups. Its accurate tracking allows researchers to use it in videos compiled outside studio conditions. As a result, studies in the audio-visual domain can benefit from community-created materials involving natural and unbiased communication. Using our tools, these study areas can analyze and classify human activity beyond the sign language discipline in large scale cultural archives or specific domains such as gestural research, dance or theater and cinema related studies, to name but a few. For example, video analyses in gestural and media studies can benefit from such an automatic approach to find relevant information regarding user-generated data on social media and other popular platforms.

49 Finally, due to the cumbersome installation process of OpenPose for the majority of SL linguists, we have decided to implement part of the tools in an online collaborative environment on a cloud service provided by Google (i.e. Google Colab). In this environment a temporary instance of OpenPose can be installed along with our developed python modules. In a simple step-based manner, the researcher can upload the relevant videos and download the automatically generated annotation files. Find the link to this Colab in the footnote below [4]. Additionally, we have a created another environment for re-training purposes. By doing so, the researcher can re-train the models on his or her particular data and ensure the aforementioned accuracy on them[5].

Conclusion

To summarise, glossing sign language corpora is a cumbersome and time-consuming task. Current approaches to automatize parts of this process need special video recording devices (such as Microsoft Kinect), large amount of data in order to train deep learning architectures to recognize a set of signs and can be prone to skin-color bias. In this study we explored the use of a pre-trained pose estimation framework created by Cao et al. [Cao et al. 2017] in order to create three tools and methods to predict sign occurrences, number of hands involved, and handshapes. The results show that four minutes of annotated data are adequate enough to train a classifier (namely XGBoost) to predict whether one or more persons are signing or not as well as the number of hands used (using Random Forest). Additionally, we examined the use of K-means and DBSCAN as clustering methods to detect the different handshapes presented in the video. Because of the low complexity of the finger joint data extracted from the pose estimation library, K-means was found to produce accurate results. 50

The significance of this study lies in the fact that the tools created do not rely on specialized cameras nor require large amount of information to be trained. Additionally, they can be easily used by researchers without developing skills and adjusted to work in any kind of sign language corpus irrespective of its quality or the number of people in the video. Finally, they have the potential to be extended and used in other audio-visual material that involve human activity such as gestural corpora. 51

Appendix A

The input shape of the LSTM network trained to recognize the “number of hands” (Tool 2) feature is a three dimensional array that can be defined as: [samples × timesteps × features] where features is a 2 dimensional array of [21 × 2] containing the pixel x,y coordinates of the finger joints and timesteps are the sliding window interval. Two Dense layers of 7 and 1 unit respectively follow the previous Bidirectional LSTM layer. The activation function used is “ReLU” and the dropout rate at 0.4. The architecture that produced the highest results for this network can be seen in Figure 9. 52

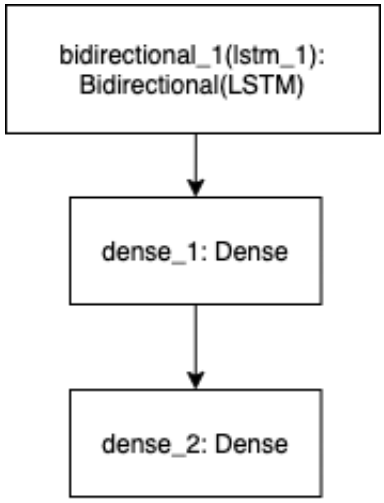


Figure 9. Architecture of the Long-Short-Term-Memory network trained for Tool 2.

Notes

[1] Linear Regression (LR), Decision Trees (CART), Support Vector Machines (SVM), Random Forest (RF) and Gradient Boosting (GBM).

[2] eta: 0.23, gamma: 3, lambda: 2, max. delta step: 4, max. depth: 37 and min. child weight: 4

[3] https://www.youtube.com/watch?v=NRe-AxZI8Hs&t=1s.

[4] https://colab.research.google.com/drive/1HwXo2Tk4uHizGTpRg-simMDMD4wPOzmA.

[5] https://colab.research.google.com/drive/10H2uxY7p59GrrDC5z85RugYB6PfQZs7Y.

Works Cited

Agha et al. 2018 Agha, R. A. A. R., Sefer, M. N. and Fattah, P. (2018). “A Comprehensive Study on Sign Languages Recognition Systems Using (SVM, KNN, CNN and ANN)”. *First International Conference on Data Science, E-Learning and Information Systems*. (DATA ’18). Madrid, Spain: ACM, pp. 28:1–28:6.

Aitpayev et al. 2016 Aitpayev, K., Islam, S. and Imashev, A. (2016). “Semi-automatic annotation tool for sign languages”. *2016 IEEE 10th International Conference on Application of Information and Communication Technologies (AICT)*. Baku, Azerbaijan: IEEE, pp. 1–4.

Andriluka et al. 2014 Andriluka, M., Pishchulin, L., Gehler, P. and Schiele, B. (2014). “2D Human Pose Estimation: New Benchmark and State of the Art Analysis”. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Buehler et al. 2009 Buehler, P., Zisserman, A. and Everingham, M. (2009). “Learning sign language by watching TV (using weakly aligned subtitles)”. *2009 IEEE Conference on Computer Vision and Pattern Recognition*. pp. 2961–68.

Cao et al. 2017 Cao, Z., Simon, T., Wei, S.-E. and Sheikh, Y. (2017). “Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields”. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, HI: IEEE, pp. 1302–10.

Charles et al. 2014 Charles, J., Pfister, T., Everingham, M. and Zisserman, A. (2014). “Automatic and Efficient Human Pose Estimation for Sign Language Videos”. *International Journal of Computer Vision*, 110(1): 70–90.

Chollet 2015 Chollet, F. and others (2015). *Keras*. https://keras.io.

Cooper and Bowden 2007 Cooper, H. and Bowden, R. (2007). “Large Lexicon Detection of Sign Language”. In Lew, M., Sebe, N., Huang, T. S. and Bakker, E. M. (eds), *Human–Computer Interaction*. (Lecture Notes in Computer Science). Springer Berlin Heidelberg, pp. 88–97.

Cooper et al. 2011 Cooper, H., Holt, B. and Bowden, R. (2011). “Sign Language Recognition”. In Moeslund, T. B., Hilton, A., Krüger, V. and Sigal, L. (eds), *Visual Analysis of Humans: Looking at People*. London: Springer London, pp. 539–62.

Dreuw and Ney 2008 Dreuw, P. and Ney, H. (2008). “Towards automatic sign language annotation for the ELAN tool”. *LREC Workshop: Representation and Processing of Sign Languages*. Marrakech, Morocco, pp. 50–53.

Eichner and Ferrari 2009 Eichner, M. and Ferrari, V. (2009). “Better appearance models for pictorial structures”. *British Machine Vision Conference 2009*. London, UK: British Machine Vision Association, pp. 3.1-3.11.

Eichner et al. 2012 Eichner, M., Marin-Jimenez, M., Zisserman, A. and Ferrari, V. (2012). “2D Articulated Human Pose Estimation and Retrieval in (Almost) Unconstrained Still Images”. *International Journal of Computer Vision*, 99(2): 190–214.

Fang et al. 2004 Fang, G., Gao, W. and Zhao, D. (2004). “Large Vocabulary Sign Language Recognition Based on Fuzzy Decision Trees”. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions On*, 34: 305–14 doi:10.1109/TSMCA.2004.824852.

Farhadi et al. 2007 Farhadi, A., Forsyth, D. and White, R. (2007). “Transfer Learning in Sign language”. *2007 IEEE Conference on Computer Vision and Pattern Recognition*. Minneapolis, MN, USA, pp. 1–8.

Felzenszwalb and Huttenlocher 2005 Felzenszwalb, P. F. and Huttenlocher, D. P. (2005). “Pictorial Structures for Object Recognition”. *International Journal of Computer Vision*, 61(1): 55–79.

Forster 2012 Forster, J., Schmidt, C., Hoyoux, T., Koller, O., Zelle, U., Piater, J. and Ney, H. (2012). “RWTH-PHOENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus”. Istanbul, Turkey, pp. 3785–3789.

Johnson and Everingham 2009 Johnson, S. and Everingham, M. (2009). “Combining discriminative appearance and segmentation cues for articulated human pose estimation”. *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*. pp. 405–12.

Johnston 2010 Johnston, T. (2010). “From archive to corpus: Transcription and annotation in the creation of signed language corpora”. *International Journal of Corpus Linguistics*, 15(1): 106–31 doi:10.1075/ijcl.15.1.05joh. http://www.jbe-platform.com/content/journals/10.1075/ijcl.15.1.05joh (accessed 19 May 2020).

Kelly et al. 2009 Kelly, D., Reilly Delannoy, J., Mc Donald, J. and Markham, C. (2009). “A framework for continuous multimodal sign language recognition”. *Proceedings of the 2009 International Conference on Multimodal Interfaces*. pp.

351–358.

Koller et al. 2016 Koller, O., Ney, H. and Bowden, R. (2016). “Deep Hand: How to Train a CNN on 1 Million Hand Images When Your Data is Continuous and Weakly Labelled”. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA: IEEE, pp. 3793–802.

Kumer 2017 Kumar, N. (2017). “Motion trajectory based human face and hands tracking for sign language recognition”. *2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON)*. Mathura: IEEE, pp. 211–16.

Lin 2014 Lin, T.-Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L. and Dollár, P. (2014). *Microsoft COCO: Common Objects in Context*.

Lubbers and Torreira 2013 Lubbers, M. and Torreira, F. (2013). *A Python Module for Processing ELAN and Praat Annotation Files: Dopefishh/Pympi*. Python https://github.com/dopefishh/pympi.

Moselund et al. 2006 Moeslund, T. B., Hilton, A. and Krüger, V. (2006). “A survey of advances in vision-based human motion capture and analysis”. *Computer Vision and Image Understanding*, 104(2): 90–126.

Nyst 2012 Nyst, V. (2012). *A Reference Corpus of Adamorobe Sign Language*. A digital, annotated video corpus of the sign language used in the village of Adamorobe, Ghana.

Nyst et al. 2012 Nyst, V., Magassouba, M. M. and Sylla, K. (2012). *Un Corpus de reference de la Langue des Signes Malienne II*. A digital, annotated video corpus of local sign language use in the Dogon area of Mali.

Pedregosa et al. 2011 Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., et al. (2011). “Scikit-learn: Machine Learning in Python”. *Journal of Machine Learning Research*, 12: 2825–2830.

Pigou et al. 2015 Pigou, L., Dieleman, S., Kindermans, P.-J. and Schrauwen, B. (2015). “Sign Language Recognition Using Convolutional Neural Networks”. In Agapito, L., Bronstein, M. M. and Rother, C. (eds), *Computer Vision - ECCV 2014 Workshops*. (Lecture Notes in Computer Science). Cham: Springer International Publishing, pp. 572–78.

Ramanan et al. 2007 Ramanan, D., Forsyth, D. A. and Zisserman, A. (2007). “Tracking People by Learning Their Appearance”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(1): 65–81.

Roussos et al. 2012 Roussos, A., Theodorakis, S., Pitsikalis, V. and Maragos, P. (2012). “Hand Tracking and Affine Shape-Appearance Handshape Sub-units in Continuous Sign Language Recognition”. In Kutulakos, K. N. (ed), *Trends and Topics in Computer Vision*, vol. 6553. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 258–72.

Sapp et al. 2010 Sapp, B., Jordan, C. and Taskar, B. (2010). Adaptive pose priors for pictorial structures. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. pp. 422–29.

Sivic et al. 2006 Sivic, J., Zitnick, C. L. and Szeliski, R. (2006). “Finding people in repeated shots of the same scene”. *British Machine Vision Conference 2006*, vol. 3. Edinburgh: British Machine Vision Association, pp. 909–18.

Sloetjes and Wittenburg 2008 Sloetjes, H. and Wittenburg, P. (2008). “Annotation by category - ELAN and ISO DCR. Marrakech, Morocco”, p. 5 https://tla.mpi.nl/tools/tla-tools/elan/.

Starner et al. 1998 Starner, T., Weaver, J. and Pentland, A. (1998). “Real-time American sign language recognition using desk and wearable computer based video”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(12): 1371–75.

Yang and Ramanan 2011 Yang, Y. and Ramanan, D. (2011). “Articulated pose estimation with flexible mixtures-of-parts”. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 1385–92.

Zhang et al. 2013 Zhang, C., Yang, X. and Tian, Y. (2013). “Histogram of 3D Facets: A characteristic descriptor for hand gesture recognition.” *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. pp. 1–8.