EE491-EE492
Senior Design Project Final Report

KEYWORD SEARCH FOR SIGN LANGUAGE

by

Aras Güngöre
Burak Batuhan Polat

A report submitted for EE491 senior design project class
in partial fulfillment of the requirements for the degree of
Bachelor of Science
(Department of Electrical and Electronics Engineering)
in Boğaziçi University

January 13th, 2023

Principal Investigator:
Murat Saraçlar

ACKNOWLEDGMENTS

We would like to thank our principal investigator Prof. Dr. Murat Saraçlar, Prof. Dr. Lale Akarun, and our former partner Göksu Karaman. The progress we made to this day became possible thanks to their contributions and guidance.

ABSTRACT

Integration of sign language detection software into our daily lives conveys importance in the aspect of increasing the living standards of deaf and mute people. This project aims to come up with a solution for this problem by expanding content already available to deaf and mute people. This motivation is accomplished by trying to create new tools and implementations to process the aforementioned content to present new content with a much wider range of use. In this project, we have taken the first steps of creating a software that recognizes sign language from visual content using MediaPipe hand tracking algorithm and minimum distance algorithms. Our research is not completed yet, however, we are getting promising preliminary results already. The prototype algorithm we used successfully recognizes the sign embeddings in its default dataset. We aim to expand the training dataset and further develop the existing prototype algorithm in the upcoming sections of this project.

LIST OF FIGURES

LIST OF TABLES

LIST OF APPENDICES

1. INTRODUCTION

Sign language is a visual-based language used by hearing-impaired people to communicate. Meaning is conveyed to the other side through hand shapes, facial expressions, mouth movements, and upper-body poses. Sign language recognition and translation from sign language are still unsolved problems due to their unique grammar and complexity [1]. In addition, the lack of a universal dataset that can be studied on sign language recognition, and the fact that sign language consists of thousands of different signs. The meanings of these signs change with some minor differences in the signs, so it causes some difficulties in recognizing these signs. Therefore, applications to be developed in this area are of great value.

The purpose of this project is to detect signs performed for the deaf from the videos in the data set and derive an efficient and low-cost keyword search (KWS) algorithm that maps sign expressions used in these videos to their respective words, and search for the word or sign received from the user in the relevant video. To accomplish such a task, the algorithm will learn from sign expression sequences and their text translations extracted from example videos. Keywords learned from example videos will be added to the vocabulary of the algorithm. Therefore, the algorithm will be able to find and translate sign expressions used in other videos if they exist in the vocabulary of the algorithm. Such an algorithm would make many operations related to sign-to-text translations possible.

2. OBJECTIVES

In modern society, since the term “equity” has its meaning more emphasized, all campaigns and applications that target to integrate disabled people including deaf people to society have gained more importance. One of the best improvements that can be done for deaf people is arguably performing sign language translation economically and effectively. The algorithm we developed has the purpose of helping deaf people to get more involved in modern society. It should make social interactions of these people easier and help them in their business lives.

The vocabulary of the algorithm consists of the keywords learned from the data videos the algorithm trained with. Thus, it is possible to extend the vocabulary of the algorithm by training it with new data sets if needed. This indicates a customizable and configurable vocabulary. A customizable and configurable vocabulary makes the algorithm applicable in more aspects of daily life, such as work, education, and healthcare. Also, the algorithm can be easily utilized by people from different countries and different native speakers.

Other than the aspects mentioned above, the program can translate the signs included in the video which is given as input from sign language to spoken language. This can be accomplished by learned keywords that are added to its vocabulary. Not only the program will do translation, but it will also present the other possible meanings and possibilities of the signs in the video to the user. By doing so, we hope to decrease the number of misunderstandings to a minimum.

3. BBC-OXFORD BRITISH SIGN LANGUAGE DATASET [2]

The How2Sign dataset we planned to use in the project was replaced with the BBC-Oxford British Sign Language Dataset upon the advice of our consultant. Thanks to this new dataset, we have more videos and better-subtitled data to use in the project. In this process, articles describing the data set in detail were read and detailed information about the data set was obtained. The information and methods we learned from these articles will be explained in detail below.

3.1. DATASET CONTENT

In the BOBSL dataset, there are BBC broadcasts, subtitles corresponding to the speeches in these broadcasts and sign language images of these speeches prepared according to British Sign Language. There are 426 different TV Shows in this data set and there are 1962 episodes for these TV shows in total. These publications are divided into 5 different genres, respectively, children, comedy, drama, entertainment, and factual. The distribution of these genres among the total departments is shown in figure 3.2, and the majority of these publications belong to the factual genre. In addition, these TV shows are divided into 22 different branches according to their subjects such as drama, history, horror, nature, and science documentaries. This distribution can be easily observed from figure 3.3.

Figure 3.1: BOBSL data [2]

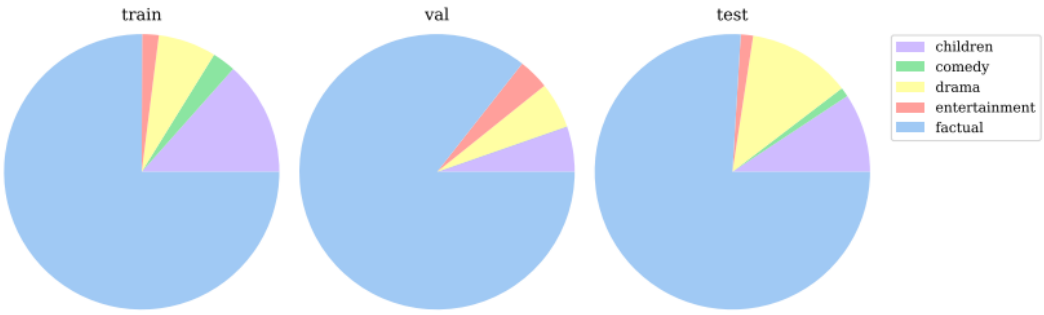

Figure 3.2: BOBSL genre distribution [2]

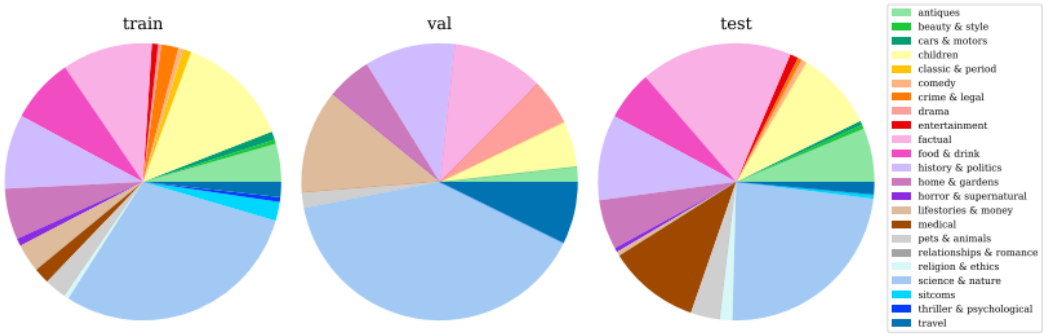

Figure 3.3: BOBSL topic distribution [2]

This 1962-episode dataset takes an average of 45 minutes and includes a total of 1467 hours of broadcasts. These videos are 444x444 pixels and have a frame rate of 25 fps.

A total of 1.2 million lines with a vocabulary of 78 thousand English words have been taken from English subtitles. A total of 39 signers are used in BOBSL (interpreters) but a few signers appear very frequently. To allow signer-independent assessment, we divided the data into train, validation, and test splits.

3.2. COMPARISON TO EXISTING DATASETS

Looking at other existing datasets for sign language recognition, sign spotting and translation such as American, German, Swiss-German, Finnish, Indian, and so on, some of these datasets have a limited number of different signers. It also has a limited vocabulary of signs and a limited number of times in some datasets. For example, the PHOENIX14T dataset has only 11 hours of broadcasting. The BOBSL data set, on the other hand, promises a better analysis than other data sets in terms of containing a large number of signers, long duration, and videos on different genres and subjects. It uses co-articulated signs and isolated signs, which is more representative of natural signing.

3.3. BOBSL DATASET CONSTRUCTION

In this section, methodology used while constructing the BOBSL dataset will be discussed and presented.

3.3.1. Source Data and Pre-processing

The source of BOBSL dataset is the TV episodes provided by BBC. These episodes were on air from 2007 to 2020 and have varying durations, the shortest being a few minutes and the longest being about 2 hours. The episodes are also drawn from different shows with various topics to enforce variety in future vocabulary. Most of the shows include a BSL interpreter, located at the lower right corner of the video track.

The videos are 16:9 display aspect ratio with a height of 576x pixels. Videos are recorded in 25 FPS.

Videos without the BSL interpreter are excluded from the dataset and pre-processing. A small minority of videos that cause major corruption of data were also excluded.

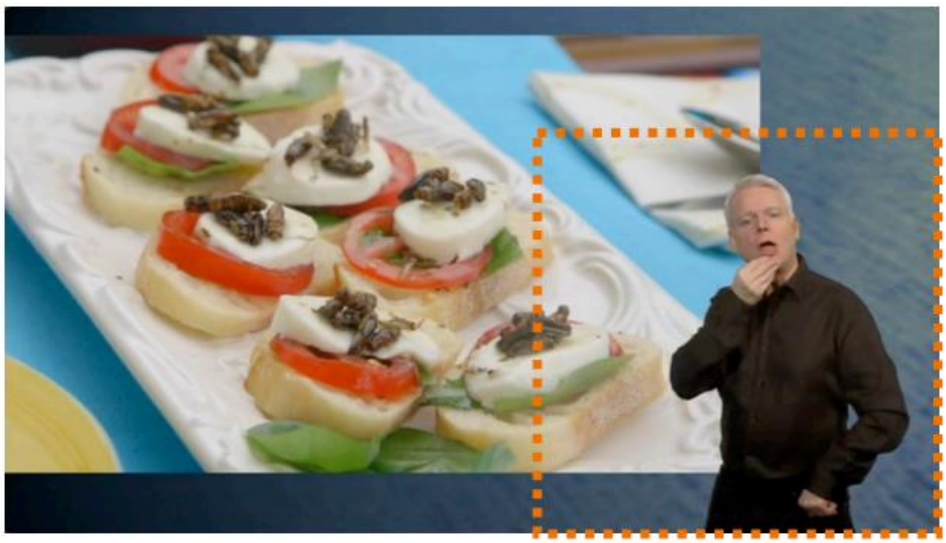

Figure 3.4: Preprocessed frame to only include BSL interpreter [2]

All videos are cropped to 444x444 pixel parts that only include the BSL interpreter, since the rest of the videos are irrelevant to the processing. An example of a cropped frame is demonstrated in figure 3.4. Then, the face of the BSL interpreter is detected using OpenPose to blur out other faces that appear in the background. A Gaussian filter is utilized to blur irrelevant faces from cropped frames.

After pre-processing the subtitles and video frames, audio from each video was removed since the audio of videos is not necessary for further processing operations. In the end, our set contained 1962 pre-processed videos with BSL interpreters and aligned English subtitles.

3.3.2. Dataset Splits

To endorse signer-independent systems, the datasets are classified in three different groups (train, validation, test splits) with respect to identified signers.

Identification of signers done by utilizing the RetinaFace face detector. The resulting face embeddings are calculated with a SE-50 network. Then, faces are classified into clusters by using spatial overlaps and cosine similarities. After that, clusters are checked manually to reassign erroneous cluster elements to their respective clusters. A total of 39 faces are identified using this methodology.

3.3.3. Automatic Annotation via Sign Spotting and Localization Methods

To extract meaningful data from weakly-aligned subtitles, three methodologies were introduced:

- mouthing keyword spotting
- dictionary spotting
- attention spotting

3.3.4. Sign Recognition Evaluation

The performance of models are evaluated with automatic training and validation sets and human-verified test set.

3.3.4.1. Recognition Vocabulary

BSL grammar is distinct from English grammar as the mapping between signs and words are many-to-many, while many signs can correspond to many words some words cannot correspond to any sign. This proves designing a proper vocabulary set a difficult task. For building a vocabulary set for sign recognition, every word is lemmatized, a set of words with at least 0.8 confidence that appear in the training set of mouthing annotations are filtered, and the words that don’t have any sign counterparts in this set are removed. 2281 words are in the sign recognition vocabulary.

3.3.4.2. Automatic Training and Validation Set

Figure 3.5 shows how many sign annotations when filtering and without filtering according to the recognition vocabulary and words in the unfiltered vocabulary are yielded for each training and validation set. We observe that mouthing spotting gives 707K and 15K, dictionary spotting gives 5M and 126K, attention spotting gives 434K and 9K, and total spotting (M, D, A) gives 6M and 151K annotations on training and validation sets respectively.

Split	Spotting source	#annots-2K	#annots-full	vocab
SIGN-TRAIN^{M}	mouthing	502K	707K	22.3K
SIGN-TRAIN^{D}	dictionary	1.587M	5.030M	6.7K
SIGN-TRAIN^{A}	attention	286K	434K	1.4K
SIGN-TRAIN^{M,D,A}	mouthing, dictionary, attention	2.374M	6.171M	24.7K
SIGN-VAL^{M}	mouthing	11K	15K	3.9K
SIGN-VAL^{D}	dictionary	38K	126K	3.9K
SIGN-VAL^{A}	attention	6K	9K	0.7K
SIGN-VAL^{M,D,A}	mouthing, dictionary, attention	56K	151K	5.9K
SIGN-TEST	mouthing, dictionary	25K	25K	1.8K

Figure 3.5: Sign instances for mouthing, dictionary, and attention partitions of the training set [2]

Since the BOBSL dataset was very large in size, we have switched to a ChaLearn LAP dataset for the time being. The dataset is called LSE_eSaude_UVIGO which is a dataset of Spanish Sign Language in the health domain.

4. LSE_ESAUDE_UVIGO SPANISH SIGN LANGUAGE DATASET [3]

4.1. DATASET CONTENT

The LSE_eSaude_UVIGO dataset is signed by 10 people and partially annotated with 100 signs. While common datasets of continuous sign language are collected from real-time translation or subtitled broadcasting, the LSE_eSaude_UVIGO dataset is collected from signers expressing the contents of Spanish Sign Language regarding the health domain. This increases the expressivity and naturalness in the dataset.

The dataset is made at 25 fps using uniform illumination in studio conditions. The dataset includes many meticulously hand-made annotations made by deaf people and professional translators sign by sign.

4.2. ANNOTATION

Annotators use an associated Tier ‘M_Glosa’ to indicate the selected gloss locations and an associated Tier ‘Var’ to indicate variants of signed gloss. Such variants can be classified as linguistic and non-linguistic. Linguistic variants are minor alterations of the sign which can stem from and can be coded as morphology changes (MPH), unusual non-dominant hand use (MAN), change of location (LOC), and relaxed execution (LAX). Non-linguistic variants can be classified as the existence of a similarly signed gloss with the selected gloss (SIM), blockage of the sign if it is out of frame (OUT) or if it is blocked by other body parts (OUT), and sign appearance in very short durations (SHO).

The annotation criteria which appear at all the annotators are:

- begin_timestamp is set when the hand configuration, location, and movement match a certain sign more than the previous sign.
- end_timestamp is set when the hand configuration, location, and movement don't match the sign more than the following sign.
- Transitions are excluded from annotated time intervals.
- Linguistic (MPH, MAN, LOC, LAX) and non-linguistic (SHO, OCC, OUT, SIM) variants in the signed gloss are indicated with a '*' prefix.

4.3. DATA SPLIT AND STRUCTURE

The dataset is divided into 5 splits and a query set for both tracks. Track 1 is named multiple-shot supervised learning (MSSL) and track 2 is named one-shot learning and weak labels (OSLWL). The contents of the training, validation, and testing sets are given below.

4.3.1. MSSL

In the MMSL track, each video file in the datasets contains a unique signer and the video filenames are in the format p##_n###, which indicate the code of the signer and a unique sequential number.

4.3.1.1. MSSL_Train_Set

The training set for the MSSL track includes video files with annotations of 60 signs which have a total duration of 2.5 hours.

4.3.1.2. MSSL_Val_Set

The validation set for the MSSL track includes video files with annotations of the same 60 signs which have a total duration of 1.5 hours.

4.3.1.3. MSSL_Test_Set

The testing set for the MSSL track includes video files with annotations of the same 60 signs which have a total duration of 1.5 hours.

4.3.2. OSLWL

4.3.2.1. OSLWL_Query_Set

The query set for the OSLWL track includes 40 short videos corresponding to 40 isolated signs. 20 videos are recorded by a deaf man included in the dataset whereas the other 20 is recorded by a female interpreter not included in the dataset to evaluate more realistic query searches.

4.3.2.2. OSLWL_Val_Set

The validation set for the OSLWL track includes nearly 1500 video files with a 4 second duration where each contain one sign out of 20 signs to simulate weak labels. Video filenames are in the format p##_s##_n###, which indicate the code of the signer, code of the sign to be retrieved, and a unique sequential number.

4.3.2.3. OSLWL_Val_Set

The testing set for the OSLWL track is structured similarly to the OSLWL_Val_Set and only differ in the 20 query signs and signers.

5. SIGN LANGUAGE DETECTION

To detect and recognize signs performed in a video, we deployed a prototype algorithm [4] that uses MediaPipe [5] and DTW (Dynamic Time Warping). MediaPipe is an open-source cross-platform framework that offers customizable Machine Learning solutions. Utilities provided by MediaPipe framework includes face detection, hand tracking, pose detection and many more. In this project, we deployed MediaPipe to detect and track spatial and temporal hand movements. Solution offers hand tracking by establishing 21 key points in a human hand. We plan to implement the datasets provided to us, namely LSE_eSaude_UVIGO Spanish Sign Language Dataset and BOBSL (BBC-Oxford British Sign Language Dataset), to this algorithm.

The algorithm we deployed uses MediaPipe framework to extract the locations of key points of the hand or hands (also referred as landmarks) in a frame. To accurately extract information about signs performed in a frame, algorithm calculates the relative angle of each landmark rather than their distance and absolute position since hand size and position may differ signer to signer. To extract hand information from a frame, all 21 hand connections established by MediaPipe are utilized in the algorithm. These connections and landmarks are shown in Figure 5.1.

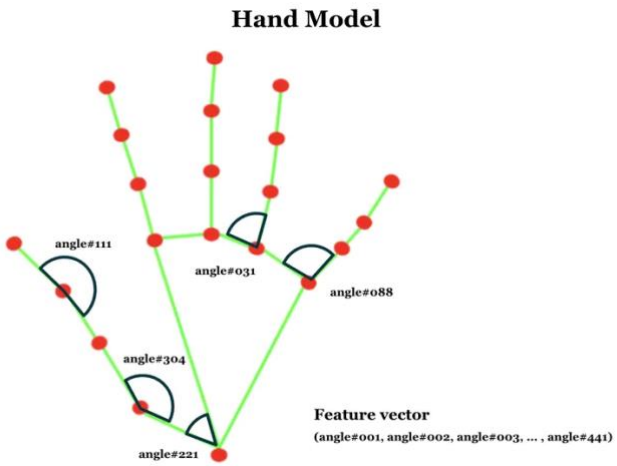

Figure 5.1: Hand connections and feature vector [4]

A feature vector containing all 21 x 21 = 441 angles between all landmarks is constructed to identify hand gesture performed in a frame. Feature vectors for every frame of a training video in training set is obtained to compare and detect signs performed in a test video. Then, spatial landmark information from each frame is combined into a matrix to obtain temporal information of the hand movements.

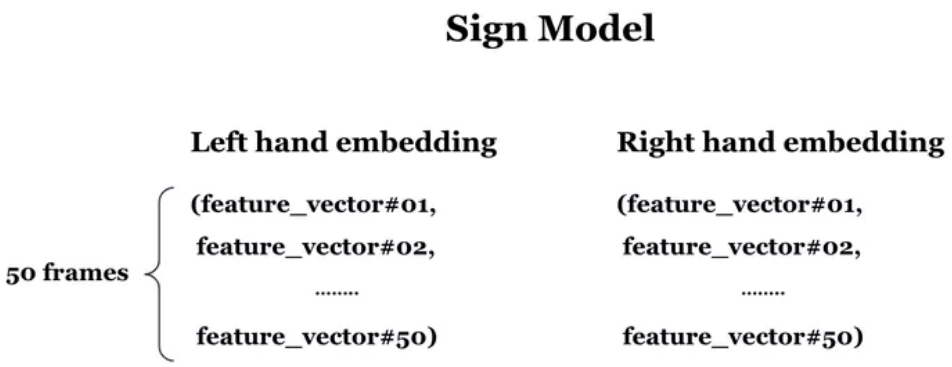

Figure 5.2: Sign model representation [4]

After the detection of temporal and spatial information of hands during a sign, the algorithm classifies a sign model by combining these details. Both left and right hand (if present in the video) feature vectors for every frame are assembled to create a sign model that contains all information associated with that sign. The sign models are used to predict the sign performed in a test video.

After the construction of sign models, Dynamic Time Warping algorithm is deployed to detect similarities between signs. This algorithm allows sign prediction with less training data compared to deep learning methods. DTW algorithm calculates the similarities between patterns instead of sequences, resulting in better performance while comparing samples with different lengths. More information and mathematical background of DTW algorithm can be found in Appendix A.

To predict a sign from test set, the algorithm calculates distances between the test feature matrix and all other training feature matrices. Then, each training sample is ranked based on a similarity score obtained from DTW distances to the training feature matrices. After these operations, the algorithm calculates the n most probable signs based on similarity and classifies the test sign according to an arbitrary threshold value. If the test sign cannot be classified, algorithm outputs “Unknown Sign”.

6. CONCLUSION

6.1. RESULTS

After many hours of researching and article-reading, we were able to get our first material results. The algorithm we found successfully recognized the narrow set of signs given in its default training sets.

6.2. DISCUSSION AND FUTURE WORK

Since start of the first semester, considerable forward steps had been taken in this project. We were extremely unfamiliar with the subject, however, we were ready to study and comprehend topics required. After many hours of preliminary study and literature survey, we were able to get our first results using the aforementioned algorithm. These results may not look that much, however, they are promising. Our next milestone would be widening the vocabulary of the keyword recognition by implementing LSE_eSaude_UVIGO Spanish Sign Language Dataset and BOBSL (BBC-Oxford British Sign Language Dataset) in near future. We had some delays due to unforeseen circumstances but we believe we can fulfill the requirements of the project in proposed time schedule.

6.3. REALISTIC CONSTRAINTS

Since sign language consists of various components such as hand and arm movements, lip movements, and facial expressions; it makes it a challenging problem to extract feature vectors that fully reflect the sign language in all aspects. Moreover, there is not an universal data set for sign language recognition, since sign languages vary from region to region. Also, the fact that sign languages consist of thousands of different signs and the differences between these signs is minor proposes yet another challenge for sign language recognition studies. While there are various labeled data

sets for sign language recognition, these data sets are labeled by hand and thus their production is pretty expensive.

6.4. SOCIAL, ENVIRONMENTAL AND ECONOMIC IMPACT

This project plays an important role in integrating people with speaking and/or hearing disabilities to the society. The two most important elements of social construct, communication and culture is provided to the disabled people. The bond between abled and disabled people is strengthened as it will create a common ground between sign written languages for information retrieval and make it easier for non-sign language speakers to learn the sign language. Hence, disabled people don't have to adapt to lip reading or orally speaking as effective sign language communication can be made with people who don't understand sign language. Since communication is improved, deaf people can easily be integrated to the workforce in various industries. Moreover, this study can be used in educational purposes involving teaching sign language to both disabled and abled people. Also, this project will make it possible to use parallel databases between spoken and sign languages. Finally, this project can also be used as the first step for helping hearing-impaired people extracting the information, they want from a video in sign language.

6.5. COST ANALYSIS

R&D cost (28 weeks x 6 hours x 2 students x $5/hour)	$1680
Data storage (physical or cloud)	~$60
GPU server (~$75/month x 7 months)	~$525
TOTAL	***~$2265***

6.6. STANDARDS

The designs in this project will conform to the IEEE, IET, EU and Turkish engineering standards. Engineering code of conduct will be followed during all processes including the reports and presentations.

APPENDIX

APPENDIX A: DYNAMIC TIME WARPING (DTW)

Dynamic Time Warping (DTW) is an algorithm that allows the calculation of the distance between sequences with different lengths. This method is widely used in signal processing since most operations often require comparison of data sequences with varying lengths.

Suppose two one-dimensional vector sequences $\mathbf{x} = \{x_1, x_2, \dots, x_N\}$ and $\mathbf{y} = \{y_1, y_2, \dots, y_M\}$ where N > M (**y** is longer than **x**). To calculate DTW distance between **x** and **y**, the cost matrix $\mathbf{D} \in R^{(N+1)x(M+1)}$ must be constructed. For initialization, let $D_{i,0} = \infty$ for i = 1, 2, … , N; $D_{0,j} = \infty$ for j = 1, 2, … , M and $D_{0,0} = 0$. Then, fill the rest of the construction matrix using the equation

$$D_{i,j} = d(x_i, y_j) + \min(D_{i-1,j-1}, D_{i-1,j}, D_{i,j-1}) \quad (1)$$

$$d(x_i, y_j) = |x_i - y_j| \quad (2)$$

where $d(x_i, y_j)$ is the distance function, given in Equation 2.

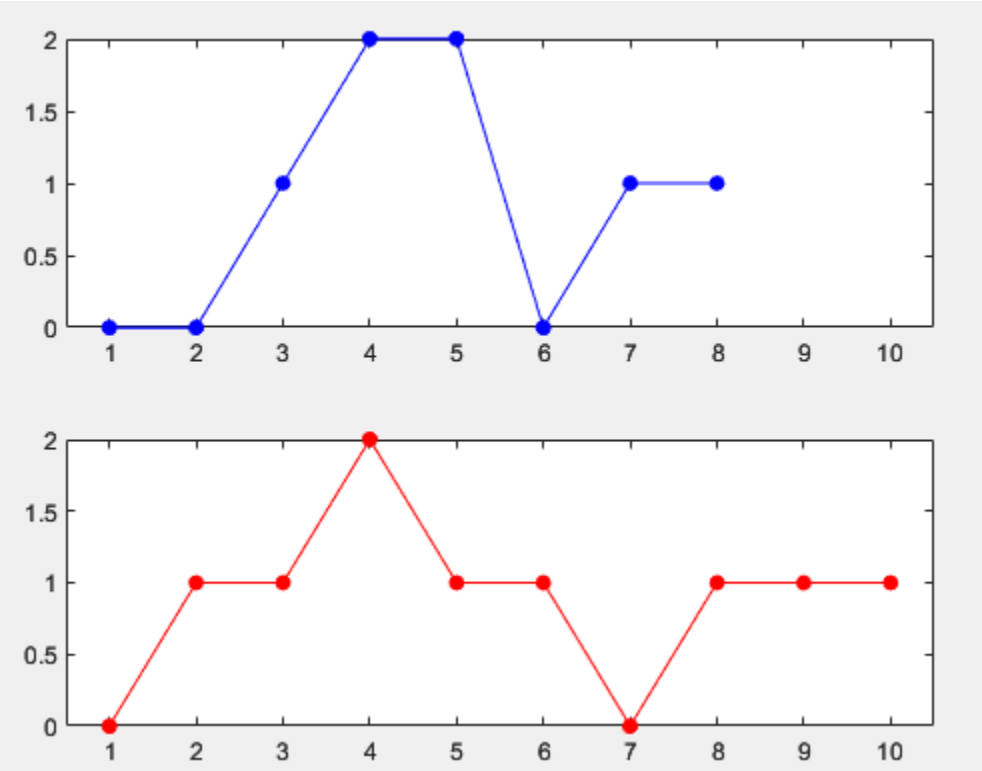

Figure A.1: Plots of x (upper plot) and y (lower plot)

After the cost matrix **D** is constructed, optimal element match between **x** and **y** can be determined by tracing back from $D_{M,N}$ to $D_{0,0}$ using the path with the minimum cost. An example is provided below.

Suppose the sequences **x** = {0, 0, 1, 2, 2, 0, 1, 1} and **y** = {0, 1, 1, 2, 1, 1, 0, 1, 1, 1}. Plots of **x** and **y** is given in Figure A.1.

Initialize **D** by letting $D_{i,0} = \infty$ for i = 1, 2, … , N; $D_{0,j} = \infty$ for j = 1, 2, … , M and $D_{0,0} = 0$. Initialized cost matrix is provided in Table A.1.

After filling the cost matrix according to Equation 1 and tracing accordingly, cost matrix is completed. Completed cost matrix is given in Table A.2. Optimal element matching between **x** and **y** is given in Figure 2. As can be seen from Table A.2, more than one element matching may be optimal. Deciding factor of the distance between **x** and **y** is the cost of the matching, which is 2 in this case.

x_8	1	∞									
x_7	1	∞									
x_6	0	∞									
x_5	2	∞									
x_4	2	∞									
x_3	1	∞									
x_2	0	∞									
x_1	0	∞									
x_0	-	0	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
		-	0	1	1	2	1	1	0	1	1	1
			y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}

Table A.1: Initialized cost matrix

x_8	1	∞	7	3	3	4	1	1	2	2	2	2
x_7	1	∞	6	3	3	3	1	1	2	2	2	2
x_6	0	∞	5	3	3	2	1	2	2	3	4	5
x_5	2	∞	5	2	2	0	1	2	4	4	4	4
x_4	2	∞	3	1	1	0	1	2	3	3	3	3
x_3	1	∞	1	0	0	1	1	1	2	2	2	2
x_2	0	∞	0	2	2	4	5	6	6	7	8	9
x_1	0	∞	0	1	2	4	5	6	6	7	8	9
x_0	-	0	∞	∞	∞	∞	∞	∞	∞	∞	∞	∞
		-	0	1	1	2	1	1	0	1	1	1
		y_0	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8	y_9	y_{10}

Table A.2: Complete cost matrix (one of the optimal element matchings is highlighted)

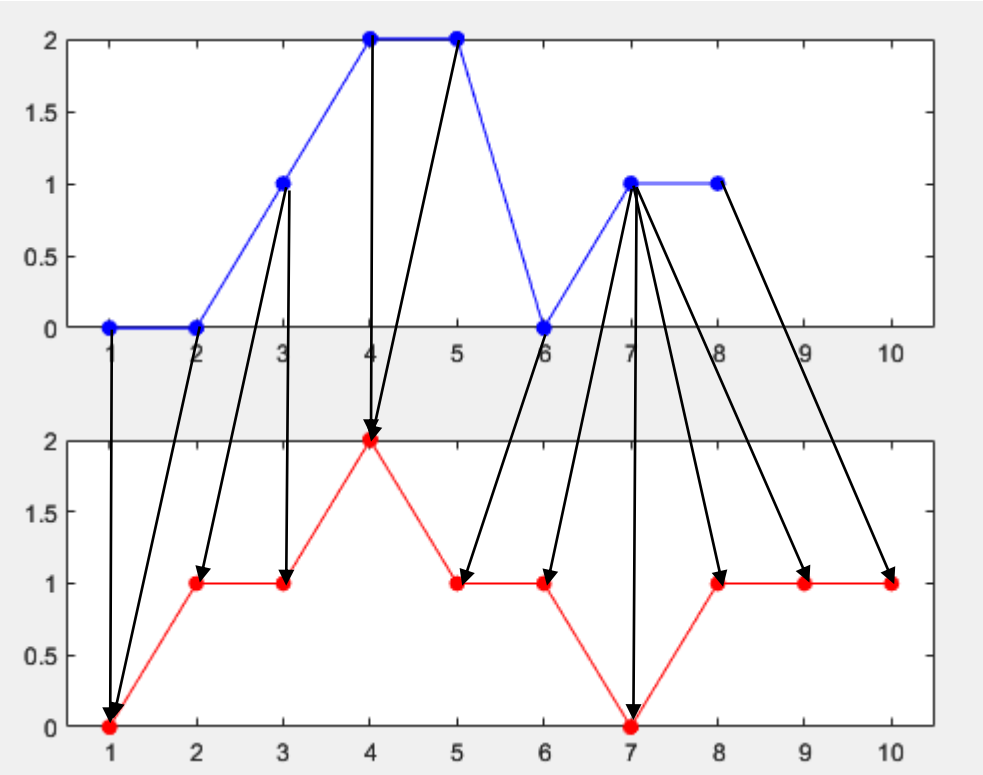

Figure A.2: Optimal element matching between x and y

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] N. C. Tamer and M. Saraçlar, "Keyword Search for Sign Language," *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8184-8188, doi: 10.1109/ICASSP40776.2020.9054678.

[2] Albanie, S., Varol, G., Momeni, L., Bull, H., Afouras, T., Chowdhury, H., Fox, N., Woll, B., Cooper, R., McParland, A., & Zisserman, A. (2021). BBC-Oxford British Sign Language Dataset. *arXiv*. https://doi.org/10.48550/arXiv.2111.03635

[3] https://chalearnlap.cvc.uab.cat/dataset/42/description/

[4] https://www.sicara.fr/blog-technique/sign-language-recognition-using-mediapipe

[5] https://mediapipe.dev