Journal of Advanced Research in Technology and Management Sciences
Volume: 05 Issue: 01 ISSN: 2582-3078 January 2023
Available online at: http://www.jartms.org

Sign Language Interpreter Using HPR-SLI

Dr. Rubini P[1], Dr. Saravana Kumar S[2], Pavan Sreenivas[3], Sacharith Ashok[4], Surya V[5], Vamshidhar Reddy B[6]

[1-4]*Department of Computer Science and Engineering,* [5-6]*Department of IT(CTIS),* [1-6]*CMR University*
[1-6]*Off-Baglur Main Road, Chagalatti, Bengaluru – 562149, Karnataka, India*
[1]rubini.p@cmr.edu.in, [2]sarvana.k@cmr.edu.in, [3]pavan.19bcs@cmr.edu.in
[4]sacharith.19bcs@cmr.edu.in, [5]surya.19bct@cmr.edu.in, [6]vamshidhar.19bct@cmr.edu.in

***Abstract* – The deaf and dumb people widely use sign language for communication; however, they find it difficult to communicate with people who don't understand Sign Language. In this paper, a Hand Pose Recognition-based sign language Interpreter (HPR-SLI) system is proposed, which translates sign language using Natural Language Processing (NLP). Sign languages are expressed by hand and come in combination with elements such as postures, body movements, head, eyebrows, eyes, face and mouth, which are used in different combinations to present several pieces of information. The HPR-SLI system enables consecutive sign languages to be translated into readable English. A hand pose recognition method based on an Artificial Neural Network (ANN) and a language processing system based on a Text-to-Text Transfer Transformer (T5) model which is assisted by a tokenizer. The proposed system can be deployed as a mobile application or into a physical device designed specifically for the intended purpose. The whole idea of the paper is to design and develop a system that significantly lowers the communication gap between the speech-hearing impaired and the normal world. The technology HPR-SLI aims to help PwDs (Person with Disabilities) overcome most of their disadvantages and enable them to communicate and develop better social relationships to lead a normal life in the society.**

***Keywords:* Social well-being, hand pose recognition-based sign language interpreter, LSTM, Linear Regression, Artificial Neural Network (ANN), Natural Language processing (NLP), T5 Tokenizer.**

I. INTRODUCTION

By sharing their ideas, opinions, and experiences with others around them, humans get to know one another. There are several methods to accomplish this, but the gift of "Speech" stands out among the rest as the best one. Everyone can communicate their thoughts and comprehend one another through speech quite effectively. It would be unfair to neglect the deaf and people who are denied this priceless gift. The usage of "Sign Language" is the only form of communication available to the deaf and mute. They can only communicate in their small environment through sign language. Due to this restriction, they are unable to communicate with others and express their emotions, original ideas, and potential. Only a handful of people in the society come forward to learn and comprehend sign language. Due to these restrictions, mute and deaf persons are increasingly excluded from mainstream culture. One approach to overcome this obstacle and help these people is through technology.

An intelligent and natural method of interacting with computers for people is hand gesture recognition (HCI). A subfield of artificial intelligence, human computer interaction (HCI) is concerned with the creation of algorithms that take empirical data from sensors or databases as input and produce patterns or predictions that are believed to be characteristics of the mechanism that produced the data. The development of algorithms that recognise intricate patterns and make wise decisions based on incoming data is a key area of research in HCI. There is a great possibility to create more natural Human-Computer Interfaces that rely on user gestures as the integration of digital cameras into personal computing devices becomes a major trend. Computer science and language technology research in the field of hand gesture recognition tries to define human gestures using mathematical methods. Humans can interact with machines effortlessly without the use of any mechanical equipment thanks to gesture recognition. Among the various gestures, the hand gesture is one of the most expressive and widely used. Through hand gestures, body language, facial emotions, and sign language, people can express themselves visually. There is no universally acknowledged sign language. There are many variations of sign languages because, like spoken languages, they evolved naturally via interaction between various groups of people. Between 138 and 300 different sign languages are currently in use all over the world. It's interesting to note that most nations with similar spoken languages do not necessarily have the same sign language. Australian Sign Language, American Sign Language (ASL), British Sign Language (BSL) are variants of English. This paper addresses key aspects of Sign Language Recognition (SLR), starting with a brief introduction to motivations and requirements, followed by a summary of Sign Linguistics and its impact on the field. There is an undeniable communication problem between the Deaf community and the hearing majority. Thus, we propose a new system called HPR-SLI which will be helpful for the impaired people for conveying their views to others.

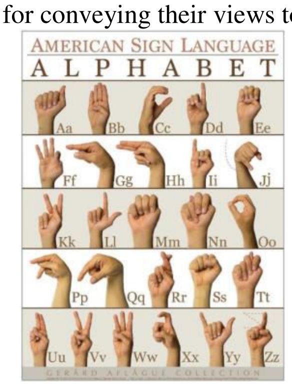

Fig. 1 Alphabets of American Sign Language

Journal of Advanced Research in Technology and Management Sciences
Volume: 05 Issue: 01 ISSN: 2582-3078 January 2023
Available online at: http://www.jartms.org

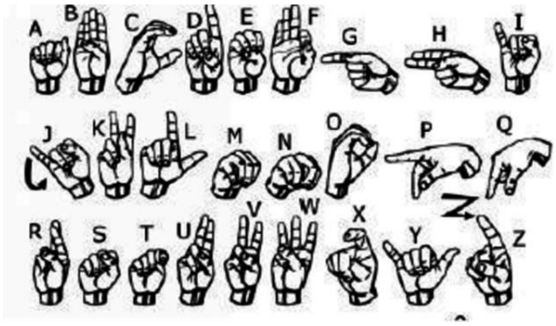

Fig. 2 Single hand alphabets of Indian Sign Language

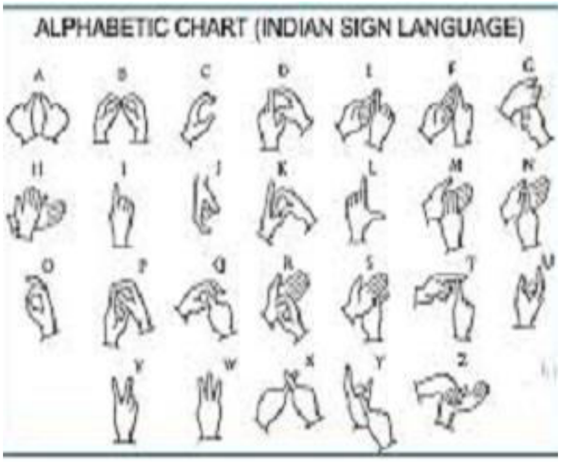

Fig. 3 Double Hand Alphabets of Indian Sign Language

II. LITERATURE SURVEY

A work on Sign language Recognition Using Machine Learning Algorithm was presented by Prof. Radha S. Shirbhate, Mr. Vedant, and D. Shinde in 2020[1]. The approach used to address the classification problem was broken down into three parts. As the remaining portion of the image can be viewed as noise in terms of the character classification problem, the first step is to separate the skin portion from the image. The second step is to extract pertinent features from the skin-segmented images that can be useful for the learning and classification stages to follow. The extracted features are used as input into multiple supervised learning models in the third stage, as indicated above, and then the trained models are used for classification.

A work on Real-Time Sign language Detection using position estimation was presented in 2020 by Amit Moryossef, Loannis Tsochantaridis, and Roee Aharoni [2]. The task of determining whether or not a person is using sign language in each particular frame of a video is known as sign language detection. The goal of sign language detection is to identify when something is being signed, as opposed to sign language recognition, which aims to recognise the shape and meaning of signs in videos, or sign language identification, which aims to determine which sign language is being used. Based on position estimation, we suggest a straightforward human optical-flow representation for videos, which is then input to a temporally sensitive neural network to conduct a binary classification for each frame: is the person signing or not? And the results were compared and used for real time application.

In their study Sign Language Recognition Using Deep Learning on Custom Processed Static Gesture Images, Aditya Das, Shantanu Gawde, and Khyati Suratwala (2018) [3] used convolution neural networks to recognise sign language motions. The static sign language gestures were photographed using an RGB camera and comprise the image dataset used. The photos underwent pre-processing before being used as the input after being cleaned. Inception v3 convolutional neural network model was used to retrain and test this dataset of sign language motions, and the findings are presented in this study. The resulting validation accuracy was greater than 90%. This study also examines the many attempts at sign language utilising machine learning and picture depth information.

The paper titled “Visual Alignment Constraints for Continuous Sign Language Recognition” [4] by Yuecong Min et.al., (2021) published their results to identify an unsegment signs from image streams, the Continuous Sign Language Recognition (CSLR) method was developed. One of the key problems of CSLR training is over fitting, and previous studies have shown that while an iterative training approach can partially address this problem, it requires additional training time. This work takes the iterative training approach used in his recent CSLR work and concludes that feature extractors need to be well trained to address the over fitting problem. Therefore, we propose Visual Alignment Constraint (VAC) to improve the feature extractor using alignment supervision. Specifically, the proposed VAC consists of two additional losses. In addition, two metrics are provided that highlight over fitting by evaluating the prediction discrepancy between the feature extraction module and the targeting module. Experimental results on two difficult CSLR datasets demonstrate that the proposed VAC delivers competitive performance and makes CSLR networks end-to-end trainable.

III. PROBLEM STATEMENT

The only way we can share our ideas or spread a message is through communication, but a person with a disability (such as someone who is deaf or mute) may find it difficult to communicate with a normal person. As a result, a person who has trouble hearing and speaking is unable to keep up with the rest of society. By sharing their ideas, opinions, and experiences with others around them, humans get to know one another. There are several methods to accomplish this, but the gift of "Speech" stands out

Journal of Advanced Research in Technology and Management Sciences
Volume: 05 Issue: 01 ISSN: 2582-3078 January 2023
Available online at: http://www.jartms.org

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1	class	x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4	x5	y5	z5	x6	y6	z6	x7	y7	z7	x8	y8
2	A	0.596074	0.990067	-3.81E-07	0.561661	0.954868	-0.0102	0.541067	0.903594	-0.0178	0.53001	0.860454	-0.02538	0.521192	0.822098	-0.03064	0.568175	0.836219	-0.00938	0.532086	0.834578	-0.02663	0.528777	0.886381
3	A	0.573347	1.009118	-3.88E-07	0.547576	0.973246	-0.00613	0.533219	0.912237	-0.00975	0.527215	0.863874	-0.01436	0.523727	0.824516	-0.01648	0.565868	0.835709	0.003294	0.53307	0.837982	-0.01121	0.53393	0.890058
4	A	0.586314	1.017193	-3.56E-07	0.552913	0.980016	-0.00521	0.536607	0.919895	-0.00856	0.53061	0.86852	-0.01336	0.525346	0.827872	-0.01559	0.565951	0.83855	0.006639	0.533517	0.835272	-0.00903	0.53294	0.888967
5	A	0.587999	1.024097	-3.72E-07	0.553208	0.979351	-0.00212	0.536545	0.917214	-0.00499	0.530152	0.866269	-0.0106	0.525536	0.823584	-0.0138	0.56692	0.840745	0.009916	0.533989	0.837084	-0.00656	0.533278	0.890346
6	A	0.58129	1.032066	-4.11E-07	0.546293	0.984322	-0.00255	0.531127	0.920594	-0.00448	0.527113	0.869513	-0.00886	0.525887	0.831621	-0.01079	0.565042	0.847901	0.009445	0.531689	0.844578	-0.00708	0.530653	0.900183
7	A	0.584838	1.032886	-3.86E-07	0.549703	0.985514	-0.00376	0.532343	0.923427	-0.00666	0.526185	0.872949	-0.01175	0.523229	0.833093	-0.01435	0.564157	0.851569	0.007484	0.5303	0.846611	-0.00944	0.528633	0.902305
8	A	0.584048	1.034338	-3.90E-07	0.547997	0.982759	0.001632	0.531024	0.921862	-0.00045	0.526368	0.871534	-0.00629	0.523292	0.83264	-0.0102	0.563577	0.852585	0.010215	0.529199	0.848056	-0.00663	0.527373	0.903531
9	A	0.569553	1.035871	-3.84E-07	0.539485	0.985894	-0.00619	0.528431	0.92232	-0.00957	0.52657	0.872777	-0.01423	0.52585	0.836622	-0.01641	0.56282	0.852896	0.003354	0.533575	0.850066	-0.01154	0.531218	0.904175
10	A	0.581663	1.034203	-3.55E-07	0.553638	0.989384	-0.00943	0.538373	0.92765	-0.01347	0.533462	0.875704	-0.01774	0.534657	0.836626	-0.01934	0.567918	0.856664	0.001027	0.541007	0.849832	-0.01336	0.537481	0.90214
11	A	0.580705	1.029114	-3.56E-07	0.553809	0.989682	-0.00947	0.538092	0.927269	-0.01346	0.532867	0.876138	-0.01766	0.535471	0.837744	-0.01922	0.566153	0.857421	0.000311	0.54098	0.851129	-0.01546	0.538349	0.902872
12	A	0.614943	0.945994	-4.51E-07	0.581493	0.907497	-0.00851	0.562368	0.826198	-0.01295	0.562437	0.765166	-0.01914	0.569999	0.719142	-0.02273	0.584806	0.765227	-0.00095	0.583534	0.716172	-0.02385	0.574581	0.773988
13	A	0.62989	0.783768	-3.27E-07	0.596095	0.755079	-0.01074	0.568779	0.684556	-0.01335	0.561621	0.620499	-0.01729	0.563094	0.569833	-0.01816	0.582904	0.627717	0.008235	0.578959	0.583311	-0.01048	0.579788	0.628574
14	A	0.64713	0.867259	2.49E-07	0.605731	0.802745	-0.00375	0.579145	0.734915	-0.00556	0.565825	0.675134	-0.00853	0.563804	0.623974	-0.01089	0.583467	0.721204	0.003667	0.574987	0.659261	-0.00615	0.573838	0.615752
15	A	0.619364	0.61154	-3.06E-07	0.586592	0.600681	-0.0142	0.555707	0.546369	-0.01856	0.542912	0.488737	-0.02269	0.542783	0.441424	-0.02305	0.567264	0.483027	0.001061	0.565466	0.438947	-0.01715	0.569687	0.485985
16	A	0.627074	0.618384	-2.81E-07	0.590698	0.60174	-0.01186	0.557727	0.548544	-0.01549	0.543322	0.492187	-0.01997	0.542185	0.444748	-0.02078	0.568514	0.48779	0.007058	0.562909	0.440611	-0.01065	0.568656	0.486315
17	A	0.62286	0.579646	-2.29E-07	0.583701	0.566436	-0.01239	0.547557	0.515023	-0.01494	0.530794	0.455544	-0.01837	0.530724	0.406029	-0.01802	0.557074	0.450037	0.011706	0.550108	0.403623	-0.00458	0.55551	0.450712
18	A	0.617298	0.570031	-3.00E-07	0.577944	0.562037	-0.01346	0.540746	0.510785	-0.01675	0.523252	0.453152	-0.02072	0.522293	0.405938	-0.02109	0.547409	0.443737	0.007666	0.542761	0.39698	-0.01079	0.550771	0.446963
19	A	0.61836	0.569966	-3.23E-07	0.579423	0.561843	-0.01247	0.542423	0.510438	-0.01501	0.524412	0.452698	-0.01838	0.52239	0.405905	-0.01801	0.547959	0.444219	0.008669	0.542873	0.397694	-0.00875	0.550893	0.445829
20	A	0.610208	0.568118	-3.22E-07	0.571757	0.56324	-0.01517	0.53368	0.512857	-0.01896	0.514583	0.454038	-0.02315	0.514359	0.407019	-0.02327	0.538628	0.44576	0.005587	0.533647	0.397242	-0.01333	0.542905	0.448886
21	A	0.609318	0.569215	-3.21E-07	0.570668	0.561346	-0.01377	0.532894	0.513088	-0.01725	0.513943	0.453698	-0.02131	0.513191	0.404904	-0.02135	0.538609	0.449475	0.006585	0.531206	0.399796	-0.0119	0.540686	0.448131
22	A	0.609437	0.569501	-3.32E-07	0.57072	0.56358	-0.01402	0.532174	0.514495	-0.01747	0.51343	0.454206	-0.02135	0.512484	0.405653	-0.02117	0.537781	0.449261	0.005661	0.531133	0.398569	-0.01311	0.540356	0.448124
23	A	0.610992	0.566003	-3.53E-07	0.570749	0.561885	-0.01262	0.532733	0.511391	-0.01554	0.51541	0.448039	-0.01919	0.51578	0.399738	-0.01901	0.540235	0.446356	0.006518	0.533727	0.39525	-0.01207	0.542064	0.445516
24	A	0.614469	0.567635	-3.49E-07	0.574395	0.564018	-0.01362	0.537494	0.511859	-0.01712	0.52076	0.448247	-0.02118	0.519872	0.398472	-0.02126	0.543489	0.447149	0.006097	0.536752	0.39621	-0.01322	0.545826	0.44695
25	A	0.614094	0.569828	-3.38E-07	0.574876	0.560856	-0.01281	0.538009	0.510218	-0.01576	0.520034	0.449666	-0.01951	0.519251	0.402254	-0.01941	0.544761	0.445215	0.007654	0.538125	0.397165	-0.01038	0.546403	0.447198
26	A	0.614689	0.568808	-3.34E-07	0.57569	0.562732	-0.0134	0.538097	0.512665	-0.01652	0.519722	0.452388	-0.02017	0.518126	0.4054	-0.01988	0.543932	0.448082	0.005781	0.53762	0.397145	-0.01254	0.546513	0.446742
27	A	0.615989	0.571928	-3.37E-07	0.576722	0.562695	-0.01154	0.540571	0.51285	-0.01436	0.523319	0.454602	-0.01833	0.521638	0.407123	-0.01863	0.547387	0.446094	0.008339	0.539677	0.398172	-0.00968	0.547708	0.446164
28	A	0.623048	0.56521	-3.16E-07	0.582009	0.552141	-0.00969	0.547794	0.504205	-0.01245	0.53137	0.446784	-0.01673	0.530594	0.398668	-0.01741	0.554971	0.4388	0.010799	0.549471	0.391158	-0.0061	0.557492	0.434504
29	A	0.631343	0.552543	-3.06E-07	0.588352	0.53824	-0.00754	0.555587	0.489861	-0.00887	0.541832	0.43265	-0.01222	0.540926	0.385889	-0.01244	0.563749	0.427185	0.013795	0.55891	0.379799	-0.00216	0.566387	0.422749
30	A	0.632321	0.550545	-2.97E-07	0.590075	0.53709	-0.00682	0.556325	0.488376	-0.00802	0.542317	0.433276	-0.01121	0.54085	0.388282	-0.01139	0.564252	0.427294	0.012111	0.55782	0.380571	-0.00351	0.565983	0.422679
31	A	0.658487	0.519459	-3.01E-07	0.613784	0.496619	-0.00738	0.578224	0.450413	-0.00872	0.563387	0.399853	-0.01236	0.564742	0.354812	-0.01318	0.588569	0.392379	0.013904	0.582564	0.346635	-0.00227	0.591648	0.388857
32	A	0.681522	0.504812	-3.39E-07	0.639496	0.481368	-0.00846	0.606716	0.428296	-0.0095	0.592686	0.37477	-0.01281	0.594883	0.331006	-0.01281	0.613337	0.369799	0.016061	0.608641	0.324784	-0.00198	0.618224	0.371539
33	A	0.692161	0.495202	-3.18E-07	0.652797	0.471434	-0.00799	0.62089	0.425503	-0.01029	0.605123	0.373295	-0.01481	0.605815	0.327822	-0.0159	0.625751	0.366244	0.014547	0.621268	0.319454	-0.00341	0.630652	0.36366
34	A	0.691593	0.491597	-2.82E-07	0.651993	0.467551	-0.0086	0.61992	0.424212	-0.01132	0.603801	0.373877	-0.01614	0.605905	0.330265	-0.01767	0.626535	0.364609	0.014004	0.621191	0.319025	-0.00372	0.630035	0.361711
35	A	0.707735	0.492255	-3.12E-07	0.668896	0.469602	-0.00922	0.637129	0.424688	-0.01239	0.622301	0.371469	-0.01745	0.625635	0.325929	-0.01922	0.645586	0.367118	0.01209	0.643468	0.319937	-0.0062	0.649909	0.36093
36	A	0.716086	0.487829	-3.01E-07	0.678133	0.469242	-0.01083	0.647197	0.425817	-0.0143	0.633454	0.369939	-0.019	0.634221	0.323699	-0.02002	0.655307	0.361077	0.012002	0.652861	0.316659	-0.00544	0.660199	0.361311
37	A	0.719279	0.489729	-2.97E-07	0.682965	0.475016	-0.01339	0.650725	0.431986	-0.01824	0.635884	0.374072	-0.02378	0.63785	0.325372	-0.02547	0.661151	0.366398	0.0081	0.657721	0.315995	-0.01104	0.664228	0.359127
38	A	0.720041	0.492845	-2.87E-07	0.683054	0.476597	-0.01219	0.65067	0.432775	-0.01641	0.635567	0.374265	-0.02156	0.639157	0.326299	-0.02295	0.661071	0.370152	0.008661	0.657931	0.3189	-0.01013	0.664227	0.361065
39	A	0.718467	0.494839	-2.87E-07	0.681387	0.479954	-0.0134	0.648731	0.436071	-0.01816	0.633565	0.378826	-0.02363	0.636546	0.331296	-0.02533	0.657613	0.371665	0.00673	0.654137	0.32043	-0.01253	0.662368	0.364554

Fig. 4 A sample of the dataset created

among the rest as the best one. The usage of "Sign Language" is the only form of communication available to the deaf and mute. For such communications, interpreters are both expensive and scarce. The concept of using pens and paper is cumbersome and unpleasant. As a result, there is a demand for automated technology that makes it simple to communicate with someone who is disabled (PwDs).

IV. METHODOLOGY

The workflow of the project is divided into 4 major steps namely.

A. Image Acquisition

This part contains, capturing image frames and loading the coordinates into an CSV file. A person sits and performs the sign of all the alphabets, each alphabet sign is captured in different positions and different areas of the camera view. An approximate of 800 frames is captured for each alphabet and loaded into the CSV file (Fig 4). To locate accurate key point localization of 21-3D hand-knuckle coordinates inside the observed hand areas, here the MediaPipe hands library's hand landmark is utilised, the hand landmark model uses regression, or direct coordinate prediction,

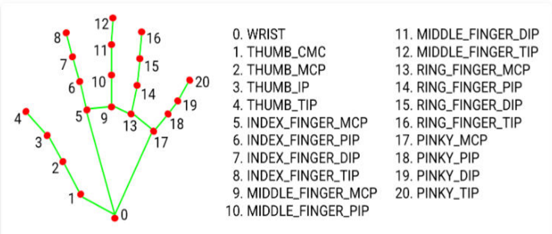

Fig. 5 Hand Landmark Model

The steps involved in this are:

1. Initially a CSV sheet is created and the first row is the class name and the coordinates x0, y0,z0,x1… are labelled.
2. Then for each alphabet class name is entered and an approximate of 850 frames are recorded manually and appended to the excel sheet.
3. For each alphabet the recorded values are stored in NumPy arrays [x0,y0,z0]… [x21,y21,z21], which is then converted into a single array using the flatten function and inserted as single row into the CSV sheet.
4. An approximate of One Hundred Thousand values are collected in the CSV sheer for alphabets A to Z.

B. Pre-Processing

In this segment the collected coordinates are processed and named as different classes i.e., the coordinates are given class names A, B, C and so on. Any noise or NaN values are processed and the CSV file is made ready for the model training.

C. Data Recognition

There is an approximate of One Hundred Thousand frames recorded for all the alphabets. This data is divided into 70% training data and 30% test data. Here in this segment, we choose the appropriate model and train it with the values and create an .pkl file which contains the class names and the training data. Here a python file named model.py is created where all the required algorithms are present.

The steps involved in this are:

1. The recorded coordinates are divided into 70% training data and 30% as test data.

2. We select the desired algorithm and then pass the training data and algorithm to the function.
3. A pickle file is created with the required algorithm name and this file is passed to the final code to get the prediction.

D. Classification/Output

In this segment a different user is made to perform the sign and the model analyses the sign and prints the output on the screen. The .pkl file is loaded along with the new coordinates to the model and the prediction is done to output the alphabet. Here in the predict stage a file named predict.py is created where the pickle file is sent as an input.

The steps involved in this are:

1. In this predict.py file the user is prompted to show the sign and the recognised alphabet is displayed on the screen.
2. The pickle file is sent as input to the inbuilt function which then returns the outputs which is the class name of the alphabet.

V. SYSTEM SPECIFICATIONS AND DESIGN

The required specifications are divided into two parts namely hardware requirements and software requirements:

A. Software Requirements and Libraries used

1. IDE – PyCharm Community Edition 2021.2.3
2. Coding Language – Python 3.9
3. List of libraries used:
   a. CSV
   b. CV2
   c. MediaPipe
   d. NumPy
   e. Pandas
   f. Pickle
   g. Scikit-Learn

B. Hardware Requirements

These are requirements recommended to run the code smoothly and without any interruptions.

1. Processor – Inteli7-9750HF or higher versions
2. Graphic Card – GTX 1650(4 gb) or higher
3. RAM – 8GB minimum
4. Hard Drive – 512GB minimum
5. Digital Camera or a webcam

C. Design

The design phase of the proposed project is divided into three different parts namely the dataset loading, model training and prediction phase. The following are the flow charts of the 3 phases –

1) *The dataset creation*

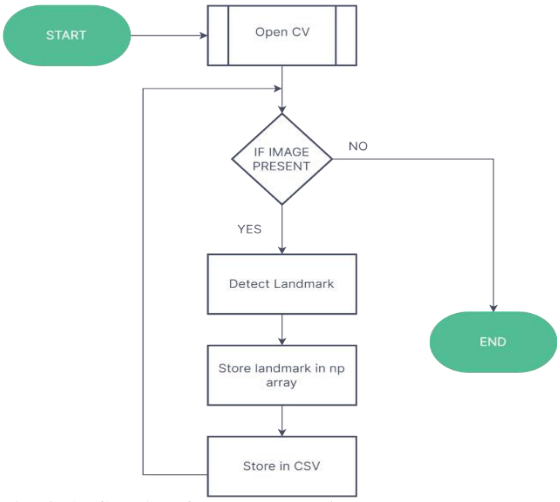

Fig. 6 The flowchart for Dataset Creation

2) *Model Training*

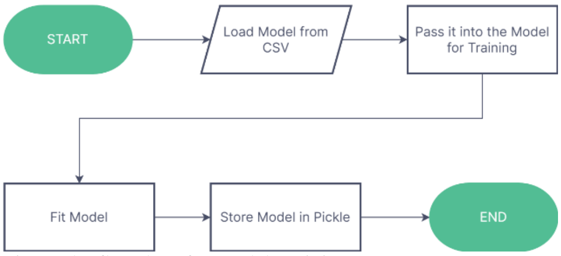

Fig. 7 The flowchart for Model Training

3) *Prediction analysis*

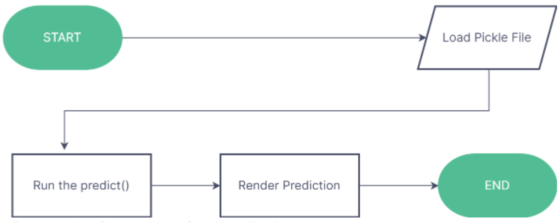

Fig. 8 The flowchart for prediction

VI. RESULTS

This project will be able to detect and recognize all the static signs available in Indian Sign Language (ISL). The dataset for these is already loaded into the required model which is well trained, tested and ready-to-use. The user while using this software, will have to make a static-ISL sign into the pop-up OpenCV image capture window. When this is done, the program recognizes all the hand landmarks present on the screen. The co-ordinates of these landmarks are then passed into the program that converts this raw data into the format required for the ML model. This new data is then passed into the model for prediction and the predicted sign is then rendered back onto the OpenCV window as user output. On testing several times and under various different conditions, it is concluded that the model is functioning very well with a prediction accuracy of almost seventy percent. The figures below try to illustrate the usage and output of this software for an average user.

Fig. 9 B Alphabet Prediction

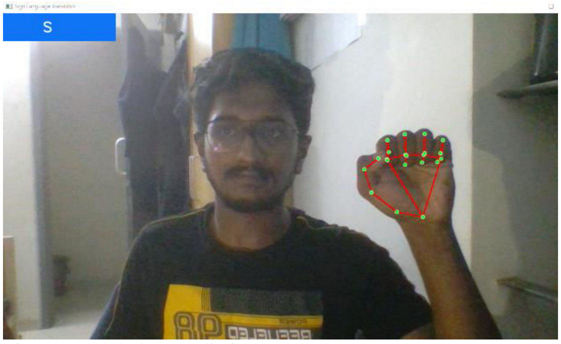

Fig. 10 S Alphabet Prediction

Fig. 11 X alphabet Prediction

Fig. 12 K alphabet Prediction

Here, the figures shows the entire working of this software. The detected landmarks, are displayed on the output image along with the pose connections. This pose is recognized and the predicted output is then displayed on the top left of the output dialog box.

VI. CONCLUSION

The suggested project does static, or single frame, sign recognition in ISL quite effectively. The fact that it cannot be used for dynamic and continuous indications is its main shortcoming. In the process of further developing this technology, we hope to address this issue. Each technique also has restrictions when compared to other techniques. One of the non-verbal communication techniques utilised in sign language is the hand gesture. It is mostly used by deaf and mute individuals to communicate with other people or among themselves when they have hearing or speech issues. Many manufacturers around the world have developed various sign language systems, but they are neither customizable nor cost effective for end users. Therefore, this paper presents a prototype system that automatically recognizes sign language to enable deaf and mute people to communicate more effectively with each other and with the general public. Pattern recognition and gesture recognition are developing research areas.

The two emerging topics of study are pattern recognition and gesture recognition. With the aid of a hand gesture detection system, we have access to a novel, comfortable, and user-friendly method of interacting with computers that is more suited to human needs. We believe this project could play a significant role in uniting local communities, particularly in schools, training where there is a sizeable and vibrant Deaf and Mute populations.

VII. FUTURE SCOPE

Sign Languages, like any other modes of communication involve several types of parameters (signs). This includes both static as well as continuous gestures. Both of these are required for effective communication. When training a computer to essentially learn sign language, it requires only single frame data to recognize static signs. However, to recognize a dynamic sign or gesture, the computer has to be programmed to consider multiple frames and recognize all of them as a single input. Additionally, the same machine learning used for static signs cannot be implemented for continuous signs as the dataset varies in format and structure. Famous deep learning models under RNN (Recurrent Neural Networks) are commonly used for solving such types of complications. Hence, implementing a variant of this model on the current project might solve the problem.

On further development, this project and its fundamental concepts can be implemented for the recognition of dynamic as well as static signs. On successful implementation of this feature into this project, it would become a device capable of translating all gestures used in Indian Sign Language into English effortlessly. In turn, making this device a very useful tool for integration and empowerment of PwDs into common society.

REFERENCES

[1] Prof. Radha S. Shirbhate, Mr, Vedant, D. Shinde, *"Sign language Recognition Using Machine Learning Algorithm"*, 2020
[2] Amit Moryossef, Loannis Tsochantaridis, Roee Aharoni, "*RealTime Sign language Detection using pose assessment*", 2020

[3] Aditya Das, Shantanu Gawde, Khyati Suratwala, "*Sign Language Recognition Using Deep Learning on Custom-Processed Static Guest Images"*, 2018

[4] Dongxu Li, Cristian Rodriguez Opazo, Xin Yu , Hongdong Li, *"World-Level Deep Sign Language Recognition from Video",* 2019

[5] Byeongkeun Kang, Subarna Tripathi, Truong Q. Nguyen, *"Real-time Sign Language Fingerspell Recognition Using Convolutional Neural Networks*", 2015

[6]] Christian Zimmermann, Thomas Brox, *"Learning to estimate 3D hand poses from a single RGB image*," 2017

[7] Ashish S. Nikam; Aarti G. Ambekar, *"Sign Language Recognition Using Image-Based Hand Gesture Recognition Techniques*", 2020

[8] Yuecong Min, Aiming Hao, Xiujuan Chai, Xilin Chen, "*Visual Placement Constraints for Continuous Sign Language Recognition"* , 2021