2023 IEEE 8th International Conference for Convergence in Technology (I2CT)
Pune, India. Apr 7-9, 2023

American Sign Language Fingerspelling Recognition using Attention Model

Amruta E Kabade
KLE Technological University
Hubli-580031
amruta.e.kabade@gmail.com

Padmashree Desai
KLE Technological University
Hubli-580031
padmashri@kletech.ac.in

Sujatha C
KLE Technological University
Hubli-580031
sujata_c@kletech.ac.in

Shankar G
KLE Technological University
Hubli-580031
shankar@kletech.ac.in

2023 IEEE 8th International Conference for Convergence in Technology (I2CT) | 979-8-3503-3401-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/I2CT57861.2023.10126277

Abstract—**Sign Language Recognition(SLR) is a complex gesture recognition problem because of the quick and highly co-articulated motion involved in gestures. This research work focuses on Fingerspelling recognition task, which constitutes 35% of the American Sign Language (ASL). Fingerspelling identifies the word letter by letter. Fingerspelling is used for signing the words which do not have designated ASL signs such as technical terms, content words and proper nouns. In our proposed work for ASL Fingerspelling recognition, we consider ChicagoFSWild dataset which consists of occlusions and images captured in varying illuminations, lighting conditions (in the wild environments). The optical flow is obtained from Lucas-Kanade algorithm, prior is generated, images are resized and cropped with face-roi technique to get the region of interest (ROI). The visual attention mechanism attends to the ROI iteratively. ResNet, pretrained on Imagenet is used for the extraction of spatial features. The Bi-LSTM network with Connectionist Temporal Classification (CTC) is used to predict the sign. It provides the accuracy of 57% on ChicagoFSWild dataset for Fingerspelling recognition task.**

Index Terms—**ASL, Fingerspelling, Optical Flow, Lucas-Kanade, Attention, DNN, ResNet, Bi-LSTM**

I. Introduction

In this paper, we propose a framework for American Sign Language(ASL) Fingerspelling Recognition. Automatic sign language recognition can potentially overcome the communication gap between hearing-impaired and others. To both hearing-impaired and normal people, sign language benefits with enhanced spatial reasoning, peripheral vision, reaction time and long-term cognition. Therefore, sign language is not just for deaf or hard of hearing individuals. Unfortunately, not all hearing impaired people have the means to learn ASL. In the US, ASL is ranked as the fourth most common language.

Sign Language Recognition presents a variety of difficulties. Sign consists of highly co-articulated and quick arm movements, facial expressions, and hand gestures. There exists a lot of difference in the way various signers' hands and bodies look. For sign language, there is substantially less annotated data.

In fingerspelling, signers spell out a word, letter by letter. It involves signing sequence of individual letters in the word. The finger spelled hand shapes are used through out ASL. Fingerspelling is constrained but holds prominent role in ASL, accounting for up to 35% of ASL. Typically, fingerspelling is used for signs that do not have a designated signs. Fingerspelling can be used for content words, technical words, proper nouns or new coinages. Single hand is used for ASL fingerspelling but it involves small, quick and highly co-articulated motion of fingers and hand.

In our approach, we first preprocess the ChicagoFSWild dataset and find the optical flow features to identify the motion as it is most likely to be the signing hand in motion. We then, get the prior, generate a clear preprocessed frame from optical flow features. ResNet backbone architecture is used as spatial feature extractor. The features obtained with the earlier time step RNN output is transmitted to the attention module to take into consideration both spatial and temporal features for the training process. The Connectionist Temporal Classification (CTC) loss is calculated as there could be no alignment between frames and letters. The language model is used for beam search to identify the most probable next letter in the fingerspelling sequence. The DNN model performs well in vivid conditions.

II. LITERATURE SURVEY

This section examines research in the field of sign language recognition (SLR). In earlier studies, the signs were recognised from RGB photos using hand-crafted characteristics. Scale Invariant Feature Transform (SIFT) [24], [30] features used with Clustering [8] or Classification techniques to identify signs. The works that employ hand-crafted features on static hand poses are based on HOG-based features, SIFT-based features [8], [30] and frequency domain features to extract spatial features. Later, the temporal dependency in video sequences is obtained using Hidden Markov Models(HMM).

The automatic recognition of many sign languages has been the subject of extensive research. Early work on SLR from video focused on isolated signs [2], [7]. Recent works focus on datasets and continuous sign language recognition [9], [10], [17], [18]. Most of the work is on the Video corpora for multiple sign languages [9]–[11], which is all recorded in a studio environment. This induces lower variability in the dataset.

Convolutional Neural Networks (CNNs) have been used in studies on the recognition of sign language [8], [11], [17]–[19]. Sequence modelling has used Hidden Markov Models (HMMs) [13], [17], [18], [20]. The majority of earlier research used frame-level annotation as the basis for the training data, whereas more current efforts concentrate on learning just from sequence labels due to the difficulty in acquiring the annotation [11], [17], [19]. Authors in papers [22], [25], [26], [28] discuss different deep learning models for video synopsis, frame prediction, person tracking, and classification problem, which can assist in other video processing applications in the field of computer vision.

Recent Deep Learning(DL) based efforts can be divided into two categories: end-to-end systems vs. two-stage techniques. In a two-stage method, the hand landmarks are first retrieved using a pose-network or a hand detector, which yields the hand bounding box. In [1], To first identify the signing hand and subsequently the fingerspelling sign, the authors suggest using Faster R-CNN. To add hand landmark points to a Gated Recurrent Unit (GRU) network and Graph Convolutional Network in [23], the authors employed OpenPose (GCN) [5], [15]. However, they require additional annotations in the dataset for better performance. The End-to-end systems use a single network to detect sign from RGB video stream. There is no need of additional supervision. A module for sequence-to-sequence prediction is directly passed 2D or 3D CNN features [6]: Connectionist Temporal Classification based networks, encoder-decoder networks [4], Hidden Markov Models (HMMs) [12] etc.

Most of the previous work on fingerspelling recognition approaches involve extraction of the signing hand from the image frames [11], [13], [21]. Spatial attention mechanism is applied in vision tasks i.e. image recognition [8], [14], [16], [27] and image captioning [15]. In our approach, the optical flow features are extracted in the preprocessing step to recognize the motion, as it is most likely to be the signing hand and iterative visual attention mechanism [3] is used to attend to the ROI.

The Challenges identified from literature survey are as follows:

- Finger-spelling includes rapid, highly coordinated movements that are frequently difficult to differentiate from one another.
- The way hands are actually articulated in continual signing in the real world differs greatly from canonical hand shapes.
- The presence of complex backgrounds and low fps and resolutions in real-time applications making the recognition task difficult.
- The system's accuracy is impacted by the wild dataset.

III. Proposed Framework

The methodology for SLR on ChicagoFSWild dataset is proposed by [3]. It is a complete system, which uses iterative approach of visual attention mechanism to attend to the area of interest by zooming in iteratively. The ROI obtained surrounding near to the signer's face in the frames. The approach iteratively works on the ROI using the attention mechanism to narrow down the informative region. CNN is employed for the spatial characteristics and the sequential features considering temporal characteristics are obtained using Recurrent Neural Network (RNN). These spatial and temporal features along with the optical flow are passed as input to the attention network. The optical flow feature vectors capture the movement. The iterative behaviour of updating the attention map provide significant improvement in inference time.

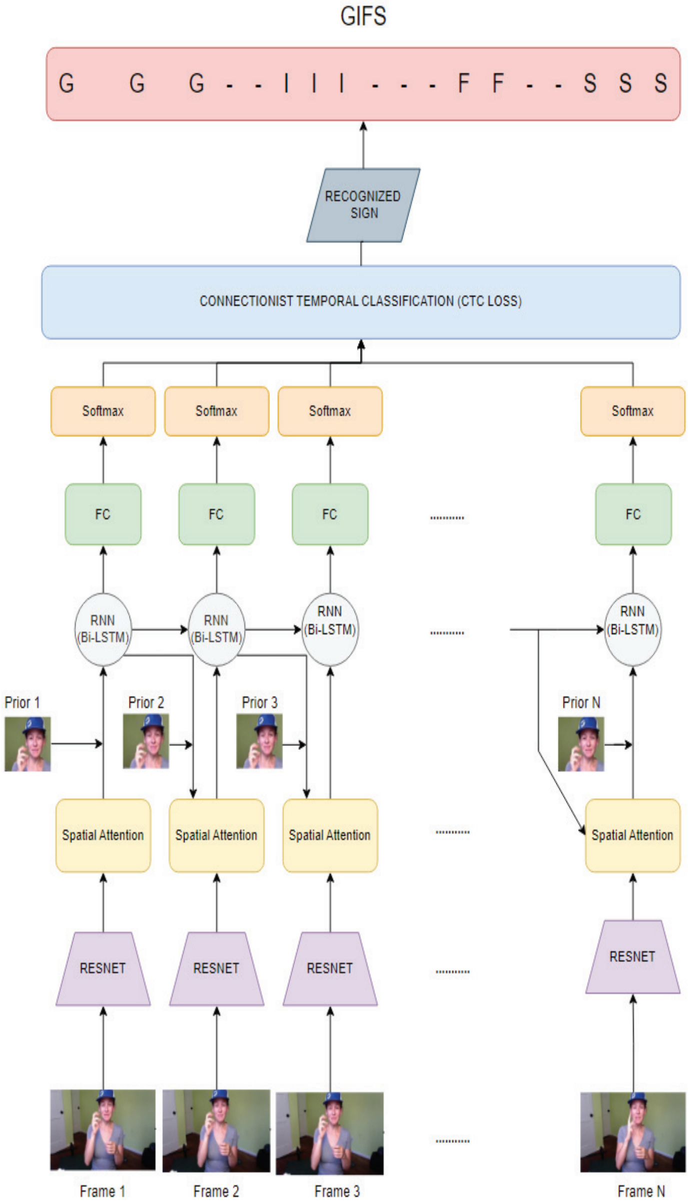

Fig. 1. Proposed framework for ASL Fingerspelling recognition

The figure 1 illustrates the proposed framework for the ASL fingerspelling recognition For face detection, we employ the implementation, which was tuned using the WIDER data set.

We only execute the face detector once every five frames to reduce computation time.

Initial frame processing involves FaceROI which crops the image frame region 3 times larger, centered towards the face-detection box. Each and every input frame is scaled to a maximum of 224 pixels. When more than one face is found, the face tubes are connected based on the degree of connecting box overlap in succeeding frames.

Our model's convolutional layers are built using ResNet101 that has been pre-trained on ImageNet. We use a Bi-LSTM network for the RNN. The average of the optical flow features for time-instants t, t-1, and t+1 is used as the prior map for time instant t.

The input to the spatial attention network is a sequence of image frames $I_1, I_2, I_3, ..., I_k$. The output is a word fingerspelled in a series of letters i.e. $W_1, W_2, W_3, W_4..., W_k$. There exists no frame-level alignment. So $k << T$ as there can be several frames representing the same letter. Built on convolution recurrent framework, the attention system. A CNN is applied to image frame I_t at any frame t to get the feature space f_t. Let's assume that e_{t-1} is the hidden state of the recurrent unit at time instant t-1. The hidden state e_{t-1} and feature f_t are used to calculate the attention map β_t.

β_t identifies the prominent characteristics at various spatial locations in the frame corresponding to the letter sequence. Consider i and j denote the index spatial locations. The equations used for the attention map are described below:

$$v_{tij} = {u_f}^t tanh(W_d e_{t-1} + W_f f_{tij}) \tag{1}$$

$$\beta_{tij} = \frac{exp(v_{tij})}{\sigma exp(v_{tij})} \tag{2}$$

The weighted average of features f_{tij} is the visual feature vector at time step t.

$$A_t = \frac{\beta_t \odot {M_t}^\alpha}{\sigma \beta {M_t}^\alpha} \tag{3}$$

$$h_t = \sigma f_{tij} A_{tij} \tag{4}$$

The M denotes prior knowledge about the prominent and relevant spatial locations. A defines the subsequent attention map while α determines the relative weight of the attention weights learned and that of the prior obtained from optical flow features.

The recurrent unit state for the time-instant t is revised by $e_t = BiLSTM(e_{t-1}, h^t)$. A letter sequence is generated by decoding the high level characteristic sequence from the image frames.

Using greedy approach, we can create a complete frame-level label series during test time by selecting the label with the maximum likelihood for each frame. Ultimately, label collapsing method is used to first remove duplicate frame labels before blanking, the label series $w = a_1, a_2, a_3, ..., a_K$ is generated from the frame-level series.

In order to optimise ultimate label sequence 's log probability during training, CTC uses a forward-backward algorithm to add up all compatible frame-level labels.

IV. Motion Estimation

In this section, we discuss motion estimation strategies, such as the comparison of Lucas Kanade with Farneback Algorithm for obtaining optical flow properties. A method used to explain the motion in a picture is optical flow. It is used on images with a short temporal gap like video frames. Based on the current position and velocity of the points inside the image, optical flow provides an estimation of where the points may be in the following frame sequence.

A. Optical Flow

Optical flow is the structure of seemingly shifting contours, planes, and entities in a digital representation that is caused by the movement of the observer relative to the field. A further approach to optical flow is as the spread of the brightness pattern's apparent motion velocities. There are different techniques available for motion estimation based on phase correlation, differential techniques, block based approaches and discrete optimization techniques.

1) Farneback Method: OpenCV provides the Farneback algorithm that provides the dense optical flow characteristics. It finds the motion feature for every point space in the frame space. The dense optical flow feature is built on Gunnar Farneback's algorithm proposed in [15]. It calculates the flow vector for each pixel in the frame. So, it may provide a better result but requires more time, because of it's slow speed. The technique uses quadratic polynomial to approximate the image frames. Observing the polynomial transforms to estimate displacement fields. After a series of refinements, dense optical flow is computed.

2) Lucas-Kanade Method: The lucas-kanade approach anticipates that adjacent pixels would move similarly. Around the point, it considers a 3x3 patch. With strong movements, it could break. We employ pyramids to address this. The large movements become minor as we move up the pyramid. The optical flow and scale are obtained by using the lucas-kanade method.

In our proposed methodology, we create a simple application to track the interesting points in the video. From the first frame, we identify the hand gesture points, which is the region of interest and iteratively track the ROI using the flow vectors obtained with Kanade methodology. As the tracking of motion is only with respect to the ROI, it iteratively focuses on those points and tracks the motion, providing the feature vector with relevant features of hand motion instead of providing the motion of all the pixels in the frame.

V. Extraction of Spatial and Temporal features

In this section, we discuss the proposed framework for extraction of spatial and temporal features. we discuss the architecture of ResNet101 and it's performance as a backbone extractor for extracting the spatial features in fingerspelling recognition task in comparison with AlexNet architecture. Deep Learning integrates the features extraction and classification modules into one integrated system. Based on the data and extracted features, it learns the discriminating representational features from the image frames. One such system is the Deep Neural Network(DNN). The multilayer perceptrons consists of densely connected multiple layers of neurons.

A. Residual Networks

Residual neural networks are a part of artificial neural networks. It is a gateless HighwayNet gateless variant [2], operationally functional very deep feedforward NN consisting of multiple layers(Hundreds), far deeper than preceding NNs. Use skips to bypass some layers. To build ResNet networks, batch normalisation is typically applied in the middle of two or three layer shortcuts, including Rectified linear function. A model type that uses numerous concurrent skips is called DenseNet [3]. In the area of ResNet, a simple network can be known as a non-residual network.

AlexNet won the ImageNet 2012 challenge. After which every other winning architecture increases the number of layers in a DNN to lower the error rate. This is effective for fewer layers, but as the number of layers rises, the vanishing gradient problem occurs. As a result of this the gradient becomes very large or 0. As the layer count rises, so do the test and training error rates. A 20-layered CNN design outperforms a 56-layer CNN during both testing as well as training. The idea that it is driven by Gradients that are vanishing or exploding comes from further analysis of error rate.

Firstly, increasing network depth can provide improved accuracy but it leads to the problem of vanishing gradient. Secondly, while learning the deep networks, the optimizing process on large number of parameters leads to degradation problem and higher training error. The residual networks overcome the aforementioned problems, by allowing the learning of DNN such that the network is constructed through residual models. It uses global average pooling. It achieves better accuracy and lesser training error than AlexNet while being computationally more efficient for the Fingerspelling task on ChicagoFSWild dataset.

B. Bi-LSTM

Long Short Term Memory (LSTM) is the term for the artificial RNN's with memory. The LSTM-based models have more "gates" in order to remember longer sequences of data. Bidirectional LSTMs (BiLSTMs) allow for further training in both forward and backward directions, further enhancing LSTM prediction. BiLSTM performs better than LSTM models owing to this added training capabilities [29].

The two directional neurons of BRNNs do not interact, making it possible to train them using RNN-like procedures. Updating input and output layers simultaneously is not possible when back-propagation is used, hence additional procedures are required. The following general training techniques apply: In the case of a forward pass, output neurons come after forward and backward states have been transferred. Forward states and backward states are passed after passing output neurons for the backward pass. The weights are updated after the forward and backward passes.

The model's performance on sequence classification tasks can be boosted by using a B-i LSTM, which is an enhancement of regular LSTMs. When every timestep of the data set pattern is available, bidirectional LSTMs train dual LSTM networks instead of one. One on input series and other on reverse series of input. This can provide the network with additional context and result in a problem-learning process that is both fast and complete. This enhances the understanding of the context better to recognize the sign more accurately.

VI. Results and Discussion

In this section, we discuss the dataset,performance metrics and comparison of results of the proposed method with [3]. We report the results on the evaluation set of ChicagoFSWild dataset.

A. Dataset Description

The Chicago Fingerspelling in the wild (ChicagoFSWild) dataset proposed by [3] is used for model training and testing.The dataset consists collection of ASL videos with fingerspelling obtained from deafvideo.tv, aslized.org and YouTube. The variety of viewpoints and styles is included in the dataset. There is no frame-level labeling done to the dataset. Only sequence level labels have been used. The fingerspelled data means the words are signed letter by letter. Proper nouns, content terms, abbreviations and other technical words without corresponding ASL signs are among the data segments that are fingerspelled. The dataset consists of 7304 fingerspelling sequences divided into 5455 training, 981 development and 868 test sequences. There is no overlap in the 168 signers who signed the signs in the three sets, making the signs unique. The dataset is challenging and complex, being closer to real-world scenarios. The vast majority of the dataset videos weren't made in well orchestrated studio environments.

The figure 2 illustrates photos of fingerspelling in studio data as opposed to in natural settings. The leftmost frame in the figure depicts the frame from ChicagoFSVid studio dataset and the remaining frames are from the ChicagoFSWild dataset.

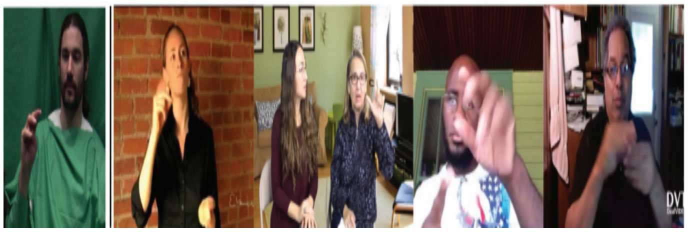

Fig. 2. Studio v/s wild environment dataset

B. Performance Metrics

The letter accuracy is used as a Performance metric given in percentage. The letter accuracy is computed by minimum hamming distance alignment between the ground truth and hypothesized letter sequences. The letter accuracy is computed using the following formula:

$$LA = 1 - (X + Y + Z)/N \qquad (5)$$

TABLE I
DATASET DESCRIPTION

Dataset	Split	# of videos	# of signers
FSWild dataset	Train	5455	87
	Validation	981	37
	Test	868	36
FSWild+ dataset	Train	50402	216
	Validation	3115	22
	Test	1715	22

where LA stands for Letter Accuracy, X represents Substitutions, Y indicates Deletions, Z denotes Insertions and N stands for number of ground truth letters.

C. Results

The Results of the optical flow and prior generation is shown below. The figure 3 shows the input image, The figure 4 represents the result of optical flow. It has identified the motion of signer's hand and subtracted the other background variations, however there exists still a little of background noise. The background noise is eliminated, during the prior generation. The clear image indicating only the signers hand being zoomed is shown in the figure 5.

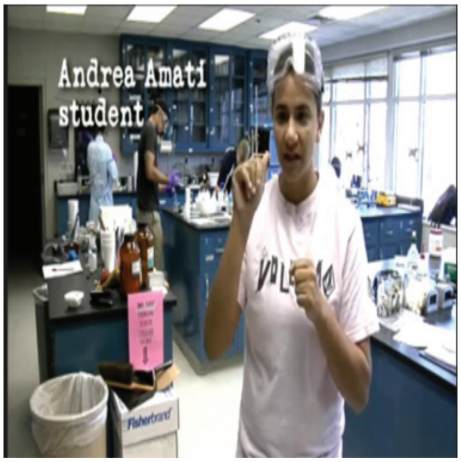

Fig. 3. Input Image

TABLE II
COMPARATIVE RESULTS

Approaches(Optical Flow+ConvNet+RNN)	Letter Accuracy(%)
State-of-the-art model [3](Farneback+AlexNet+LSTM)	42.674%
Our Approach 1(Lucas-Kanade+ResNet+LSTM)	48.895%
Our Approach 2(Lucas-Kanade+ResNet+Bi-LSTM)	57.840%

The comparison of the results with the model proposed in [3] for fingerspelling task is provided in the above table. It is observed that the accuracy is low and it is obvious, because the dataset ChicagoFSWild is collected in the wild environments and not in a studio or lab-environments. Because the dataset is full of varying conditions in illumination, occlusions, the performance is low. There is future scope of improvement for the research work on Fingerspelling task, by training it on more robust dataset such as ChicagoFSWild+.

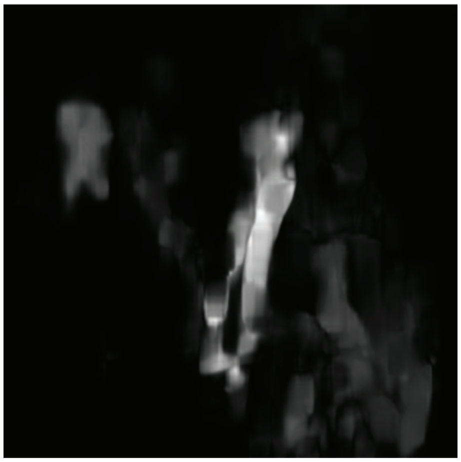

Fig. 4. Optical flow Output

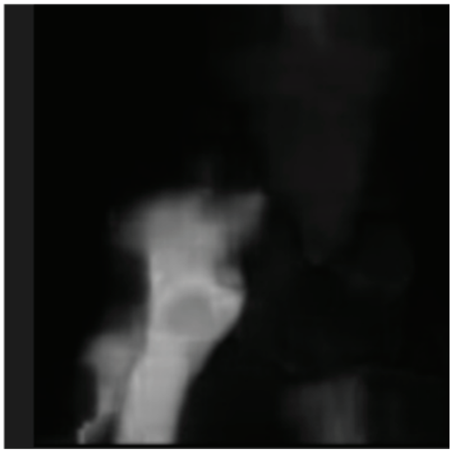

Fig. 5. Prior

VII. CONCLUSION

Our work focused on the ChicagoFSWild dataset collected in vivid environments and is near to real-world scenarios. Though the dataset is challenging with occlusions, the proper pre-processing to extract optical flow features and the use of Recurrent CNN with attention mechanism and bidirectional LSTM, the performance is enhanced significantly in comparison model proposed by [3] for the fingerspelling task on ChicagoFSWild dataset.

As a future scope of study, the model can be trained and tested on more robust datasets like the crowd sourced ChicagoFSWild+ which contains 50,402 training sequences performed by 216 signers.

REFERENCES

[1] Shi, Bowen, Aurora Martinez Del Rio, Jonathan Keane, Jonathan Michaux, Diane Brentari, Gregory Shakhnarovich and Karen Livescu. "American Sign Language Fingerspelling Recognition in the Wild." 2018 IEEE Spoken Language Technology Workshop (SLT) (2018): 145-152.

[2] V. Athitsos et al., "The American Sign Language Lexicon Video Dataset," 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, 2008, pp. 1-8, doi: 10.1109/CVPRW.2008.4563181.

[3] Shi, Bowen, Aurora Martinez Del Rio, Jonathan Keane, Diane Brentari, Greg Shakhnarovich and Karen Livescu. “Fingerspelling Recognition in the Wild With Iterative Visual Attention.” 2019 IEEE/CVF International Conference on Computer Vision (ICCV) (2019): 5399-5408.

[4] Sutskever, Ilya, Oriol Vinyals and Quoc V. Le. “Sequence to Sequence Learning with Neural Networks.” NIPS (2014).

[5] Cao, Zhe, Gines Hidalgo, Tomas Simon, Shih-En Wei and Yaser Sheikh. “OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields.” IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (2018): 172-186.

[6] Carreira, João and Andrew Zisserman. “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset.” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017): 4724-4733.

[7] N. C. Camgoz, S. Hadfield, O. Koller, H. Ney and R. Bowden, ”Neural Sign Language Translation,” 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 7784-7793, doi: 10.1109/CVPR.2018.00812.

[8] F. Yasir, P. W. C. Prasad, A. Alsadoon and A. Elchouemi, ”SIFT based approach on Bangla sign language recognition,” 2015 IEEE 8th International Workshop on Computational Intelligence and Applications (IWCIA), 2015, pp. 35-39, doi: 10.1109/IWCIA.2015.7449458.

[9] Forster, Jens, Christoph Schmidt, Thomas Hoyoux, Oscar Koller, Uwe Zelle, Justus H. Piater and Hermann Ney. “RWTH-PHOENIX-Weather: A Large Vocabulary Sign Language Recognition and Translation Corpus.” International Conference on Language Resources and Evaluation (2012).

[10] Jens Forster, Christoph Schmidt, Oscar Koller, Martin Bellgardt, and Hermann Ney. 2014. Extensions of the Sign Language Recognition and Translation Corpus RWTH-PHOENIX-Weather. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14), pages 1911–1916, Reykjavik, Iceland. European Language Resources Association (ELRA).

[11] Fu, Jianlong, Heliang Zheng and Tao Mei. “Look Closer to See Better: Recurrent Attention Convolutional Neural Network for Fine-Grained Image Recognition.” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017): 4476-4484.

[12] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In Proceedings of the 23rd international conference on Machine learning (ICML ’06). Association for Computing Machinery, New York, NY, USA, 369–376. https://doi.org/10.1145/1143844.1143891

[13] K. He, X. Zhang, S. Ren and J. Sun, ”Deep Residual Learning for Image Recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

[14] Cho, Jaechan, Yongchul Jung, Dong-Sun Kim, Seongjoo Lee, and Yunho Jung. 2019. ”Moving Object Detection Based on Optical Flow Estimation and a Gaussian Mixture Model for Advanced Driver Assistance Systems” Sensors 19, no. 14: 3217. https://doi.org/10.3390/s19143217

[15] Farnebäck, Gunnar. “Two-Frame Motion Estimation Based on Polynomial Expansion.” Scandinavian Conference on Image Analysis (2003).

[16] Li, Dongxu, Chenchen Xu, Xin Yu, Kaihao Zhang, Ben Swift, Hanna Suominen and Hongdong Li. “TSPNet: Hierarchical Feature Learning via Temporal Semantic Pyramid for Sign Language Translation.” ArXiv abs/2010.05468 (2020): n. pag.

[17] Koller, Oscar, Jens Forster and Hermann Ney. “Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers.” Comput. Vis. Image Underst. 141 (2015): 108-125.

[18] O. Koller, H. Ney and R. Bowden, ”Deep Hand: How to Train a CNN on 1 Million Hand Images When Your Data is Continuous and Weakly Labelled,” 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3793-3802, doi: 10.1109/CVPR.2016.412.

[19] Koller, Oscar, Sepehr Zargaran, Hermann Ney and R. Bowden. “Deep Sign: Hybrid CNN-HMM for Continuous Sign Language Recognition.” British Machine Vision Conference (2016).

[20] Koller, Oscar, Sepehr Zargaran and Hermann Ney. “Re-Sign: Re-Aligned End-to-End Sequence Modelling with Deep Recurrent CNN-HMMs.” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2017): 3416-3424.

[21] Koller, Oscar, Sepehr Zargaran, Hermann Ney and R. Bowden. “Deep Sign: Enabling Robust Statistical Continuous Sign Language Recognition via Hybrid CNN-HMMs.” International Journal of Computer Vision 126 (2018): 1311-1325.

[22] Desai P.; Sujatha C.; Chakraborty S.; Ansuman S.;Bhandari S.; Kardiguddi S, “Next frame prediction using ConvLSTM”, Journal of Physics: Conference Series, Volume 2161, Year 2022, DOI:10.1088/1742-6596/2161/1/012024

[23] Li, Dongxu, Cristian Rodriguez-Opazo, Xin Yu and Hongdong Li. “Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison.” 2020 IEEE Winter Conference on Applications of Computer Vision (WACV) (2019): 1448-1458.

[24] Lowe, D.G. Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004). https://doi.org/10.1023/B:VISI.0000029664.99615.94

[25] Desai P.; Sujatha C.; Chakraborty S.; Ansuman S.;Bhandari S.; Kardiguddi S, “Next frame prediction using ConvLSTM”, Journal of Physics: Conference Series, Volume 2161, Year 2022, DOI:10.1088/1742-6596/2161/1/012024

[26] P. Kumar, Sujatha. C and P. Desai, ”Person Tracking with Re-Identification in Multi-Camera Setup: A Distributed Approach,” 2022 2nd International Conference on Intelligent Technologies (CONIT), 2022, pp. 1-5.

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: an imperative style, high-performance deep learning library. Proceedings of the 33rd International Conference on Neural Information Processing Systems. Curran Associates Inc., Red Hook, NY, USA, Article 721, 8026–8037.

[28] Makanapura N.; Sujatha C.; Patil P.R.; Desai P, “Classification of plant seedlings using deep convolutional neural network architectures”, Journal of Physics: Conference Series, Volume 2161, Year 2022

[29] S. Siami-Namini, N. Tavakoli and A. S. Namin, ”The Performance of LSTM and BiLSTM in Forecasting Time Series,” 2019 IEEE International Conference on Big Data (Big Data), 2019, pp. 3285-3292, doi: 10.1109/BigData47090.2019.9005997.

[30] Tharwat, Alaa, Tarek Gaber, Aboul Ella Hassanien, Mohamed K. Shahin and Basma Refaat. “SIFT-Based Arabic Sign Language Recognition System.” Afro-European Conference for Industrial Advancement (2014).