American Sign Language Gesture Recognition using CNN

Nasreen Taj M B
Department of ISE
GMIT, Davangere
nasreent@gmit.ac.in

Nikhitha H B
Department of ISE
GMIT, Davangere
hbnikhitha@gmail.com

Aishwarya S Yeli
Department of ISE
GMIT, Davangere
aishwaryasyeli30@gmail.com

Pooja K M
Department of ISE
GMIT, Davangere
poojakm2007@gmail.com

Pavan S Raikar
Department of ISE
GMIT, Davangere
pavanraikar333@gmail.com

***Abstract*— Hand-based gestures are used by those who are unable to speak. Regrettably, the vast majority of individuals are unaware of the meanings behind these gestures. We present a real-time hand gesture detection system based on data recorded by the Microsoft Kinect RGB-D camera in an attempt to bridge the gap. We employed computer vision techniques like 3D building and affine transformation because there is no one-to-one mapping between the pixels of the depth and the RGB camera. After one-to-one mapping was achieved, the hand motions were segmented from the background noise. Convolutional Neural Networks (CNNs) were used to train 36 static motions related to the alphabets and numbers in American Sign Language (ASL). Using 45,000 RGB photos, the model obtained a training accuracy of 98.81 percent. There are 45,000 depth images and a total of 45,000 depth photographs. Furthermore, 10 ISL dynamic word gestures were trained using convolutional LSTMs, and 1080 films were trained with an accuracy of 99.08 percent. The model predicted ISL static gestures accurately in real time, indicating that more research into sentence production through gestures is needed. When the ISL model weights were translated from learnt to ASL, the model demonstrated competitive adaptation to ASL gestures, with an accuracy of 97.71 percent.**

***Index Terms*—American Sign Language, Gesture Recognition, Microsoft Kinect, LSTMs, and Convolutional Neural Networks.**

I. INTRODUCTION

The American Sign Language system includes conventional hand-based motions that are used by speech-impaired people in America for communication. Given the complexity and breadth of these indications, many individuals are unaware of them, obstructing communication between the speech impaired and non-speech impaired. With the recent popularity of deep learning, there is a lot of active applied research in the field of computer vision [1]. However, despite this, research into ISL gesture recognition has been restricted and insufficient. With the goal of improving the situation, our article focuses on establishing a standard for the recognition of ISL gestures as well as developing a model to assist speech-impaired communication.

In ISL, recognising gestures is a difficult challenge. In contrast to ASL (alphabets), which employs only one hand, ISL uses both hands to represent a motion. When using feature extractors like Hough Transform [2] and Scale Invariant Feature Transform (SIFT) [3, this adds to the complexity. Also So, using data from the Microsoft Kinect RGB-D camera, we used depth-based segmentation. The hand gesture region can be readily segregated from the background using data from the Kinect's depth channel. However, there is a lack of one-to-one mapping between the pixels of the depth channels and the pixels of the RGB channels in this case. This prevents direct overlap segmentation of the hand region in the RGB frame, which is necessary for feature extraction. In the following section of this paper, the problem is solved and discussed. Using the Kinect, this technique allows for the segmentation of the hand region from any backdrop conditions. When attempting to anticipate motions in real time, the problem of background complexity arises, which may obstruct good gesture prediction. As a result, separating the hand motion zone from the background becomes critical. Though techniques such as colour space segmentation and the Otsu's methodology [4] exist, they all have limitations in terms of backdrop conditions.

As previously stated, handmade feature techniques for extracting information from a gesture will not generalise effectively across all gestures. For instance, Fig. 1 shows the Gradient Hough Transform (GHT) [5] applied to various hand orientations. The circles computed along the margins of the hand in Fig. 1(a) are based on parameters that are appropriate for this pose. The circles computed using the same parameters on a different attitude of the hand are shown in Fig. 1(b). All of the computed circles are clearly beyond the region of interest and would be useless for extracting features.

Similarly, any other handcrafted feature technique would have limits when it came to adapting to all varied postures and flexural angles of the hand motions. The use of learning-based approaches like CNNs in image processing has shown to produce great outcomes and is setting new standards. As a result, CNNs were selected as the core deep learning architecture for the recognition challenge. The photos of static motions from ISL were trained using a multi-layered CNN-based architecture. The Kinect was used to create around 90,000 depth and RGB images. For the training of dynamic hand motions, LSTMs with a convolutional kernel were chosen. In total, about 1,080 videos were created for training purposes. The outcomes of the two strategies are presented later in this paper.

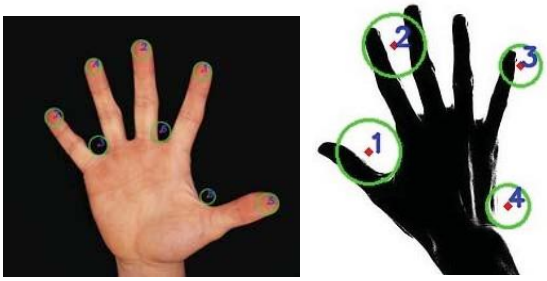

Fig. 1. (a) Successful application of GHT to a posture of a hand by estimating parameters and (b) Unsuccessful application of GHT to a different view of the hand applying the same parameters.

The prior relevant work is covered in Section II. The technique for mapping between depth and RGB pixels is explained in Section III. The datasets used and the approaches used to achieve generalisation are discussed in Section IV. The alternative designs for training the two datasets are explained in Section V. Section VI presents the findings, and Section VII brings the work to a close.

II. RELATED WORK

There has been a lot of study focused on hand gesture detection up until now, but there has been no good research focused on ISL. Singha et al. [6] developed a database with 240 photos for 24 ISL

symptoms. They took the photos' eigen values and categorised them based on the Euclidean distance. Their model had a 97 percent classification accuracy, however the images were constrained to a set of fixed background circumstances and the gestures were static. Pei Xu et al. [7] created a real-time Human-Computer Interaction system based on hand movements, in which each hand image was preprocessed using hand colour filtering, Gaussian blurring, morphological modifications, and other techniques. Following that, a CNN was trained to recognise the motions, and a Kalman estimator was used to move the mouse cursor in response to the gestures detected. On 16 gestures, our system achieved an average accuracy of 99.8%. Liao et al. [8] employed an Intel Real Sense RGB-Depth sensor to segregate the hand region from the background using depth perception algorithms. In RGB photos, they also apply the Generalized Hough transform to detect and segment the hand. A dual channel convolutional neural network was used to train the segmented depth and RGB pictures. While identifying 24 ASL gestures, the suggested approach obtains a classification accuracy of 99.4 percent. Furthermore, the outcomes were based solely on the photos used for training. The model's performance in real life was not specified in the research. In depth photos, the gradient kernel descriptor was employed as a feature extractor, whereas in RGB images, SIFT was used. On the ASL database, the combined data were fed into an SVM, which yielded a classification accuracy of 90.2 percent. To identify dynamic gestures in real time, Molchanov et al. [10] used a Recurrent 3D Convolutional Neural Network (R3DCNN) with temporal classification. Depth, colour, and stereo-IR sensors were used to acquire the data. The large-scale Sport- 1M was used to pre-train the 3D CNN[13] offers a unique neural network-based technique known as Dense Image Network (DIN) that compresses a video incorporating hand gestures into a compact form that distils their spatiotemporal evolution. The DIN is sent into a CNN, which learns the video's features more quickly and efficiently, saving time and space. On action and gesture recognition, the technique reached benchmark results. Kopuklu et al. [14] adopt a strategy that feeds RGB modalities and optical flow data into a deep neural network. Each RGB image has a corresponding optical flow image, which may be used to trace the hand's movement. On the NVIDIA benchmark, a neural network was trained on these photos and achieved a classification accuracy of 84.7 percent, while the Jester and ChaLearn benchmarks yielded 96.28 percent and 57.4 percent, respectively. Mukesh [15] Using the ASL database (alphabets and numbers) and CNNs, they were able to obtain an 85.51 percent training accuracy. Then, using the same CNN network, a new ISL database was created, which obtained 85.51 percent training accuracy. Again, the model's performance was not evaluated in real time.To our knowledge, this is the first time a model for recognising ISL gestures in real time has been constructed. Our device works effectively in every environment and with hands of any size.

III. MAPPING DEPTH AND RGB PIXELS

As previously stated, there is no one-to-one mapping between RGB and Depth pixels. To accomplish this, we used computer vision techniques. To begin, a 3D representation of the entire scene in the depth camera's range of view is calculated. Since the kinect only provided raw depth readings between 0 and 2047, triangulation was used to derive the depth value in millimetres. Equation 1 is the solution to the problem.

$$z^{-1}_{world} = \frac{m \cdot d' + Z^{-1} + n}{} \quad (1)$$

where z_{world} is the distance in millimeters between the Kinect and a real-world point, and d is the normalized disparity value d = m . d + n, where m and n are the de-normalization parameters, by normalising the raw disparity value d between 0 and 2047. z_o is the distance between the Kinect and the predefined reference pattern, and b and f are the depth camera's base length and focal length. Equations 2 and 3 can estimate the entire 3D world model from the image locations (x, y) after obtaining the real world location of each pixel from the depth camera.

$$x_{world} = -Z_{world} \cdot \ x - x_o + \delta x \quad (2)$$
$$y_{world} = -Z_{world} \cdot \ y - y_o + \delta y \quad (3)$$

where xworld and yworld are the coordinates of a point in 3D space, (x0, y0) is the primary location of the depth image, and x and y are lens distortion adjustments. Affine transformation is used to the resulting 3D model to convert it to RGB camera point of view. Equation 4 is the solution to the problem.

(4)

The LHS of equation 4 denotes the 3D coordinates with regard to the RGB camera point of view, while R and T are rotation and translation parameters, respectively.

The true location of the points in the RGB plane can be calculated using equation 5 and the derived 3D co-ordinates. The above example proves to be effective for gesture segmentation in both the RGB and depth planes.

where fRGB is the RGB camera's focal length. This method effectively registers the depth and RGB camera's pixels. Figure 2(a) depicts an example of gesture segmentation in the RGB plane without registration, while Figure 2(b) depicts the result of using the above-mentioned registration approaches.

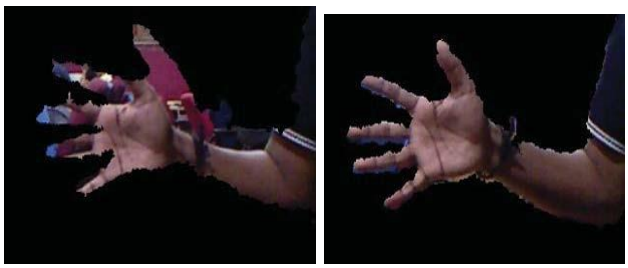

Fig. 2. (a) Segmentation of gesture in RGB without registration (b) Segmentation of gesture in RGB after segmentation.

The above example proves to be effective for gesture segmentation in both the RGB and depth planes.

IV. DATASET

A. *Details of the dataset*

We built our own dataset for training and testing because there isn't a standard dataset accessible on ISL. The dataset was split into two halves, one for Fig. 3 and the other for Fig. 4. Examples of depth and RGB pairs for the alphabets I, V, W, and the numbers 1, 2, and 3.

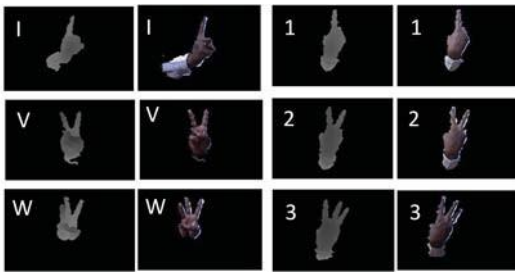

Fig. 3. Examples of depth and RGB pairs for alphabets I, V, W and numbers1, 2, 3

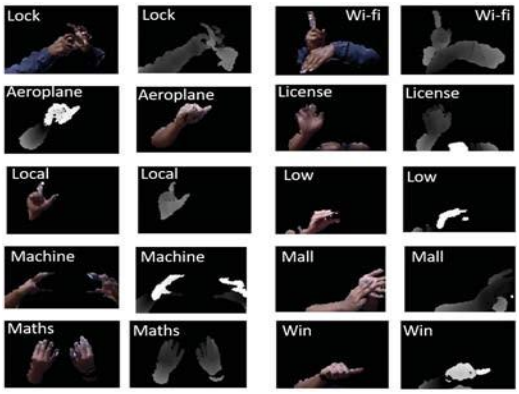

Fig. 4. Database for Dynamic words

Hand-based movements can be split in the RGB plane following registration of depth and RGB pixels, as indicated in the preceding section. As a result, the model's complexity is reduced as it tries to maximise its performance regardless of the backdrop circumstance. The data was collected from five different topic areas, with an average age of 25 and both male and female genders represented. Our dataset includes 45,000 photos based on the depth camera and 45,000 images based on the RGB camera for the static based gestures, i.e., for the dataset containing gestures of alphabets and digits.

Aside from that, we took and saved additional 45,000 photographs that were not segmented by background. In order to generalise the parameters of the training algorithm while predicting in real-time, the dataset was gathered in our lab under two different lighting conditions. We used videos of ten regularly used ISL words to create the dynamic movements. The terms that were picked were Wi-Fi, Lock, Maths, Mall, Low, Win, Machine, Local, License, Wi-Fi, Lock, Maths, Mall, Low, Win, Machine, Local, License as well as an aeroplane at 9 p.m., a total of 1,080 videos were taken. various frame rates The Kinect's standard frame rate is 30 frames per second. 30 frames per second However, we recorded the videos at frame rates of 27, 24, and 25. 21, 18, 15, 33, 36, and 39 frames per second This was put in place in order to allow for timing variation between the same movements Additional temporal features can be extracted using the training technique.

Figure 3 shows instances of segmented RGB-D pairings acquired during data gathering for the alphabets I, V, W, and numbers 1,2, and 3. Except for the fact that the postures are inverted, the pairs 1 and I, 2 and V, 3 and W appear to be quite similar. The findings are shown in Section VI, where the model correctly distinguishes between the pairs of signs. The RGB-Depth single frame database image for dynamic ISL words is shown in Figure 4. The whole database for static gestures is shown in RGB plane in Figure 5.

B. Techniques to attain Generalization

CNNs are not invariant in terms of scale, rotation, or translation. The dataset for static gestures is made up of gestures that are concentrated in a small area of the visual frame.

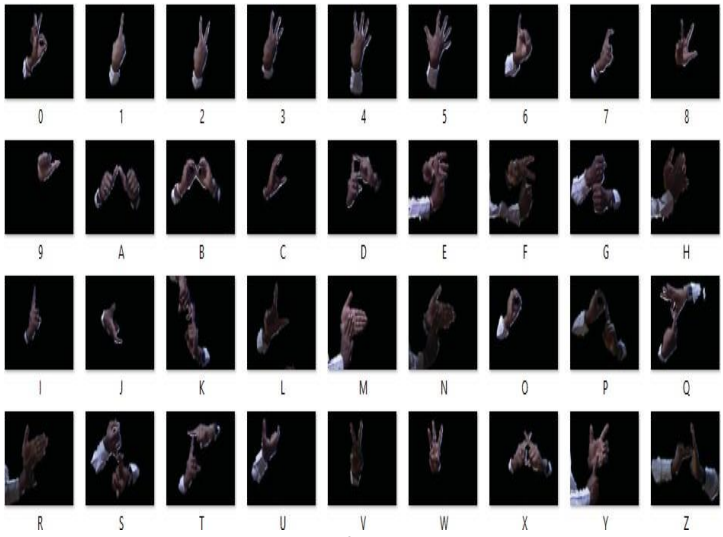

Fig. 5. Database for Static gestures

Though this may not have an impact on training, it may have an impact on real-time performance because the CNN requires the gesture to be presented to the camera in the same narrow area during data gathering. This isn't ideal in real time, because the CNN needs to be able to forecast accurately regardless of where the gesture is in the picture. We employed artificial data synthesis to generate more data in order to accomplish generalisation and improve real-time performance. The effect of applying label preserving changes to the original image of a dog to portray it in a slightly different orientation is shown in Fig. 6. The CNN will be able to learn more features as a result of these images, making it more resilient and invariant to scale, rotation, and translation.

V. METHODOLOGY

The approach and architectures used for training are discussed in this section.

A. Static-based gesture training architectures

Liao et al. [8] employed a double-channel convolutional neural network architecture to learn segmented depth and RGB pictures at the same time. We used a similar strategy to train our RGB-D pairs at the same time. We also trained depth and RGB pictures individually using a similar CNN-based architecture (Fig. 7) and evaluated their performance both offline and in real time.

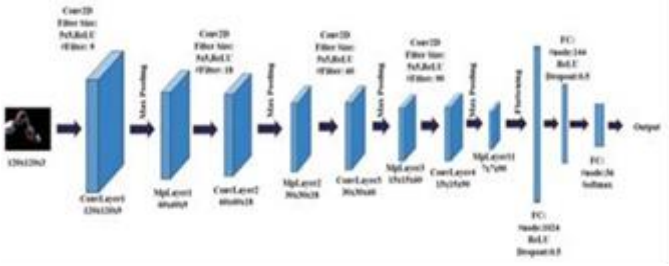

Fig. 7. CNN architecture for training RGB and depth images separately

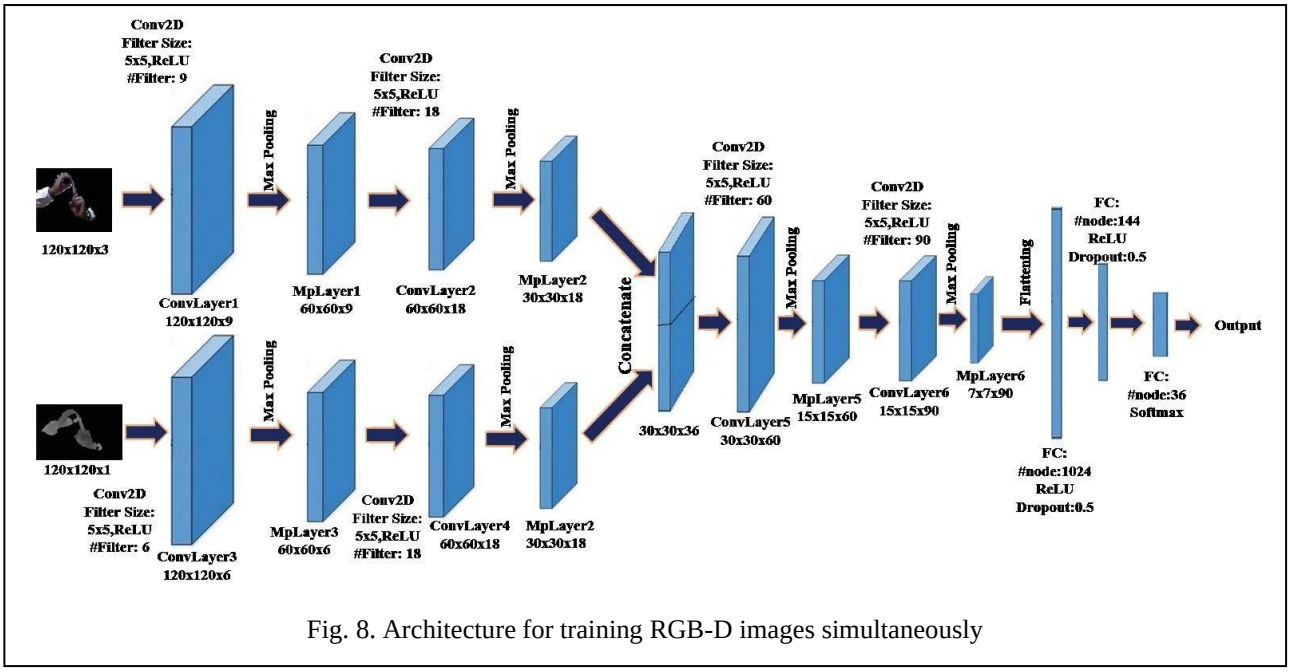

Fig. 8. Architecture for training RGB-D images simultaneously

The photos are resized to 120x120 pixels while keeping the aspect ratio. They are reduced to 90 feature maps of size 7x7 after passing through a succession of convolutional layers, ReLu layers, and max-pooling layers. Then they're sent to two completely connected layers, each with 1024 and 144 neurons. Finally, softmax activation is used to classify the movements in the final layer.

We also wanted to see how well our model performed in terms of ASL. We employed transfer learning from the model trained with only RGB pictures, using datasets from [16]. The weights from the convolutional layers were kept, while those from the dense layer were reset to random integers. The following section explains the results for the same. The CNN-based architecture utilised to train is shown in Figure 8. the RGB and depth photos at the same time. The photos can be found below. after resizing to 120x120 pixels while keeping the aspect ratios are sent into two distinct channels, where they are passed back and forth. 30 feature maps of 18x18 pixels were created using a sequence of convolutional, Relu, and maxpooling layers. Then the two channels are combined into two. They are then they've been passed through 5 convolutional layers once more 90 7x7 feature maps have been trimmed. They are then passed. through two completely connected layers with 1024 and 144 bits each respectively, neurons Softmax activation is used in the final layer. classification.

B. Dynamic-based gesture training structures

The dataset for dynamic gestures was acquired at 9 distinct frame rates, as mentioned in the previous section. Downsampling a video by 5 or 6 times removes unnecessary frames while preserving temporal information. As a result, all of the videos were down-sampled by taking into account every 6th frame and discarding the remainder. In addition, the maximum number of frames was set at 20, and any video sequence with fewer than 20 frames was zero padded at the start.

Because 3D CNNs are computationally intensive, LSTMs with a convolutional kernel were chosen for training the movies over a 3D-CNN architecture. We created distinct architectures for training the depth and RGB based films individually, as well as a single dual channel design to train both at the same time.

The Convolutional LSTM-based architecture used to train the depth and RGB films separately is shown in Fig. 9. There are two convolutional LSTM layers, each with a kernel size of (5,5) and a total of 32 or 54 filters. The strides were meant to be two. These

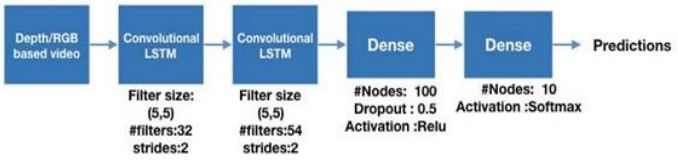

Fig. 9. Single channel architecture for video training

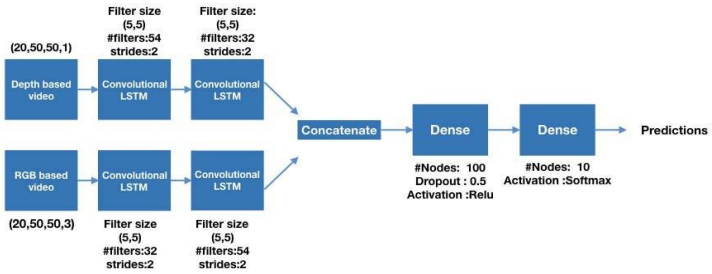

Fig. 10. Dual channel architecture for video training

two layers' output is then fed into a fully connected layer with 100 nodes with RelU activation. The final softmax layer divides the videos into ten categories. The dual channel LSTM architecture is similar to Fig. 9, except that the depth and RGB films are routed through two different channels of Convolutional LSTMs at the same time before the layers are fused. The architecture used in this case is depicted in Fig. 10.

VI. RESULTS AND TRAINING

The results of training the models outlined in the preceding part are discussed in this section. NVIDIA Tesla K80 GPUs were used to train all of the models. After mixing the data acquired from all five subject topics, the dataset was divided 70:30 across the training and testing sets.

The following parameters has been employed in the training:

- Epochs : 40 for static based gestures and 30 for dynamic based gestures
- Batch Size : 32
- Optimizer : Adam
- Learning rate : 0.01
- Weight Initializers : Xavier/Zero's

Artificial data synthesis was implemented in real time for the training of static based motions. Table. I present the findings

obtained after training all of the models with the hyper-parameters specified.

The training accuracies of the depth-only CNN architecture and the RGB-based CNN architecture are 97.43 percent and 97.21 percent, respectively. Both the depth and RGB based models have testing accuracies of 99.4 percent and 99.75 percent, respectively. The testing accuracy is higher than the training accuracy because the testing set was not subjected to artificial data creation, making it considerably easier to predict than the training set. The combined RGB + Depth based CNN model outperformed the other two models with a training accuracy of 98.81 percent and a testing set accuracy of 99.6 percent, outperforming them both. The plot of epochs vs. Loss/Accuracy for the same is shown to the left of Fig. 11.

People ranging in age from 5 to 50, a total of roughly 50 people (non-inclusive of the topic areas chosen for training) were asked to do the gestures in front of the camera in real time to evaluate the models' performance. It was discovered that all three models performed similarly, meaning that all of the people's movements were precisely predicted.

The training accuracy for unsegmented RGB images was 95.87 percent. When compared to the model trained using segmented RGB images, the model demonstrated a drop in performance when deployed in real-time. As a result, while applying our model in real-time, background subtraction proved to be critical and successful.

Figures 12,13,14,15,16,17 demonstrate real-time prediction examples for the alphabets I, V, W, and integers 1, 2, 3. As can be seen, the model correctly predicts couples V and 2, I and 1, and W and 3 with high accuracy. The accuracy of transfer learning on the ASL dataset was 97.71 percent, almost identical to the accuracy obtained in [8]. In addition, the model in [8] was trained for 48 epochs, which is eight more than the value of our epoch hyper-parameter. As a result, the model demonstrated adaptability by extracting useful features from the ASL dataset. As a result, while deploying our model in real time, background subtraction proved to be critical and effective. Examples of real-time prediction for the alphabets I, V, W, and integers 1, 2, 3 are shown in Figures 12,13,14,15,16,17. The model accurately predicts couples V and 2, I and 1, and W and 3 as can be shown.

Artificial data synthesis was not used for the dynamic dataset due to computational resource constraints, but the data was captured at 9 distinct frame rates to achieve temporal diversity, as previously indicated. On the training set, the depth only dynamic model had a precision of 97.52 percent, whereas the RGB model had a precision of just 98.11 percent. On the training set, the combined RGB + Depth model has a precision of 99.08 percent. The plot of Epochs vs. Loss/Accuracy for the same is shown to the right of Fig. 11. Despite the use of regularisation techniques such as dropout, the performance on the testing set was 78.3 percent and about the same level for depth only and RGB only dynamic models, implying the necessity to produce datasets having more variability.

VII. Conclusion and Future Work

Based on the incoming picture input from the camera, this research provides a real-time model for ISL gesture identification. Kinect. It was possible to perform effective real-time backgorund subtraction. Using approaches for depth perception To create one-to-one mapping between the two, computer vision techniques were applied. depth, as well as the RGB pixels For training, a custom dataset was created and various models were utilised. + the depth On the training set, the segmented static model gets 98.81 percent accuracy while the dynamic model achieves 99.08 percent accuracy.

TABLE I
Results of Training

Sl No.	Method	Classes	Training accuracy (%)	Testing accuracy (%)
1.	Depth only CNN	36	97.43	99.4
2.	Segmented RGB only CNN	36	97.21	99.75
3.	Unsegmented RGB only CNN	36	95.87	99.68
4.	Depth + Segmented RGB CNN	36	98.81	99.6
5.	ASL with transfer learning	24	97.71	99.0
6.	RGB based ASL from [8]	24	97.8	100
6.	Dynamic Depth only	10	97.52	76.4
7.	Dynamic RGB only	10	98.11	77.6
8.	Dynamic Depth + RGB	10	99.08	78.3

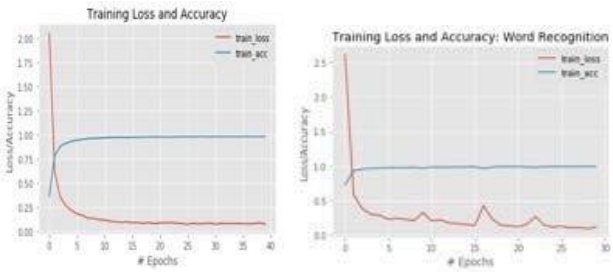

Fig. 11. **Left:** Plot of training loss/accuracy for dual channel static gesture prediction model.**Right:** Plot for dual channel dynamic gesture prediction model

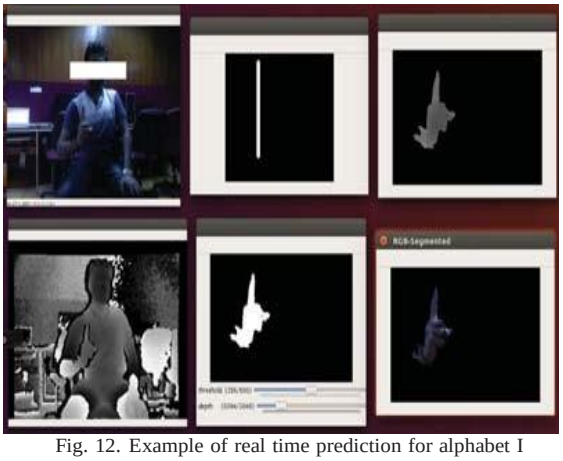

Fig. 12. Example of real time prediction for alphabet I

Fig. 13. Example of real time prediction for number 1

Fig. 14. Example of real time prediction for alphabet V

Fig. 15. Example of real time prediction for number 2

Fig. 16. Example of real time prediction for alphabet W

REFERENCES

[1] Q. Wu,Y. Liu,Q. Li andS. Jin, " The operation of deep literacy in CV ", CAC, Jinan, 2017,pp. 6522- 6527.

[2] D. Ballard, Generalizing the Hough transfigure, Pattern Recognition,vol. 13,no. 2,pp. 111122, 1981.

[3] 3)D.G. Lowe, Image Features from Scale Invariant Keypoints, IJCV,vol. 13,no. 2,pp. 111122, 1981.

[4] V. Bhavana,G.M. Surya Mouli andG.V. Lakshmi, " Hand Gesture Recognition Using Otsu's Method, " IEEE ICCIC, Coimbatore, 2017,pp. 1- 4.

[5] Y. Liu,J. Zhang, andJ. Tian, An image localization system grounded on grade Hough transfigure, MIPPR 2015 Remote seeing Image Processing, Geographic Information Systems, and Other operations, 2015

[6] J. Singha andK. Das, " subscribe Language Recognition Using Euclidean Distance Grounded Bracket fashion ", IJACSA,vol. 4,no. 2, 2013.

[7] Pei Xu, A real time hand gesture recognition and mortal- computer commerce system, In Proceeding of the Computer Vision and Pattern Recognition, 2017.

[8] B. Liao,J. Li,Z. Ju andG. Ouyang, " Hand Gesture Recognition with Generalized DC CNN Using Real sense, " 2018 Eighth ICIST, Cordoba, 2018,pp. 84- 90.

[9] K.O. Rodriguez andG.C. Chavez, Finger Spelling Recognition from RGB- D Information Using Kernel Descriptor, 2013 XXVI Conference on plates, Patterns and Images, 2013.

[10] P. Molchanov,X. Yang,S. Gupta,K. Kim,S. Tyree, andJ. Kautz, Online Discovery and Bracket of Dynamic Hand Gestures with intermittent 3D Convolutional Neural Networks, IEEE Conference on CVPR, 2016.

[11] A. Karpathy,G. Toderici andS.Shetty, " Large scale videotape bracket with CNN " In CVPR, 2014.

[12] Okan Kpkl, Ahmet Gunduz, Neslihan Kose, Gerhard Rigoll, Real-time Hand Gesture Detection and Classification using Convolutional Neural Networks, paper accepted to IEEE International Conference on Automatic Face and Gesture Recognition(FG 2019).

[13] Xiaokai Chen, Ke Gao DenseImage Network Video Spatial-Temporal elaboration Garbling and Understanding, paper submitted to ArXiv on 19 May 2018.

[14] O. Kopuklu,N. Kose, andG. Rigoll, Motion Fused Frames Data Level Fusion Strategy for Hand Gesture Recognition, 2018 IEEE/ CVF Conference on Computer Vision and Pattern Recognition Workshops(CVPRW), 2018.

[15] Mukesh Kumar Makwana, subscribe language Recognition, Mtech thesis submitted to Indian Institute of Science, Bengaluru, June 2017.

[16] ASL database source Kaggle- ASL ABC dataset