Received: 4 November 2023 | *Revised: 15 April 2024* | *Accepted: 25 April 2024*
DOI: 10.37917/ijeee.22.1.44 *Vol. 22 | Issue 1 | June 2026*

Open Access

Iraqi Journal for Electrical and Electronic Engineering
Original Article

Enhancing American Sign Language Recognition Through Transfer Learning Techniques

Wahhab Muslim Mashloosh*[1], Murteza Hanoon Tuama[1], Ghadah Adil Najm[2]
[1]Department of Computer Techniques Engineering, Imam Al-Kadhum College (IKC), Baghdad, Iraq
[2]Department of Science, Directorate of Al Qadissiya Education, Ministry of Education, Qadissiya, Iraq

Correspondance
*Wahhab Muslim Mashloosh
Department of Computer Techniques Engineering, Imam Al-Kadhum College (IKC), Baghdad, Iraq
Email: wahhabmuslim@alkadhum-col.edu.iq

Abstract
Aiming to enhance the accuracy of sign classification in sign language (SL), this research presents an innovative approach that combines hand-engineered characteristics with deep learning (DL) algorithms. The focus is on American Sign Language (ASL), a critical communication tool for the deaf and hard-of-hearing community. The goal is to bridge the existing communication chasm between SL users and the general public by designing a real-time SL recognition system that allows non- SL users to converse with the hearing-impaired individuals. The application and assessment of various machine learning (ML) models, such as VGG19, DenseNet, ResNet50, MobileNet, and NASNetMobile, yielded promising outcomes with superior evalu- ation metrics. These models exhibit utility in the classification of ASL signs as they can differentiate between diverse hand gestures with high accuracy (ACC). The paper highlights the potential of these models across an array of ASL recognition applica- tions, considering factors like computational resources, model dimension, and real-time functionality. The findings endorse the application of ML techniques in SL interpretation, promoting inclusive communication for those with hearing impairment.

Keywords
VGG19, Deep Learning, ResNet, American Sign Language, DenseNet, NaSNetMobile, MobileNet.

I. Introduction

The global population of individuals who are deaf or hard of hearing has now exceeded 400 million, shedding light on the growing necessity for research aimed at enhancing communication for this significant demographic [1]. ASL, a visually expressive and intricate natural language, plays a pivotal role within the deaf and hard-of-hearing communities in the United States and parts of Canada [2]. ASL relies on a complex interplay of handshapes, facial expressions, body movements, and visual cues to effectively convey messages. The drive for inclusivity and the improvement of communication within the Deaf community underscores the profound significance of ASL.

Recent years have witnessed a profound transformation in the field of ASL comprehension and interpretation, largely owing to advancements in Machine Learning (ML) technologies. Advanced Convolutional Neural Network (CNN) models like VGG19, DenseNet, MobileNet, and NASNetMobile have emerged as formidable tools for recognizing the intricate handshapes, gestures, and motions integral to ASL, whether in images or videos [3]. These models, bolstered by the capabilities of DL algorithms, have demonstrated exceptional proficiency in processing visual input and accurately discerning the diverse components of SL. The integration of ML models has catalyzed advancements in SL [4] detection, translation, and interpretation, promising to bridge communication divides, enhance accessibility, and empower individuals with hearing impairments. This technological revolution holds immense po- tential, extending its benefits across various domains, including education, employment, and daily interactions.

What distinguishes these models is their remarkable ability to classify ASL signs, achieving outstanding ACC in distinguishing between diverse hand gestures. Tailored to address specific requirements such as computational resources, model size, and real-time performance, these models unlock a plethora of possibilities for a multitude of ASL recognition applications. The research findings in bear significant implications for enhancing inclusive communication among hearing-impaired individuals and promoting the utilization of ML methodologies in SL interpretation.

The field of ASL-related ML algorithms has witnessed a slew of investigations. For example, in the realm of ASL identification, Zaki et al. employed Basic Component Analysis for feature extraction, complemented by a Hidden-Markov Model for classification [5]. Motion sensors played a pivotal role in Ching Hua et al.'s work, which harnessed the power of K-Nearest Neighbors (k-NN) and Support Vector Machine algorithms for categorization [6]. Cao et al. leveraged the depth comparison feature of the Microsoft Kinect to construct feature vectors, subsequently categorized using Random Forest and limited link angle algorithms [7]. In the quest for ASL recognition, Aryanie and team relied on the k-NN Classifier [8]. Pansare et al. concentrated on interpreting ASL with a focus on alphabet letters, employing the edge orientation histogram technique [9]. Truong et al. employed the Haar-cascade method to build an ASL translator accommodating both text and audio [10]. Joshi et al. innovatively utilized edge detection and cross-correlation techniques to fashion an ASL translator [11].

Despite the potential exhibited by ML models in interpreting sign language, the complexities inherent to ASL continue to pose formidable challenges. Ongoing research and advancements are imperative, particularly in the pursuit of heightened Accuracy (ACC), robustness, and adaptability in the face of variations in signing techniques, lighting conditions, and camera perspectives. Nevertheless, the infusion of ML methodologies into ASL research holds immense promise in propelling inclusivity and facilitating effective communication between the deaf and hearing communities. Persistent endeavors in this arena promise not only improved accessibility but also the removal of barriers for individuals who depend on ASL as their primary mode of communication.

The objective animating our research is to delve into and advance the study of ASL by embarking on a thorough exploration of diverse ML algorithms and we seek to foster an environment where the boundaries of communication for the hearing-impaired are continually pushed forward in our pursuit of innovation. Our aspiration is to yield substantial contributions to this domain while addressing the extant challenges associated with ASL recognition and interpretation. Through our concerted research efforts, we aspire to augment ACC, efficiency, and adaptability within ML models tailored to ASL, all with the ultimate aim of enhancing accessibility and communication for individuals whose primary language is ASL.

A PCA analyzer model was used in the study by Starner et al. [12] to identify the hand form, and hand movements were examined using a motion chain code to find actionable indications. The method produced an error rate of 10.91%, demonstrating its efficacy in identifying and deciphering ASL movements.

In order to improve recognition speed while maintaining high ACC, Zhang et al. [13] proposed a two-model approach. The first model is a TMDHMM network, and the second model utilizes an efficient hierarchical feature characterization technique. By combining these models, the researchers were able to achieve faster and more accurate ASL recognition.

In a study conducted by Cui et al. [14], a three-stage model was suggested on the basis of hand motion. Groups of hands were photographed in the first stage, and the sequence was evaluated to identify specific actions and pinpoint the start and stop locations. In the subsequent stage, methods for extracting hand outlines from the photos through image segmentation were used. Finally, in the determination phase, a scoring mechanism was utilized to determine if the image corresponded to the taught indicators, indicating successful ASL recognition. This multi-stage approach showcased its effectiveness in recognizing ASL gestures and achieving accurate recognition results.

In a research paper by Akmeliawati et al. [15], a novel approach utilizing a wearable glove with contrasting colors was proposed to aid the camera in detecting fingertips for ASL recognition. The technique involved capturing the image frame, separating the colors of the glove from the background, locating the centroid of the hand image, and utilizing this information to make the final decision regarding the recognized ASL gesture. This innovative method showcased its potential in real-time ASL recognition by leveraging color contrast and image processing techniques to accurately detect and interpret hand movements.

In a study conducted by Liang et al. [16], multiple criteria including hand posture, position, motion, and orientation were taken into consideration for ASL recognition. Given the signif- icant number of postures (51), orientations (6), and positions (6), a conventional Hidden Markov Model (HMM) approach was employed, which achieved an approximate precision (PREC) of 80.4%. Despite the lower ACC compared to other models, the utilization of HMM allowed for the incorporation of various factors in ASL recognition, showcasing its potential in capturing the complexity of hand gestures and achieving satisfactory results in real-time scenarios.

Neural networks combined with the Hough Transform have been utilized in ASL recognition models as described by

Munib et al. [17]. These models utilize vector feature functions that are robust against rotational and scalar disturbances, en- suring adaptability and reliability in ASL recognition. Through this approach, a notable ACC of approximately 92% has been achieved, validating the effectiveness of this method for ASL detection. The integration of neural networks and the Hough Transform provides a powerful framework for robust and accurate ASL recognition, offering potential benefits in various applications.

In the study conducted by Nandy et al. [18], a communication system for the International Sign Language (ISL) is proposed. The system incorporates both static and dynamic hand motions and involves the creation of a video database with multiple recordings capturing various indications. For feature representation, the bearing histogram is employed due to its robustness against changes in brightness and viewpoint. Recognition is performed using distance-based measures such as Euclidean distance and k-NN. This approach demonstrates the potential for accurate recognition of ISL gestures and contributes to the advancement of sign language communication systems.

The papers discussed in the context of ASL recognition encompass various techniques and approaches to improve ACC and efficiency. Examples include feature-based recognition algorithms, the Hough Transform, wearable gloves, wearable model analysis (HMM), motion chain codes, TMDHMM networks, hierarchical feature characterization, and wearable analysis (PCA). Each method advances the field by addressing a different facet of ASL recognition while enhancing the overall effectiveness of the system.

II. Methodology

A. Sign Language

SL [19] is a visual and gesture communication method used by deaf and hard of hearing people. It functions as a whole language with distinct syntax, vocabulary, and grammar. In contrast to spoken languages, SL uses a variety of hand gestures, facial expressions, body movements, and spatial signals to express meaning visually [20]. Hand forms are of utmost importance in sign language (SL), especially when conveying alphabet letters. Signers de- pend on visually representing individual letters using specific handshapes or configurations. For the intended letters to be correctly communicated, the body's position, motion, and alignment with the face or hands are essential. Signers can efficiently express the alphabet, words, names, and special concepts or terms by using these hand gestures to spell or refer to them.

The Fig. 1 aims to convert some symbols into ASL.

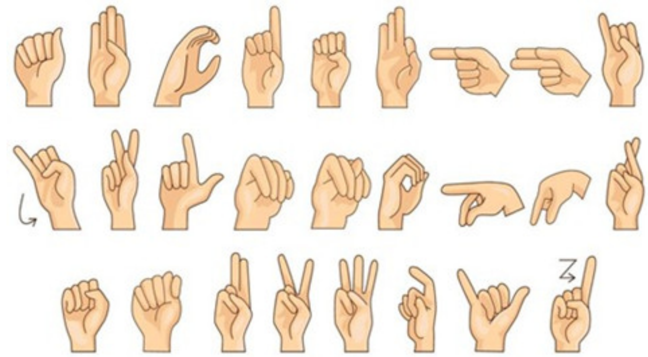

Fig. 1. Sign language

B. Proposed Model for ASL

Our research focuses on ASL, and to analyze and categorize ASL movements, we use a variety of DL models. Data preprocessing, the first step in our study approach, entails converting images into a predetermined color scheme, scaling them to a uniform resolution, and normalizing the data within a predetermined range.

In order to assess the performance of the models, we partitioned the dataset into distinct training and testing subsets. The training set was utilized to train the models, while the test set was employed as a separate and independent dataset to evaluate the models' ability to generalize to unseen data.

Our study investigates various DL model architectures, such as VGG19, DenseNet, ResNet50, MobileNet, and NASNet-Mobile, to analyze their effectiveness in recognizing ASL gestures. Each model possesses distinctive features and capabilities, enabling us to assess their performance and impact on this specific task. During the training process, the model parameters are opti- mized using techniques such as backpropagation and gradient descent. This repeated procedure allows the models to learn and extract significant elements from the ASL gesture data, boosting their prediction skills. The models are evaluated using the test set after training. The models are fed the test set data to produce predicted labels for each ASL gesture. The ACC and overall performance of the models are then evaluated by comparing these predictions to the ground truth labels.

We use numerous performance indicators, including ACC, PREC, recall REC, and F1-S, to conduct a thorough review. These measures [21] allow us to evaluate the models' ACC in identifying ASL gestures and their usefulness in ASL recognition tasks. These metrics provide insight into the models' overall performance as well as their ability to correctly identify and classify ASL gestures. Our goal in using this research methodology is to gain a better knowledge of the capabilities and limits of different DL models in ASL gesture detection. We hope that our trials and analyses will help to advance ASL

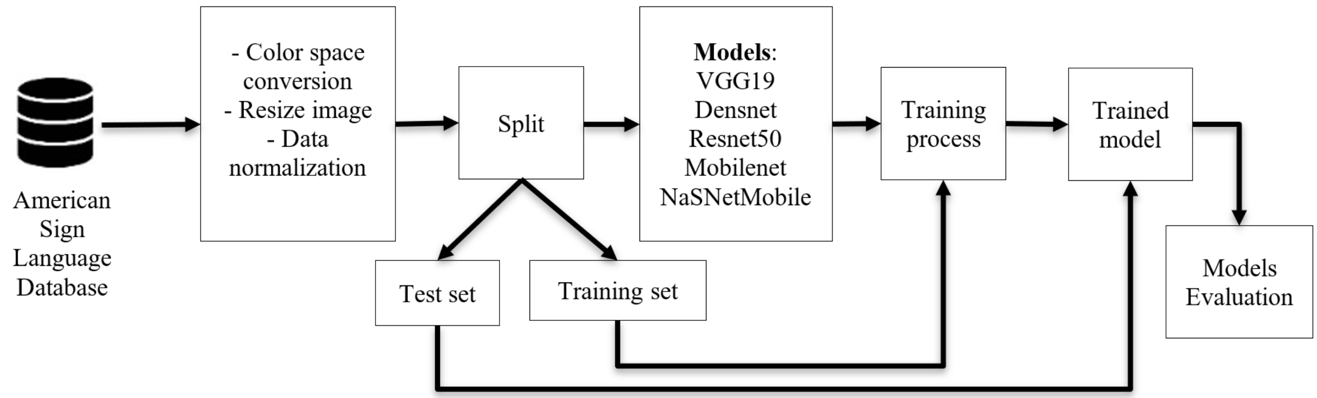

Fig. 2. Methodology of ASL

recognition systems and provide significant insights for future research and practical applications in SL interpretation. Fig. 2 depicts the approach used in this investigation.

C. Description of the Dataset

For our study, we utilized the ASL Alphabet dataset [22], which encompasses a diverse assortment of images showcasing the ASL alphabet letters. In ASL, every letter of the American alphabet corresponds to a distinct hand gesture. This compilation includes images of individuals displaying hand signs for each letter of the ASL alphabet. The images were captured under different lighting conditions and against diverse backgrounds to provide a comprehensive depiction, encompassing a broad spectrum of visual variations. This dataset is apt for training and evaluating accuracy models for tasks involving ASL recognition.

The ASL Alphabet dataset includes labeled images with the corresponding ASL alphabet letter for each sample, enabling the application of supervised learning techniques. The avail- ability of these labels makes it easier to create and test models that can correctly categorize hand actions using the supplied photos. The dataset includes additional labels for space and the gesture ”nothing” in addition to all letters from A to Z. The inclusion of these labels ensures comprehensive coverage of the ASL alphabet and enhances the dataset's suitability for training and testing ML models. Additionally, People interested in ASL recognition and SL interpretation will find this dataset to be a useful resource, offering ample opportunities for experimentation, algorithm development, and advancements in the field. Furthermore, the dataset holds potential for educational purposes, as it can be used in educational contexts to actively engage students and enthusiasts in practical ML projects. Working with the ASL Alphabet dataset allows learners to explore and gain knowledge about the ASL alphabet while actively participating in real-world ML endeavors.

D. Exploratory Data Analysis

EDA [23] is an essential data analysis procedure that involves the examination and understanding of a dataset prior to applying statistical techniques or models. Its primary objective is to discover patterns, correlations, and anomalies within the data to gain valuable insights and make well-informed decisions. By performing EDA, analysts can uncover important characteristics of the dataset, identify trends, assess data quality, and determine appropriate approaches for subsequent analysis.

Each image in the dataset boasts a resolution of 400 pixels in height and 400 pixels in width, creating a perfect square layout that lends itself to a uniform presentation across the dataset. These dimensions ensure that each image has a total of 160,000 pixels, allowing for detailed representation of the content. Furthermore, every image contains three distinct color channels: red, green, and blue, commonly abbreviated as RGB [24].

This RGB format is essential for recreating a vast spectrum of colors by combining these primary channels in various intensities. The format, which is typically denoted as (400, 400, 3), provides a clear depiction of both the spatial dimensions and the color composition of the images. In a practical sense, this consistent structure not only aids in image processing tasks but also ensures that any operations or transformations applied to one image can be uniformly applied to others in the dataset.

E. Data Preprocessing

Preprocessing image data plays a crucial role in preparing it for analysis and modeling, ensuring its compatibility with ML tasks. This involves applying various preprocessing techniques to enhance the quality and suitability of the data for further analysis and model training.

Fig. 3. Illustration of ASL dataset

- Image resizing [25], which involves changing an image's proportions while maintaining its aspect ratio, is a key approach in image processing and computer vision. This method is frequently used to uniformly scale photos for many applications, including ML algorithms or display purposes. The pixels in the image are resampled to a new resolution during the resizing process. When shrinking the image, methods like decimation or interpolation are used to generate new values or average nearby pixels. This successfully reduces the size of the image while ensuring that crucial information is kept. In this research, the images are resized using the following dimensions: IMAGE SIZE = (224, 224). Consequently, the images are reshaped to have dimensions of (224, 224, 3), where the third dimension represents the three color channels (red, green, and blue). This resizing ensures that all images in the dataset have a consistent size and can be effectively processed and analyzed using the chosen ML techniques.

- Normalization [26] is a fundamental preprocessing technique employed in image analysis and ML tasks. It involves standardizing the pixel values of an image within a specific range to ensure fair comparisons and enhance the effectiveness of model training. By normalizing pixel intensities, variations in brightness, contrast, and overall intensity are minimized, allowing models to learn mean- ingful patterns more effectively. Normalization plays a crucial role in achieving consistent and reliable results when working with image data.

  To normalize the image data in this research, a common approach is employed: dividing each pixel value of the training images by 255.0. This rescaling operation ensures that the pixel values are within the range of 0 to 1. By performing this normalization technique, the image data is brought into a consistent and standardized range, which is beneficial for ML algorithms.

- Splitting data: Within the scope of image evaluation and ACC applications, the data preprocessing stage involves segmenting the image data into distinct partitions, referred to as the training and the test sets. This segmentation is done with the aim of examining the models' generalization capabilities and performance metrics using unfamiliar data. By segregating the data into these two sets, we can educate the models on a subset of the data, and subsequently gauge their capacity to analyze and predict fresh images that were excluded during the training phase.

  The dataset is divided into two subsets: the training set and the testing set. Each subset comprises 50% of the total dataset, ensuring an equal distribution of data for training and evaluation purposes. The Fig. 4 presents the length of train ad test labels.

F. Models Implementation

In this article, we will employ five different models for ASL gesture classification: VGG19, DenseNet, ResNet50, MobileNet, and NASNetMobile. These models are selected for their unique architectures and capabilities, providing a comprehensive analysis of their performance in ASL recognition tasks. By utilizing multiple models, we aim to gain a deeper understanding of their strengths and limitations, ultimately contributing to the advancement of ASL recognition systems.

- **VGG19 model**: it [27] is a widely used deep CNN architecture that has demonstrated excellent performance in various computer vision tasks, including ASL recognition. Its architecture focuses on obtaining rich and extensive visual information from photos, making it ideal for applications like ASL image classification and identi- fication. By leveraging its deep layers and convolutional filters, the VGG19 model can effectively learn and recognize intricate patterns and features within ASL gesture images.

- **DenseNet model**: The DenseNet model [28] is a deep CNN architecture characterized by a unique connection topology and feature reuse system. With regard to ASL identification among other computer vision tasks, it has demonstrated impressive performance. The DenseNet's dense connection, which supports efficient feature learning and aids in the precise classification of ASL hand movements, is one of its unique features. The model can capture both local and global information by utilizing vast linkages across layers, which enables it to recognize and decipher the numerous nuances of ASL motions. Furthermore, the DenseNet's effective parameter sharing improves its ability to extract significant features from ASL images, making it particularly well suited for ASL recognition tasks.

- **ResNet50 model**: This refers to an advanced CNN design that is widely used in a variety of computer vision applications, including the recognition of ASL. Its deep structure and innovative implementation of residual connections, which successfully tackle the vanishing gradients issue, are highly praised, allowing for the training of more in-depth networks. The elaborate design of the ResNet50 model allows it to detect complex features and patterns in ASL images, contributing to improved ACC and performance in ASL recognition tasks. By exploiting its profound layers and residual connections, the model is able to efficiently learn and embody the intricate visual properties of ASL hand movements.

- **MobileNet model**: This is a profound CNN blueprint, honed for operation in resource-limited devices. It presents an optimal trade-off between the model's scale and performance, making it an ideal choice for ASL detection tasks. The methodology of MobileNet [29] prioritizes capturing critical components and trends in ASL hand gestures while minimizing the computational resources needed for inference. This aspect positions it as an excellent selection for ASL recognition applications on mobile and embedded systems where efficiency and real-time response are pivotal. The lean design of the MobileNet model facilitates efficient ASL image processing while preserving high ACC, qualifying it as an outstanding choice for ASL recognition in resource-restricted scenarios.

- **NaSNetMobile model**: In order to automatically find the best network configurations, it [30] uses neural archi- tecture search (NAS) methods. Stacks of repeated cells are used in this creative design to promote successful learning. The NaSNetMobile model exhibits the capacity to precisely recognize and classify hand movements into their associated ASL letters or sentences, making it particularly well-suited for ASL identification tasks. The NaSNetMobile model may automatically identify and incorporate architectural solutions that improve perfor- mance in ASL recognition by utilizing the capabilities of NAS.

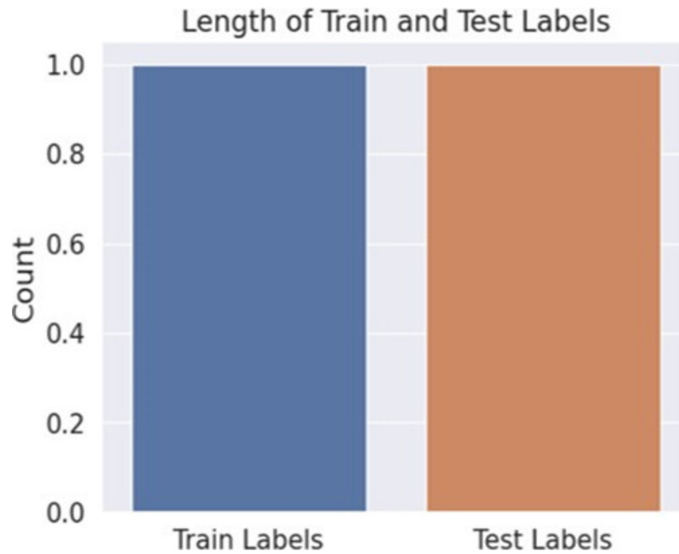

Fig. 4. Length of train and test sets

III. Results

We used the Collab Pro cloud-based computing platform to make our experimental design easier. We had access to a high-performance computing setting with potent GPUs thanks to this platform. Leveraging this computing power was essential for efficiently training the complex DL models required for our research. By utilizing the computational resources offered by Collab Pro, we were able to conduct our trials without the need for costly local hardware deployments. This cost-effective and readily accessible solution provided us with a seamless and efficient platform to carry out our experiments.

Our study revolved around the exploration and evaluation of various ML models using the ASL dataset.

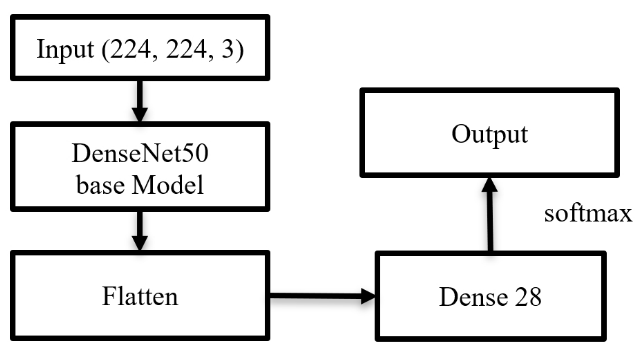

Fig. 5. VGG19 model

This dataset, encompassing a broad range of hand gestures, offered a unique platform to test the efficacy and adaptability of modern com- putational approaches. The primary objective was to employ these models to accurately classify different categories of alphabets and other signs integral to ASL. Recognizing the nuances and intricacies of sign language through automated systems remains a challenging endeavor. With this research, we aimed to push the boundaries of what's possible in ASL recognition, fostering a fusion of technology and human communication.

- **VGG19 model**: [31] The structure of the model is composed of several hierarchical layers as shown in the Fig. 5, starting with an initial layer encompassing 256 neurons utilizing the Rectified Linear Unit (ReLU) activation mechanism. This is succeeded by a layer holding 128 neurons applying the same ReLU activation, followed by an additional layer featuring 64 neurons using ReLU as well. The terminating layer is a Dense layer, with unit quantity aligning with the task's class quantity, and it formulates class probabilities via the SoftMax activation approach. The model construction comprises an input layer that's shaped by the image input dimensions and an output layer determined by the final Dense layer.

  In terms of optimization, the model takes advantage of the Adam optimization technique and employs a batch size of 256 during the learning phase. The learning process is split into 15 epochs, each signifying a thorough cycle of the learning dataset.

- **DenseNet201 model**: A pre-trained model with ImageNet weights is loaded and its layers are locked in this model as presented in the Fig. 6. A Flatten layer is implemented to turn the DenseNet201 model's output into a 1D feature vector. Then, for multi-class classification, a Dense layer with softmax activation is added. The model is trained for 20 epochs with a batch size of 64 using the Adam optimizer.

  In terms of optimization, the model takes advantage of the Adam optimization technique and employs a batch size of 256 during the learning phase. The learning process is split into 15 epochs, each signifying a thorough cycle of the learning dataset. The model is trained for 20 epochs with a batch size of 64 using the Adam optimizer.

- **ResNet50 model**: The model in this implementation makes use of pre-trained ImageNet weights. The underlying model is frozen, with all of its layers remaining unchanged, and a custom categorization layer is built on top. Softmax activation, sparse categorical cross-entropy loss, and the Adam optimizer are used to build the model. A batch size of 64 is employed during training, and the model is trained for 15 epochs. To assess the model's performance, a validation split of 0.2 is used. This method combines transfer learning with a programmable classi- fication layer to produce accurate image categorization. The Fig. 7 presents the architecture applied in this model.

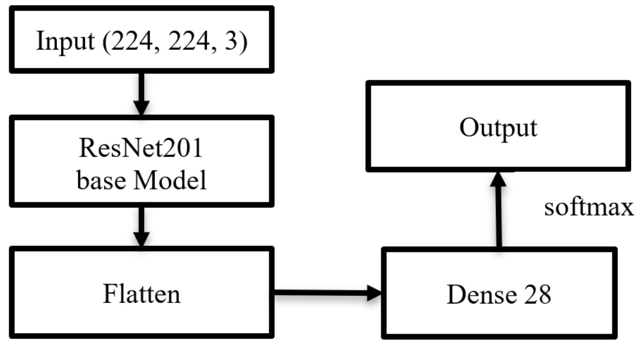

Fig. 6. DensNet model

- **MobileNet model:** The model is further elaborated by incorporating supplementary layers on top of the primary model. These layers as shown in the Fig. 8 encompass a Flatten layer, subsequently followed by a Dense layer that utilizes softmax activation for categorization tasks. The model applies sparse categorical cross-entropy loss as its loss function, the Adam optimizer for optimiza- tion purposes, and ACC as its performance assessment metric. Detailed specifications of the model, inclusive of the count of trainable parameters, are provided. A graphic depiction of the model is also offered for better comprehension. Regarding training, the model undergoes a learning process on a particular dataset for a total of 15 epochs, using a batch size of 64. For the purpose of evaluating the performance of the model, a validation split of 0.2 is employed.

- **NaSNetMobile model**: The model use ImageNet pretrained weights and does not include fully connected layers. Instead, a bespoke top layer for classification is implemented, consisting of a flatten layer followed by a dense layer with softmax activation as seen in the Fig. 9. The Adam optimizer is used to construct the model with sparse categorical cross-entropy loss. The ACC metric is monitored throughout training. The dataset is utilized to train the model for 15 epochs with a batch size of 64 and a 20% validation split.

IV. DISCUSSION

The Table I illustrates the outcomes achieved by these models.

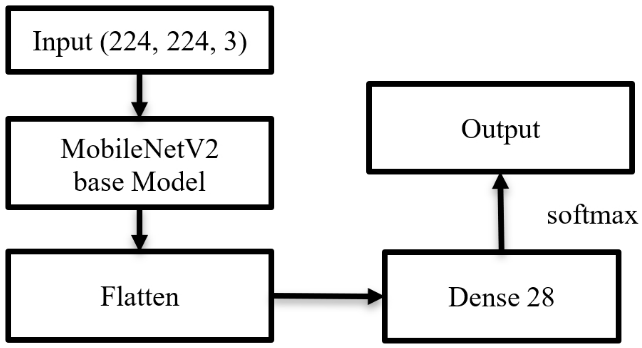

Fig. 7. ResNet model

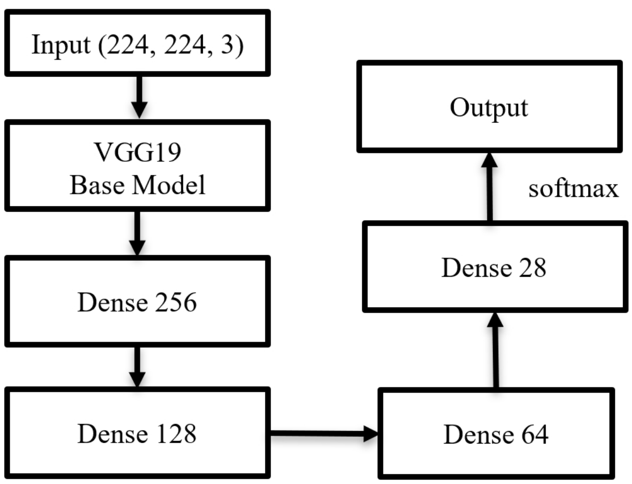

Fig. 8. MobileNet model

During the evaluation of various models for ASL classification, including VGG19, DenseNet, ResNet50, MobileNet, and NASNetMobile, several notable observations were made. The models demonstrated excellent performance, with high PREC, REC, and F1-Ss across most classes. ResNet50 got a slightly lower ACC of 98 percent compared to VGG19, DenseNet, MobileNet, and NASNetMobile. All models also showed remarkable average sensitivity and specificity, properly identifying both positive and negative events. This implies that these models exhibit a remarkable ability to distinguish between different ASL gestures, offering promising applica- tions in real-time ASL communication systems and assistive technologies for the deaf and hard of hearing. These models displayed outstanding ACC and were found to be quite good at classifying ASL signs. The model selection process should take into account certain needs, such as computational capabil- ities, model size, or real-time performance. The best model for a certain application can be found through additional research.

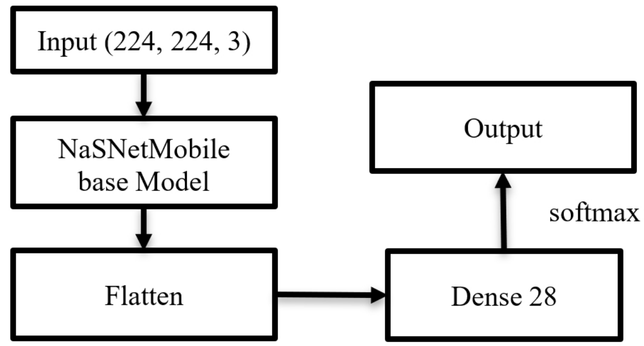

Fig. 9. MobileNet model

TABLE I.
COMPARATIVE TABLE

Model	PREC	REC	F1-S	Sensitivity	Specificity
MobileNet	1.00	1.00	1.00	1.00	1.00
VGG19	1.00	1.00	1.00	1.00	1.00
NASNetMobile	1.00	1.00	1.00	1.00	1.00
ResNet50	0.99	0.99	0.99	0.99	0.99
DenseNet	1.00	1.00	1.00	1.00	1.00

V. CONCLUSION

This article looks into the use of ML models for ASL recognition to get around difficulties with correctly deciphering and translating ASL motions. Utilizing metrics like ACC, PREC, REC, and F1-S, the performance of DL models such as VGG19, DenseNet, ResNet50, MobileNet, and NASNet-Mobile was evaluated. The outcomes show how well the models categorize ASL gestures, underscoring their potential to enhance communication between the deaf and hearing communities. Every model has advantages and disadvantages that vary based on real-time performance and computational resources, for example. To improve model robustness and ACC, future research should concentrate on tackling issues linked to differences in signing methods and illumination conditions. In areas like education and employment, ML algorithms and DL models hold immense promise for inclusivity and empowerment. Future research should also look into multimodal techniques that combine visual, textual, and audio information, as well as real-time recognition and handling variances. For future work, there must be collaboration with the Deaf community. ASL recognition could undergo a revolution thanks to continued improvement in this area, which will benefit deaf people everywhere.

CONFLICT OF INTEREST

The authors have no conflict of relevant interest to this article.

REFERENCES

[1] WHO, "The world health organization estimates 2.5 billion with hearing loss by 2050," 2024. https://www.pacifichearingcare.com/blog/the-world-health-organization-estimates-2-5-billion-with-hearing-loss-by-2050/.

[2] C. Bakersfield, "American sign language," 2023. https://www.bakersfieldcollege.edu/ academics/pathways/ach/asl/index.html.

[3] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, "Machine learning methods for sign language recognition: A critical review and analysis," *Intelligent Systems with Applications*, vol. 12, p. 200056, 2021.

[4] K. Amrutha and P. Prabu, "Ml based sign language recognition system," in *2021 International Conference on Innovative Trends in Information Technology (ICITIIT)*, (Kottayam, India), pp. 1–6, IEEE, 2021.

[5] M. M. Zaki and S. I. Shaheen, "Sign language recognition using a combination of new vision based features," *Pattern Recognition Letters*, vol. 32, no. 4, pp. 572–577, 2011.

[6] C.-H. Chuan, E. Regina, and C. Guardino, "American sign language recognition using leap motion sensor," in *2014 13th International Conference on Machine Learning and Applications*, (Detroit, MI, USA), pp. 541–544, IEEE, 2014.

[7] C. Dong, M. C. Leu, and Z. Yin, "American sign language alphabet recognition using microsoft kinect," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, (Boston, MA, USA), pp. 44–52, 2015.

[8] D. Aryanie and Y. Heryadi, "American sign language-based finger-spelling recognition using k-nearest neighbors classifier," in *2015 3rd International Conference on Information and Communication Technology (ICoICT)*, (Nusa Dua, Bali, Indonesia), pp. 533–536, IEEE, 2015.

[9] J. R. Pansare and M. Ingle, "Vision-based approach for american sign language recognition using edge orientation histogram," in *2016 international conference on image, vision and computing (ICIVC)*, (Portsmouth, UK), pp. 86–90, IEEE, 2016.

[10] V. N. Truong, C.-K. Yang, and Q.-V. Tran, "A translator for american sign language to text and speech," in *2016 IEEE 5th Global Conference on Consumer Electronics*, (Kyoto, Japan), pp. 1–2, IEEE, 2016.

[11] A. Joshi, H. Sierra, and E. Arzuaga, "American sign language translation using edge detection and cross correlation," in *2017 IEEE Colombian Conference on Communications and Computing (COLCOM)*, (Cartagena, Colombia), pp. 1–6, IEEE, 2017.

[12] T. Starner, *Visual recognition of american sign language using hidden markov models*. PhD thesis, Massachusetts Institute of Technology, Cambridge, United States, 1995.

[13] L.-G. Zhang, Y. Chen, G. Fang, X. Chen, and W. Gao, "A vision-based sign language recognition system using tied-mixture density hmm," in *Proceedings of the 6th international conference on Multimodal interfaces*, (New York, United States), pp. 198–204, 2004.

[14] Y. Cui and J. Weng, "Appearance-based hand sign recognition from intensity image sequences," *Computer Vision and Image Understanding*, vol. 78, no. 2, pp. 157–176, 2000.

[15] R. Akmeliawati, M. P.-L. Ooi, and Y. C. Kuang, "Real-time malaysian sign language translation using colour segmentation and neural network," in *2007 IEEE Instrumentation & Measurement Technology Conference IMTC 2007*, (Warsaw, Poland), pp. 1–6, IEEE, 2007.

[16] R.-H. Liang and M. Ouhyoung, "A real-time continuous gesture recognition system for sign language," in *Proceedings third IEEE international conference on automatic face and gesture recognition*, (Nara, Japan), pp. 558–567, IEEE, 1998.

[17] Q. Munib, M. Habeeb, B. Takruri, and H. A. Al-Malik, "American sign language (asl) recognition based on hough transform and neural networks," *Expert systems with Applications*, vol. 32, no. 1, pp. 24–37, 2007.

[18] A. Nandy, J. S. Prasad, S. Mondal, P. Chakraborty, and G. C. Nandi, "Recognition of isolated indian sign language gesture in real time," in *Information Processing and Management: International Conference on Recent Trends in Business Administration and Information Processing, BAIP 2010, Trivandrum*, (Kerala, India), pp. 102–107, Springer, 2010.

[19] M. S. Altememe and N. K. El Abbadi, "A review for sign language recognition techniques," in *2021 1st Babylon International Conference on Information Technology and Science (BICITS)*, (Babil, Iraq), pp. 39–44, IEEE, 2021.

[20] U. Languages, "Sign language's only related to making signs with the hands, is this true?," 2023.

https://www.languagesunlimited.com/sign-languages-only-related-to-making-signs-with-the-hands-is-this-true.

[21] L. Ferrer, "Analysis and comparison of classification metrics," *arXiv:2209.05355*, 2022.

[22] A. Grassknoted, "Asl alphabet: Image data set for alphabets in the american sign language," 2018. https://www.kaggle.com/datasets/grassknoted/asl-alphabet.

[23] S. Morgenthaler, "Exploratory data analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 1, no. 1, pp. 33–44, 2009.

[24] G. D. Finlayson, B. Schiele, and J. L. Crowley, "Comprehensive colour image normalization," in *Computer Vision—ECCV'98: 5th European Conference on Computer Vision Freiburg, Germany, June, 2–6, 1998 Proceedings, Volume I 5*, pp. 475–490, Springer, 1998.

[25] S. Saponara and A. Elhanashi, "Impact of image resizing on deep learning detectors for training time and model performance," in *International Conference on Applications in Electronics Pervading Industry, Environment and Society*, pp. 10–17, Springer, 2021.

[26] S. Patro and K. K. Sahu, "Normalization: A preprocessing stage," *arXiv:1503.06462*, 2015.

[27] V. Sudha and T. Ganeshbabu, "A convolutional neural network classifier vgg-19 architecture for lesion detection and grading in diabetic retinopathy based on deep learning.," *Computers, Materials & Continua*, vol. 66, no. 1, 2021.

[28] R. Daroya, D. Peralta, and P. Naval, "Alphabet sign language image classification using deep learning," in *TENCON 2018-2018 IEEE Region 10 Conference*, (Jeju, Korea (South)), pp. 0646–0650, IEEE, 2018.

[29] T. N. Abu-Jamie and S. S. Abu-Naser, "Classification of sign-language using mobilenet-deep learning," *International Journal of Academic Information Systems Research (IJAISR)*, vol. 6, no. 7, pp. 29–40, 2022.

[30] P. Kajonpong, *Recognizing American Sign Language Using Deep Learning*. Texas, US: The University of Texas at San Antonio, 2019.

[31] D. Kumari and R. Anand, "Static alphabet american sign language recognition using convolutional neural networks," in *2022 International Conference on Advances in Computing, Communication and Materials (ICACCM)*, (Dehradun, India), pp. 1–6, IEEE, 2022.