IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Revue d'Intelligence Artificielle
Vol. 38, No. 3, June, 2024, pp. 929-938
Journal homepage: http://iieta.org/journals/ria

Alphabet Recognition in Sign Language Using Deep Learning Algorithm with Bayesian Optimization

Antonio Josef*, Gede Putra Kusuma

Computer Science Department, BINUS Graduate Program - Master of Computer Science, Bina Nusantara University, Jakarta 11480, Indonesia

Corresponding Author Email: antonio.josef@binus.ac.id

https://doi.org/10.18280/ria.380319

Received: 10 April 2024
Revised: 15 May 2024
Accepted: 10 June 2024
Available online: 21 June 2024

Keywords:
alphabet recognition, sign language recognition, hand gesture recognition, deep learning, Bayesian Optimization

ABSTRACT

Sign language, a vital medium for communication, particularly for individuals with speech and hearing impairments, is gaining recognition for its efficacy. To evaluate the efficacy of sign language alphabet recognition systems, three prominent image classification deep learning models—ResNeXt101, VGG19, and ViT—were chosen due to their established relevance and popularity in the field. The study aimed to identify the most effective model for accurate and efficient sign language classification using the NUS hand posture dataset-II. The study utilized Bayesian optimization for hyperparameter tuning, recognizing its superiority in systematically exploring the hyperparameter space compared to other optimization methods. This approach significantly enhanced the performance of the models by tailoring their configurations, leading to improved accuracy and robustness in sign language recognition across various experimental scenarios. While the findings consistently favored ResNeXt101 over VGG19, with a notable 2% higher F1 score, ViT also showcased comparable performance in certain experiments, achieving an impressive F1 score of 99%. Despite these successes, the study encountered limitations, including dataset bias and generalization challenges, which underscore the need for further research in this domain to address these complexities.

1. INTRODUCTION

Human interaction is profoundly shaped by the vital process of communication, a dynamic phenomenon governed by several essential components. At its core, communication hinges on the presence of a communicator responsible for transmitting a message, a receptive entity in the form of a receiver, the medium or channel through which information is conveyed, and a crucial feedback loop through which the recipient responds to the conveyed message. These fundamental elements collectively underpin the intricate fabric of human socialization [1].

Effective communication encompasses a multifaceted process that employs a diverse array of mediums. This process extends beyond direct verbal exchanges to encompass a spectrum of communication tools, which serve as conduits for conveying messages. Such tools encompass both traditional and modern technologies, including landline and mobile phones, email, social media platforms, intercom systems, short message service (SMS), as well as versatile cross-platform instant messaging applications [2].

In numerous instances, individuals encounter challenges in conventional communication as a consequence of various factors, including those afflicted with speech impairments characterized by compromised articulation or an inability to vocalize. Additionally, a subset of the population comprises deaf individuals who experience hearing impairments either from birth or later in life, rendering them incapable of engaging in standard speech practices, especially during infancy or early childhood, wherein their linguistic aptitude lags cognitive development. In addressing these challenges, a system of sign language emerges, founded on intricate body language that amalgamates hand gestures, movements, lower limb actions, and overall body expressions, complemented by nuanced facial expressions to effectively convey messages, as eloquently exemplified in the study by [3].

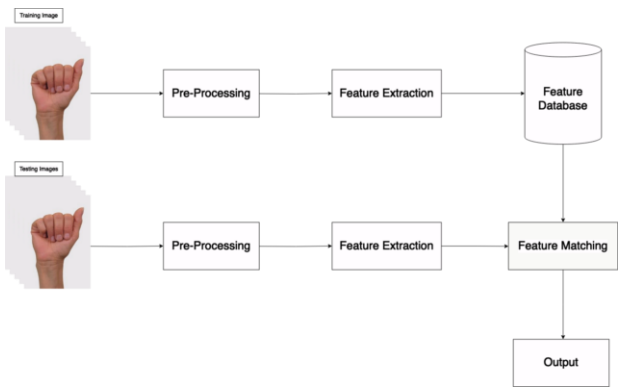

Figure 1. Process on hand gesture recognition

Sign language, traditionally associated with communication

for individuals with speech and hearing impairments, extends its utility beyond this realm. Beyond its crucial role in the lives of those with impairments, sign language serves as a versatile means of communication for diverse groups. This form of non-verbal communication finds application within the military, where it enables seamless interaction between troops amid battlefield operations [4]. Additionally, infants and toddlers employ sign language to convey their intentions and desires before developing verbal proficiency.

In Figure 1, hand gesture recognition involves training a model with a dataset of hand gestures, preprocessing frames to enhance quality, segmenting the hand, and extracting relevant features like finger positions and hand shapes. The trained model's feature set is saved for future recognition. During testing, new gestures undergo similar preprocessing, and their features are matched with the saved ones. This process classifies the gestures into predefined categories, enabling practical interaction in applications from virtual reality to human-computer interfaces.

Sign Language Recognition, a groundbreaking advancement, aims to bridge communication barriers for the hearing-impaired. Researchers have explored various aspects of SLR, from data acquisition to feature extraction and classification methods. Despite progress, cost remains a challenge [5], spurring efforts toward affordable solutions. Different approaches are being investigated, including visual-based gesture recognition and hybrid classification methods like CNN-HMM and full deep learning [6]. Challenges persist, such as overfitting and computational demands [7]. Deep learning models like AlexNet [8], VGG19 [9], CNN [10], and LSTM [6] are utilized for feature extraction and classification, showing promising results [8-11].

In recent times, there has been a proliferation of innovative models, including ResNeXt101, VGG19, and ViT, which have demonstrated commendable performance in image classification tasks. In this study, the researcher seeks to compare and evaluate the efficacy of these three models in the context of sign language recognition. ResNeXt, known for its depth and parallelism, excels at capturing intricate features in sign language gestures, which involve subtle variations in hand shapes, movements, and facial expressions. Its hierarchical representations through residual connections suit the complexity of sign language. VGG, while simpler than newer architectures, has shown impressive performance in tasks like ImageNet classification. Its straightforward architecture strikes a balance between simplicity and effectiveness, in sign language recognition, where interpretability and generalization matter, VGG may reveal crucial features of sign gestures. The vision transformer differs from CNNs by using self-attention to capture long-range dependencies in images, offering a fresh perspective on image understanding for sign language gestures. Its scalability and handling of variable-length sequences align well with the diverse nature of sign language gestures.

The specific objectives of this study encompass two main aspects. Firstly, we seek to implement these deep learning models in the context of sign language image classification, aiming to ascertain whether they can surpass the performance of previous methodologies. This involves not only assessing their accuracy but also considering factors such as computational efficiency and scalability, crucial for real-world deployment. Secondly, we aim to explore the efficacy of Bayesian optimization hyperparameter tuning techniques in enhancing the performance of these models. By systematically optimizing the hyperparameters of ResNeXt, VGG, and the vision transformer, we aim to identify the most optimal configurations for each model, thereby potentially improving their accuracy and generalization capabilities in sign language recognition tasks. Through these endeavors, this research endeavors to address the following key research questions: Can ResNeXt, VGG, and the vision transformer outperform existing methodologies in sign language recognition, thus advancing the state-of-the-art in this domain? Furthermore, can the utilization of Bayesian optimization techniques contribute to obtaining the best parameters for these deep learning models, ultimately enhancing their performance and efficacy in sign language recognition? By elucidating these questions, we aspire to contribute to the development of more robust and accurate sign language recognition systems, ultimately fostering greater inclusivity and accessibility for individuals with hearing impairments.

This study utilizes pre-trained models like ResNeXt101, VGG19, and ViT to enhance hand posture recognition in computer vision and human-computer interaction fields. By refining hyperparameters and using Bayesian optimization, we aim to optimize model performance, potentially guiding future research. Ultimately, we seek to advance sign language recognition for more accurate and efficient applications. The societal and technological impact of improving sign language recognition systems extends far beyond the realm of assistive technology. It encompasses broader goals of social inclusion, technological innovation, and advancing the frontiers of artificial intelligence. By addressing these critical challenges, the research not only enhances accessibility and empowerment for individuals with hearing impairments but also contributes to building more inclusive and equitable societies driven by technological progress.

2. RELATED WORKS

Hand gesture recognition has evolved significantly over time, starting from basic image processing techniques in the 1960s to sophisticated deep learning models today. In the 1980s and 1990s, machine learning techniques like neural networks and Hidden Markov Models (HMMs) were explored, but were limited by computational resources. The early 2000s saw a leap with Convolutional Neural Networks (CNNs), which improved accuracy by learning features directly from pixel data. Large annotated datasets further fueled progress. Recent advancements include recurrent neural networks (RNNs), attention mechanisms, and graph neural networks (GNNs), enabling models to capture temporal and spatial dependencies for better accuracy. Integration of depth sensors like Kinect and RealSense has enhanced 3D gesture recognition in real-world scenarios.

Currently, state-of-the-art hand gesture recognition methods leverage of deep learning architectures, large-scale annotated datasets, and advanced sensor technologies to achieve high accuracy and robustness across a wide range of applications, including human-computer interaction, sign language recognition, and virtual reality interfaces.

A notable example of this is the fusion approach, which involves the amalgamation of Histograms of Oriented Gradients (HOG) for hand shape depiction and Local Binary Pattern (LBP) for hand texture description, coupled with a Support Vector Machine (SVM) employing a radial basis function as its kernel. This method has demonstrated

significant promise in enhancing the classification performance for hand gesture recognition, as discussed in the study [12]. Empirical evaluations conducted on the NUS hand posture dataset-II illustrate that the proposed HOG-LBP features significantly enhance hand recognition accuracy, achieving an average recognition accuracy of 97.8% for Subset A and 95.07% for subset B. The method proposed showing the strength of achieving a better classification performance on hand gesture recognition. Nevertheless, there remains a weakness to augment the speed and efficiency of hand gesture recognition, with a specific focus on complex background scenarios.

In a study focusing on the recognition of Indonesian Sign Language (BISINDO), researchers employed deep learning techniques to analyze results from a dataset of 1100 images across 10 categories [11]. The dataset is split into subsets A (1000 images for training) and B (100 for validation), with preprocessing techniques applied before training. Results show CNN achieving 73% accuracy, LSTM 81%, and a fusion of CNN and LSTM reaching 96% accuracy with a 17% loss. This highlights the synergy between CNN and LSTM in improving sign language recognition systems, a key strength. However, limitations include the dataset's scope, lacking comprehensive coverage of all symbols. Future research should focus on expanding the dataset and exploring additional aspects like expression detection and body gesture recognition to enhance overall interpretation systems.

In a comprehensive investigation, researchers analyzed an American Sign Language dataset, which consisted of approximately 205 images per numerical class (0 to 9) at 100x100 resolution [13]. A sophisticated ten-layer Convolutional Neural Network (CNN) was deployed, showcasing a strong 87.5% accuracy rate. This demonstrates the effectiveness of the tailored approach. Additionally, preprocessing reduces computational complexity, aiding efficiency. Manual hyperparameter fine-tuning ensures optimal performance, suitable for mobile applications and embedded systems. However, scalability and generalizability to other datasets or tasks beyond ASL recognition require further investigation, marking a potential weakness.

In the realm of hand gesture recognition, an extensive study explored the use of convolutional neural networks (CNNs) in recognizing American Sign Language (ASL) gestures [14]. It involves a dataset of 2515 200x200 pixel images of ASL numerical digits and alphabet characters. The CNN architecture includes four layers: two convolutional layers, batch normalization, a pooling layer, dropout layers, and flattening operation, followed by two fully connected layers, each with dropout. This approach achieves an impressive accuracy of 94.34% in ASL gesture recognition. Its strength lies in its accessibility via a simple camera and scalability through continuous training. Moreover, it offers a cost-effective alternative to specialized cameras like Microsoft's Kinect. However, reliance on pre-made datasets may limit adaptability to diverse real-world scenarios. Further research could enhance robustness to lighting conditions and hand positions.

In a study conducted by researchers [15], the Multi-scale Cross Feature Aggregation Network (Cross-Feat) achieved remarkable accuracy rates of 98.33% on the NUS hand posture dataset-ii and 99.5% on the ASL fingerspelling dataset, showcasing its strength in hand posture recognition and ASL fingerspelling. CrossFeat’s efficient design and effective feature preservation enable precise gesture recognition by capturing both local and global information. With only 975K trainable parameters, it is lightweight and computationally efficient. However, its performance in more complex environments beyond benchmark datasets is a potential weakness, necessitating further testing in diverse real-world scenarios to assess its robustness and generalizability.

The researchers conducted a series of experiments utilizing Moore's algorithm for edge detection and Fourier descriptor transform for feature extraction. These experiments, outlined in the study [16], achieved a notable 96.92% accuracy in recognizing Indonesian sign language alphabet, BISINDO, showcasing its potential for broader applications in image processing and recognition. The strength of Fourier descriptors lies in their effective feature extraction, but a weakness arises in their performance with translated and rotated images. Future work should focus on improving accuracy under such transformations, possibly through refining feature extraction or employing more robust distance metrics.

In a pioneering study conducted by researchers [17], an innovative approach to SIBI sign language recognition was introduced, using a cutting-edge 3D-CNN methodology. This study employed advanced camera technology to analyze a self-collected dataset, leading to a significant breakthrough. The strength lies in its ability to internally gather a large and varied dataset, a departure from conventional techniques. The researchers achieved an impressive accuracy rate of 97.5% with their model. This underscores the potential of 3D-CNN technology and innovative data collection strategies. The strength of transfer learning here lies in its effectiveness with small datasets, as evidenced by the significant accuracy improvement. However, a potential weakness is its sensitivity to variations in data distribution or domain shifts, which could affect generalization across different datasets or real-world scenarios. Thus, careful consideration and further evaluation are needed to ensure the method's robustness and adaptability beyond the study's specific context.

In their comprehensive research, the authors conducted extensive experiments on the BISINDO dataset, employing various methodologies to explore sign language recognition [8]. They used deep learning models like AlexNet and VGG-16, achieving high accuracy rates of 98.6% and revealing dataset characteristics. They also developed a custom CNN model with a commendable accuracy score of 98.3%. These findings offer valuable insights into sign language recognition. The strength of the simplified CNN model lies in its effective recognition of BISINDO hand gestures, showing robustness across different conditions. However, its scalability and performance in complex environments like varied backgrounds remain unclear, affecting its real-world applicability. Additionally, concerns persist regarding its computational efficiency and speed for real-time implementations.

In their seminal study [18], researchers have introduced a novel approach using Convolutional Neural Networks (CNNs) to advance American Sign Language (ASL) recognition. The objective was to accurately recognize a diverse set of 24 ASL alphabet signs collected from real-life scenarios, showcasing the model's robustness. The CNN architecture, designed for intricate spatial feature capture, includes convolutional layers for pattern detection, max-pooling for information preservation, and a fully connected layer for feature fusion. Experimental results demonstrated an impressive 99.3% accuracy, highlighting the approach's efficacy. Strengths

include efficient Deep Learning utilization, achieving remarkable accuracy, and fast training due to small elapsed time and low loss error. However, a potential weakness lies in its reliance on static images, limiting applicability to dynamic scenarios. Further exploration into dynamic gesture recognition methods could enhance versatility and robustness.

In the realm of empirical investigations using a curated dataset [19], a novel convolutional neural network (CNN) architecture was devised, incorporating two convolutional layers, max-pooling, dropout, and densely connected layers, totaling 4,073,540 parameters. This approach yields remarkable accuracy scores of 99.72% for color data and 99.9% for grayscale data. Strengths include comprehensive methodology and robust performance, with high accuracies across various conditions and datasets. However, scalability and adaptability to real-time processing, especially with dynamic signs and video-based datasets, remain weaknesses.

After an in-depth exploration of related research in the field, this study aims to contribute new insights through a comparative analysis of three prominent image classification models: ResNeXt101 [20], VGG19 [21], and ViT [22] which have a good track record in image classification. The approach goes beyond traditional evaluation by incorporating Bayesian optimization [23] for hyperparameter tuning where no previous related research has explicitly described the hyperparameter process. This advanced optimization technique is used to systematically fine-tune the configuration of each model. The expected outcome is a better understanding of how these models work in the specific context of this study, shedding light on the impact of hyperparameter tuning on their overall effectiveness. By conducting this comparative study, the authors aim to see real differences in performance, offering valuable insights that can inform future developments in the field of image classification and model optimization.

3. PROPOSED METHOD

3.1 Dataset

The research at hand employs the NUS hand posture dataset II, as described in reference [24]. This dataset is a comprehensive collection of hand posture images that have been categorized into ten distinct classes. In total, it comprises 2000 unique data samples, each manifesting as hand posture images set against a variety of backgrounds. Specifically, there are 750 image samples featuring individuals as background, while an additional 1250 images encompass background-only data as shown in Table 1.

Table 1. NUS hand posture dataset

Subset Data	Amount of Data
A	2000
B	750
C	1250

The images within this dataset were meticulously captured in the vicinity of the National University of Singapore (NUS). They exhibit a remarkable diversity in terms of background complexity, encompassing both indoor and outdoor environments. Furthermore, the hand postures themselves exhibit notable variations, spanning different hand shapes, skin tones, and sizes. This variability is a testament to the dataset's richness, making it a valuable resource for research purposes.

One remarkable aspect of the NUS hand posture dataset II lies in its diversity of participants. It consists of 40 individuals hailing from various ethnic backgrounds. This group includes both men and women, ranging in age from 22 to 56 years. Each participant was carefully guided to display ten distinct hand positions, and each of these poses was recorded five times. The resulting image data has been standardized to 160x120 pixel dimensions, making it suitable for consistent and compatible image analysis methods. To provide a visual glimpse into the dataset's contents, an illustrative example of a hand posture image from this dataset can be found in Figure 2 and Figure 3.

Figure 2. Example hand posture images from dataset a NUS hand posture dataset-II

Figure 3. Examples of various class 9 hand posture images from dataset a NUS hand posture dataset-II

3.2 Models implementation

3.2.1 ResNeXt101

ResNeXt101, a state-of-the-art convolutional neural network (CNN) architecture, is renowned for its exceptional ability to capture complex spatial features. Rooted in the ResNeXt framework, it introduces the ResNeXt Block, which employs a novel "split-transform-merge" strategy, reminiscent of the Inception module. This strategic approach allows the aggregation of multiple transformations within a single module, setting it apart from its predecessors. The introduction of a new dimension known as "cardinality" plays a pivotal role within the ResNeXt architecture, alongside traditional dimensions of depth and width. The cardinality factor determines how transformations are organized and combined, significantly enhancing the network's capacity to capture intricate patterns and features. An overview of the resnext model architecture can be seen in Figure 4.

ResNeXt has excelled in various computer vision tasks, outperforming other network architectures like ResNet and Inception on the ImageNet dataset. A 101-layer ResNeXt, for instance, achieves superior accuracy to ResNet-200 while maintaining only half the complexity. Additionally, it boasts a more streamlined design compared to Inception models. Notably, ResNeXt served as the foundation for a second-place finish in the ILSVRC 2016 classification task [20].

ResNeXt101 facilitates real-time identification of dynamic hand movements from video feeds, employing a hierarchical structure for efficient online operation of offline-trained CNN architectures. It consists of a lightweight detector and a deep

classifier, utilizing the Levenshtein distance metric to evaluate single-time activations of detected gestures. ResNeXt-101 achieves state-of-the-art offline classification accuracy on benchmarks like EgoGesture and NVIDIA, maintaining near-offline performance in real-time detection and classification. The architecture's codes and pre-trained models are publicly available for further exploration and utilization [25].

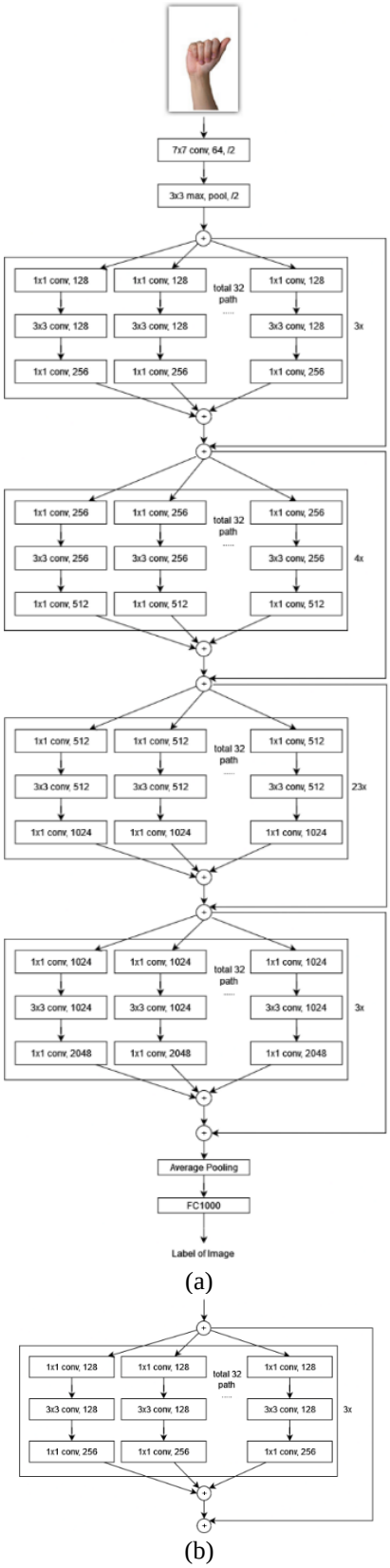

Figure 4. (a) ResNeXt-101 architecture (b) Block of ResNeXt with 32 cardinality

3.2.2 VGG19

VGG19, an innovative convolutional neural network (CNN) architecture, stood out as a significant breakthrough in the realm of computer vision, representing noteworthy progress in the field of image recognition. It was first presented in a seminal research paper published in 2014 [21], and it ranks among the remarkable achievements of its creators, setting a standard for future research undertakings.

At its core, VGG19 comprises a deep architecture, encompassing a total of 19 layers, with a specific arrangement of 16 convolutional layers and 3 fully connected layers as shown in Figure 5. A distinguishing characteristic of this model is its consistent utilization of 3×3 convolutional filters throughout its convolutional layers, coupled with a stride of 1 pixel. Additionally, VGG19 employs 2×2 max-pooling layers with a stride of 2 pixels. Notably, the authors investigation revealed that the application of these smaller convolution filters (3×3) was more efficacious in feature extraction and information representation than larger filters with a depth of 5×5 or 7×7.

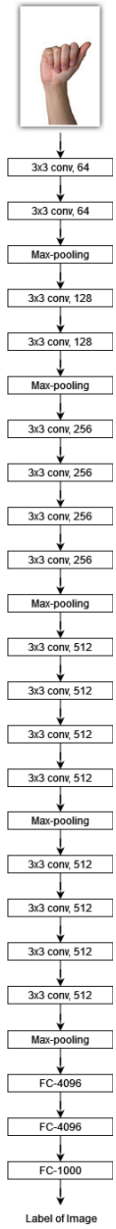

Figure 5. VGG19 architecture

VGG19's remarkable performance and influential role in the realm of deep learning are substantiated by its training on the extensive ImageNet dataset, boasting a vast repository of over one million images encompassing a thousand diverse classes. The outcome of this training was nothing short of exceptional, as the authors managed to attain state-of-the-art results in ImageNet's classification and localization tasks. Notably, VGG19 achieved a top-5 error rate of 7.3%, showcasing its competence in top-tier image classification. Furthermore, its localization error rate of 25.3% underscores its proficiency in accurately pinpointing objects within the images, a critical capability in practical applications like object detection and recognition.

3.2.3 Vision transformer

The vision transformer, or ViT, represents a paradigm shift in computer vision by leveraging the success of transformer architectures in natural language processing tasks. Traditionally, convolutional neural networks (CNNs) have been the backbone of image classification models, but ViT introduces a novel approach by treating an image as a sequence of patches, much like words in a sentence. This departure from grid-based processing enables ViT to capture long-range dependencies and relationships within an image.

At the heart of the vision transformer is the transformer architecture, originally designed for sequence-to-sequence tasks in natural language processing. The transformer's self-attention mechanism allows it to focus on different parts of the input sequence, making it highly effective for capturing global contextual information. In the context of ViT, this attention mechanism is applied to image patches, allowing the model to attend to different spatial regions simultaneously.

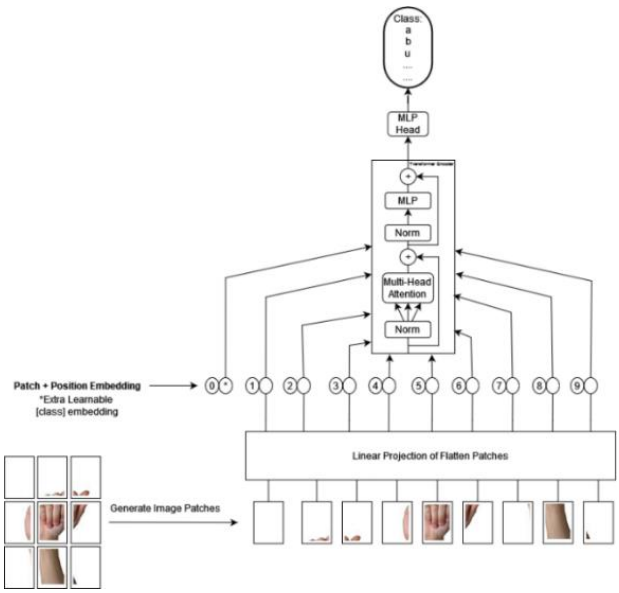

Figure 6. Vision transformer architecture

In Figure 6, The ViT architecture consists of an initial embedding layer that linearly projects the flattened image Later, as the training progresses, these earlier layers can be gradually unfrozen to allow for more adaptation to the target dataset. During the fine-tuning process, all layers are trained using the same learning rate. This decision is based on the hyperparameter tuning results obtained using Bayesian optimization.

3.3 Experimental setup

In this study, the main objective is to assess the performance of three pre-trained models, namely ResNeXt101, VGG19, and ViT, obtained from PyTorch's ImageNet dataset. These models will be fine-tuned using Stochastic Gradient Descent as the optimizer with the momentum value set to 0.9. In the fine-tuning process, only the final layers (fully connected layers or classifier layers) are modified to suit the target task. The earlier layers, responsible for learning general features, are often retained without modification, as they have already learned valuable features from the ImageNet dataset, in practice to freeze the weights of the earlier layers during the initial training epochs to prevent them from being disrupted by the random initialization of the final layers.

Later, as the training progresses, these earlier layers can be gradually unfrozen to allow for more adaptation to the target dataset. During the fine-tuning process, all layers are trained using the same learning rate. This decision is based on the hyperparameter tuning results obtained using Bayesian optimization.

To ensure a balanced and comprehensive evaluation, the dataset will be divided into A and B subsets, and in addition, a combined Subset A and B will be created. To maintain a balanced distribution strategy, the authors will perform stratification. Each of these subsets will undergo further division into training, validation, and testing sets, with a composition of 60% allocated to training, 20% to validation, and 20% to testing, as shown in Table 2. This careful dataset separation approach is implemented to facilitate a thorough and robust model evaluation process. The authors will then conduct hyperparameter tuning using Bayesian optimization on two critical hyperparameters: learning rate (ranging from 1e-5 to 1e-1), and batch size (choices of 16, 32, 64, and 128). This process will help us determine the optimal combination of hyperparameters for each model, leading to better performance.

Table 2. Composition of the separation of train, validation, and test datasets

Dataset	Amount of Image	Train	Validation	Testing
Subset A	2000	1200	400	400
Subset B	750	450	150	150
Subset A + B	2750	1650	550	550

To ensure thorough training and avoid overfitting and underfitting, the authors will monitor the model's training process and validation metrics at intervals of epochs, specifically at epoch multiples of 50. This approach ensures that the training process is observed at key checkpoints, providing insight into the model's behavior and performance. However, the authors will implement early stopping with a patience of 10 and a minimum delta of 0.001. This entails halting the training procedure if, for 10 consecutive epochs, there is no discernible decrease in the loss function beyond 0.001, where the loss function used in this research is cross entropy loss. This strategy allows for termination of training when the model's performance on the validation set ceases to improve significantly, indicating potential overfitting, avoid potential underfitting, and reaching convergence.

To conduct the experiments, the authors use 2*T4 GPUs available on Kaggle for training the model. Utilizing these resources will help the authors accelerate the training process and handle the computational demands efficiently.

3.4 Performance metrics

In evaluating hand posture recognition models, the focus lies on utilizing two key performance metrics: accuracy and the F1 score, crucial for assessing the efficacy of these models in recognizing various sign language gestures. Accuracy measures the proportion of correctly classified instances within the dataset, indicating the model's proficiency in distinguishing hand postures accurately. Additionally, the authors analyze training and validation accuracy and loss to understand the model's learning dynamics. High training accuracy and low loss demonstrate adeptness at fitting the data, while validation metrics offer insights into generalization and prevent overfitting. These metrics inform decisions on training, fine-tuning, and generalization, ensuring a robust hand posture recognition system.

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (1)$$

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (2)$$

4. RESULT AND DISCUSSION

The experiments in this research will employ parameters outlined in Table 3, which serves as a reference for the baseline model settings. These baseline parameters are shared between ResNeXt101, VGG19, and Vision Transformer models. For optimization, Stochastic Gradient Descent (SGD) will be used, with a common momentum value of 0.9. Additionally, the table provides insights into the total number of parameters for each model, which are essential in understanding the model complexity.

Table 3. Baseline model parameters

Parameters	ResNeXt101	VGG19	ViT
Optimizer	SGD	SGD	SGD
Momentum	0.9	0.9	0.9
Total Params	20,101,194	42,149,194	87,423,754

Specifically, ResNeXt101 contains 20,101,194 parameters, while VGG19 and Vision Transformer are characterized by a more extensive architecture, with a total of 42,149,194 parameters, and 87,423,754 parameters. These baseline settings serve as a starting point for the forthcoming experiments, allowing for a comparative evaluation of model performance and efficiency against the results obtained using other hyperparameter configurations.

In conjunction with these baseline settings, early stopping will be implemented with a criterion of patience set to 10 epochs and a minimum delta of 0.001. This approach aims to enhance training efficiency by halting the process if, for 10 consecutive epochs, there is no significant improvement in the loss function beyond the specified threshold. Such a strategy not only optimizes computational resources but also potentially prevents overfitting, thus positively influencing the generalization performance of the models. Consequently, integrating early stopping into the training process is anticipated to contribute to a more robust evaluation of model performance and efficiency, augmenting the findings derived from various hyperparameter configurations, bears the potential to influence the overall training dynamics and subsequently impact the results, warranting careful consideration and analysis throughout the experimentation phase.

The choice of the loss function plays a pivotal role in guiding the optimization process and ultimately determining the model's performance. The selected loss function for the task at hand is cross-entropy loss, which is commonly employed in classification tasks and particularly suited for scenarios where the model predicts probabilities across multiple classes. The appropriateness of cross-entropy loss stems from its ability to confidently penalize incorrect predictions more severely, thereby encouraging the model to learn more discriminative features and make accurate classifications.

4.1 Experiment on dataset subset A

Table 4 displays the key parameters obtained through a rigorous process of hyperparameter tuning. This tuning was carried out for three distinct models, namely ResNeXt101, VGG19 and ViT, which are used in the subsequent experiments. These parameters were determined using Bayesian optimization and are crucial in defining the performance of the models. In particular, the learning rates for ResNeXt101, VGG19, and ViT were found to be 0.0001987, 6.593e-05, and 0.0001308, respectively. The batch sizes chosen for the three models were 16 for ResNeXt101, 128 for VGG19, and 16 for ViT, while all models were trained for 200 epochs. It's worth noting that early stopping criteria were applied, resulting in the conclusion of training at the 143rd epoch for ResNeXt101, the 151st epoch for VGG19, and the 83rd epochs for ViT.

Table 4. Parameters obtained from hyperparameter tuning

Parameters	ResNeXt101	VGG19	ViT
Learning Rate	0.0001987	6.593e-05	0.0001308
Batch Size	16	128	16
Epochs (early stop)	200 (143)	200 (151)	200 (83)

Table 5. Training and validation results on accuracy and loss

Model	ResNeXt101	VGG19	ViT
Train Loss	0.0010	1.4631	0.0012
Validation Loss	0.0860	1.4692	0.0552
Train Accuracy	100.00	99.92	100.00
Validation Accuracy	98.00	96.00	98.50

The first experimental phase utilized the optimized parameters from Table 4 to conduct experiments on a subset of the data A, assessing the performance of pre-trained models ResNeXt101, VGG19, and ViT. Notably, all models showed significant improvements in accuracy and reductions in loss. Details of the loss and accuracy values for both training and validation phases are provided in Table 5. In summary, ResNeXt101 and ViT achieved remarkably low training losses of 0.0010 and 0.0012, with 100.00% training accuracy. VGG19 exhibited a slightly higher training loss of 1.4631 but maintained a commendable training accuracy of 99.92%. During validation, ViT achieved a loss of 0.0552 and an accuracy of 98.50%, showcasing its effectiveness in

generalization. ResNeXt101 and VGG19 had validation losses of 0.0860 and 1.4692, with validation accuracies of 98.00% and 96.00%, respectively, demonstrating robust performance. These results underscore the effectiveness of the chosen hyperparameters and the capabilities of the three models in both training and validation phases.

Table 6 presents the experimental results for accuracy and F1-score for three prominent models, ResNeXt101, VGG19, and ViT. These metrics are essential for evaluating the models' overall performance. ResNeXt101 achieved a remarkable accuracy of 99.03%, indicating its high precision in correctly classifying data points. Additionally, it recorded an impressive F1-score of 0.99, underscoring its effectiveness in both precision and recall. ViT and VGG19, while slightly lower in accuracy at 99.01% and 97.01%, still demonstrated strong classification capabilities. It achieved an F1-score of 0.99 and 0.97, suggesting its proficiency in achieving a balance between precision and recall. These results illustrate the excellent performance of all models, with both model ResNeXt101 and ViT exhibiting a slight edge in accuracy and F1-score, making it a potentially preferable choice for tasks that prioritize precision and recall in classification.

Table 6. Testing result on dataset subset A

Model	Accuracy	F1
ResNeXt101	99.03	0.99
VGG19	97.01	0.97
ViT	99.01	0.99

4.2 Experiment on dataset subset B

In Table 7, the authors present the parameters obtained through a meticulous hyperparameter tuning process for three distinct models, ResNeXt101, VGG19. and ViT. These parameters are critical in shaping the training and performance of the models. For all models, the author determined the optimal learning rate, batch size, and the number of training epochs. The learning rate for ResNeXt101 was set at 0.0018, while VGG19 and ViT used a slightly higher value of 0.0022 and 0.0112. All models utilized a batch size of 16 during training, and the author conducted training for a total of 200 epochs, implementing an early stopping mechanism at 32 epochs for ResNeXt101, 42 epochs for VGG19 and 34 epochs for ViT.

Table 7. Parameters obtained from hyperparameter tuning

Parameters	ResNeXt101	VGG19	ViT
Learning Rate	0.0018	0.0022	0.0112
Batch Size	16	16	16
Epochs (early stop)	200 (32)	200 (42)	200 (34)

In the second experiment, the authors focused on evaluating a subset of data labeled as 'B' using hyperparameter settings derived from Bayesian optimization results, as outlined in Table 7. Employing pretrained ResNeXt101, VGG19, and ViT models, significant improvements in accuracy and loss reduction were observed. Table 8 provides specific numerical values for training and validation loss and accuracy, offering a comprehensive overview of model performance. ViT and ResNeXt101 exhibit remarkably low training loss of 0.0001 and 0.0206, respectively, highlighting their capacity to fit the training data well. Similarly, VGG19 achieves a respectable training loss of 1.4612. Regarding validation loss, ResNeXt101 achieves a value of 0.0890, while VGG19 and ViT record validation losses of 1.5081 and 0.4611, respectively. These results underscore the robust performance of the models in generalizing to unseen data, further validating the efficacy of the chosen hyperparameter settings and the overall success of the experiment.

Table 8. Training and validation results on accuracy and loss

Model	ResNeXt101	VGG19	ViT
Train Loss	0.0206	1.4612	0.0001
Validation Loss	0.0890	1.5081	0.4611
Train Accuracy	100.00	100.00	100.00
Validation Accuracy	97.33	96.00	89.17

Table 9. Testing result on dataset subset B

Model	Accuracy	F1
ResNeXt101	95.31	0.95
VGG19	93.97	0.93
ViT	89.97	0.89

Table 9 presents the experimental results for accuracy and F1-score, focusing on three prominent models, ResNeXt101, VGG19, and ViT. In these experiments, ResNeXt101 achieved a commendable accuracy of 95.31%, showcasing its ability to correctly classify data instances. Additionally, it obtained an F1-score of 0.95, indicating its proficiency in striking a balance between precision and recall, which is crucial in classification tasks. VGG19 and ViT, while slightly lower in accuracy at 93.97% and 89.97%, still demonstrated strong performance in data classification. It obtained an F1-score of 0.93 and 0.89, highlighting its reliability in handling classification tasks with a favorable trade-off between precision and recall. These results underscore the effectiveness of all models ResNeXt101, VGG19, and ViT in the context of accuracy and F1-score, offering valuable insights into their capabilities for specific tasks.

4.3 Experiment on combination dataset subset A and B

Table 10 provides an overview of the hyperparameters obtained through a thorough tuning process for the third experiment. The three models used, ResNeXt101, VGG19, and ViT, have different parameter settings. For ResNeXt101, the learning rate was set at 0.0159, while a batch size of 128 was utilized, and training continued for 200 epochs, with early stopping occurring at the 22nd epoch. Meanwhile, VGG19 and ViT were configured with a learning rate of 0.0043 and 0.008679, with a batch size of 32 and 16, and it also trained for 200 epochs, but early stopping took place at the 19th epoch and 30th epochs.

Table 10. Parameters obtained from hyperparameter tuning

Parameters	ResNeXt101	VGG19	ViT
Learning Rate	0.0159	0.0043	0.008679
Batch Size	128	32	16
Epochs (early stop)	200 (22)	200 (19)	200 (30)

In the third experiment, a combination of data from subsets A and B was tested using the hyperparameters obtained from

Bayesian optimization, as detailed in Table 10. All models demonstrated impressive performance improvements, exhibiting increased accuracy and reduced loss within each epoch. Table 11 presents a comprehensive view of the experiment's results, showcasing the training and validation metrics. Both the ResNeXt101 and ViT models achieved remarkable 100% accuracy in training, with a negligible training loss of 0.0000. Moreover, they maintained strong accuracies of 98.36% and 97.82% on the validation set, with low validation losses of 0.021 and 0.0559, respectively. Conversely, the VGG19 model also achieved high training accuracy of 100%, albeit with a slightly higher training loss of 1.4612. In the validation phase, VGG19 demonstrated an accuracy of 96.73% with a validation loss of 1.4910. These results in Table 11 highlight the exceptional performance of both models, emphasizing their suitability for the task at hand.

Table 11. Training and validation results on accuracy and loss

Model	ResNeXt101	VGG19	ViT
Train Loss	0.0000	1.4612	0.0000
Validation Loss	0.0210	1.4910	0.0559
Train Accuracy	100.00	100.00	100.00
Validation Accuracy	98.36	96.73	97.82

In Table 12, the authors present the experimental results of accuracy and F1-score for three well-established models, ResNeXt101, VGG19, and ViT. These results provide valuable insights into the performance of these models in the context of the task at hand. The ResNeXt101 model achieved an impressive accuracy of 99.02% and an F1-score of 0.99, indicating its remarkable capability to correctly classify and provide a balanced trade-off between precision and recall. Similarly, the VGG19 and ViT models displayed a high level of accuracy, registering at 99.01% and 988.42, and while its F1-score was slightly lower at 0.97amd 0.98, it still showcased strong overall performance. These results underscore the effectiveness of all models ResNeXt101, VGG19, and ViT in accurately and reliably classifying the data, making them valuable choices for this task.

Table 12. Testing result on combination dataset subset A and B

Model	Accuracy	F1
ResNeXt101	99.02	0.99
VGG19	99.01	0.97
ViT	98.42	0.98

In comparing sign language recognition models ResNeXt101, VGG19, and ViT to the previous HOG-LBP + SVM approach on the NUS Hand Posture Dataset-II, notable advancements and limitations arise. In Subset A, VGG19 falls short while ResNeXt101 and ViT succeed. In Subset B, only ResNeXt101 succeeds, with VGG19 and ViT showing shortcomings. Combined experiments yield high accuracy, indicating overall success. ResNeXt101 consistently outperforms other models, though its computational complexity and resource-intensive training are notable. ViT may struggle with fine-grained spatial information, and VGG19's limitations underscore the need for further research on model scalability and effectiveness in sign language recognition tasks.

5. CONCLUSIONS AND FUTURE WORKS

In conclusion, the research presented valuable insights into the efficacy of deep learning models, particularly ResNeXt101, VGG19, and ViT, in hand posture recognition tasks using the NUS hand posture dataset II. Through a systematic experimental approach, the authors meticulously evaluated the performance of these models across different dataset subsets, employing rigorous hyperparameter tuning, including Bayesian optimization, and performance metrics assessment. The key findings underscored the consistent superiority of ResNeXt101, which consistently outperformed VGG19 and ViT in terms of accuracy and generalization across all experiments. However, it is crucial for the authors to concisely summarize these key findings to reinforce the main takeaways of the research, emphasizing ResNeXt101's remarkable performance while acknowledging the nuances of its superiority, including the role of Bayesian optimization in fine-tuning model parameters. Specifically, ResNeXt101's effectiveness in capturing complex spatial features and achieving high accuracy in hand posture recognition tasks was evident, but it is essential to clarify the context in which it outperforms other models, while also acknowledging potential limitations such as computational complexity and resource-intensive training requirements.

Future research should address current study limitations and explore alternative architectures or optimization techniques to improve deep learning model scalability and effectiveness in sign language recognition. Investigating model interpretability and integrating domain-specific knowledge can advance the field and enable real-world applications. Utilizing a more challenging dataset will push current models' boundaries, fostering a deeper understanding of their performance in complex scenarios and serving as a benchmark for evaluating their efficacy in challenging contexts. Engaging with demanding datasets can enhance image classification models' applicability in diverse scenarios.

REFERENCES

[1] Luhmann, N. (1992). What is communication?. Communication Theory, 2(3): 251-259. https://doi.org/10.1111/j.1468-2885.1992.tb00042.x

[2] What is the best definition of communication?. https://www.linkedin.com/pulse/what-best-definition-communication-milad-azami.

[3] Stokoe, W.C. (1980). Sign language structure. Annual Review of Anthropology, 9: 365-390. https://doi.org/10.1146/annurev.an.09.100180.002053

[4] Christopher, H.S. (2007). Military communications: From ancient times to the 21st century. ABC-CLIO: http://publisher.abc-clio.com/9781851097371

[5] Suharjito, Anderson, R., Wiryana, F., Ariesta, M.C., Kusuma, G.P. (2017). Sign language recognition application systems for deaf-mute people: A review based on input-process-output. Procedia Computer Science, 116: 441-448. https://doi.org/10.1016/j.procs.2017.10.028

[6] Suharjito, Wiryana, F, Kusuma, G.P., Zahra, A. (2018). Feature extraction methods in sign language recognition system: A literature review. In 2018 Indonesian Association for Pattern Recognition International

Conference (INAPR), pp. 11-15. IEEE. https://doi.org/10.1109/INAPR.2018.8626857
[7] Suharjito, S., Ariesta, M.C., Wiryana, F., Negara, I.G.P.K. (2018). A survey of hand gesture recognition methods in sign language recognition. Pertanika Journal of Science & Technology, 26(4): 1659-1675.
[8] Dwijayanti, S., Hermawati, Taqiyyah, S.I., Hikmarika, H., Suprapto, B.Y. (2021). Indonesia sign language recognition using convolutional neural network. International Journal of Advanced Computer Science and Applications, 12(10). https://doi.org/10.14569/IJACSA.2021.0121046
[9] Ismail, M.H., Dawwd, S.A., Ali, F.H. (2021). Static hand gesture recognition of Arabic sign language by using deep CNNs. Indonesian Journal of Electrical Engineering and Computer Science, 24(1): 178-188. https://doi.org/10.11591/ijeecs.v24.i1.pp178-188.
[10] Adithya, V., Rajesh, R. (2020). A deep convolutional neural network approach for static hand gesture recognition. Procedia Computer Science, 171: 2353-2361. https://doi.org/10.1016/j.procs.2020.04.255
[11] Aljabar, A., Suharjito, S. (2020). BISINDO (Bahasa Isyarat Indonesia) sign language recognition using CNN and LSTM. Advances in Science, Technology and Engineering Systems Journal, 5(5): 282-287. https://doi.org/10.25046/aj050535
[12] Zhang, F., Liu, Y., Zou, C.Y., Wang, Y.T. (2018). Hand gesture recognition based on HOG-LBP feature. In 2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Houston, TX, USA, pp.1-6. https://doi.org/10.1109/I2MTC.2018.8409816
[13] Chavan, S., Yu, X.R., Saniie, J. (2021). Convolutional neural network hand gesture recognition for American sign language. In 2021 IEEE International Conference on Electro Information Technology (EIT), Mt. Pleasant, MI, USA, pp. 188-192. https://doi.org/10.1109/EIT51626.2021.9491897
[14] Das, P., Ahmed, T., Ali, M.F. (2020). Static hand gesture recognition for American sign language using deep convolutional neural network. In 2020 IEEE Region 10 Symposium (TENSYMP). pp. 1762-1765. https://doi.org/10.1109/TENSYMP50017.2020.9230772
[15] Bhaumik, G., Verma, M., Govil, M.C., Vipparthi, S.K. (2020). CrossFeat: Multi-scale cross feature aggregation network for hand gesture recognition. In 2020 IEEE 15th International Conference on Industrial and Information Systems (ICIIS), RUPNAGAR, India, pp. 274-279. IEEE. https://doi.org/10.1109/ICIIS51140.2020.9342652
[16] Basri, S.E., Indra, D., Darwis, H., Mufila, A.W., Ilmawan, L.B., Purwanto, B. (2021). Recognition of Indonesian sign language alphabets using Fourier descriptor method. In 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT), Surabaya, Indonesia, pp. 405-409. https://doi.org/10.1109/EIConCIT50028.2021.9431883
[17] Thiracitta, N., Gunawan, H., Suharjito (2021). SIBI sign language recognition using convolutional neural network combined with transfer learning and non-trainable parameters. Procedia Computer Science, 179: 72-80. https://doi.org/10.1016/j.procs.2020.12.011
[18] Abdulhussein, A.A., Raheem, F.A. (2020). Hand gesture recognition of static letters American sign language (ASL) using deep learning. Engineering and Technology Journal, 38(6A): 926-937. https://doi.org/10.30684/etj.v38i6A.533
[19] Wadhawan, A., Kumar, P. (2020). Deep learning-based sign language recognition system for static signs. Neural Computing and Applications, 32(12): 7957-7968. https://link.springer.com/article/10.1007/s00521-019-04691-y.
[20] Xie, S.N., Girshick, R., Dollár, P., Tu, Z.W., He, K.M. (2017). Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5987-5995. https://doi.org/10.1109/CVPR.2017.634
[21] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556. https://doi.org/10.48550/arXiv.1409.1556
[22] Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X.H., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. https://doi.org/10.48550/arXiv.2010.11929
[23] Elshewey, A.M., Shams, M.Y., El-Rashidy, N., Elhady, A.M., Shohieb, S.M., Tarek, Z. (2023). Bayesian Optimization with support vector machine model for Parkinson disease classification. Sensors (Basel), 23(4): 2085. https://doi.org/10.3390/s23042085
[24] Pisharady, P.K., Vadakkepat, P., Loh, A.P. (2013). Attention based detection and recognition of hand postures against complex backgrounds. International Journal of Computer Vision, 101: 403-419. https://doi.org/10.1007/s11263-012-0560-5
[25] Köpüklü, O., Gunduz, A., Kose, N., Rigoll, G. (2019). Real-time hand gesture detection and classification using convolutional neural networks. In 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), Lille, France, pp. 1-8. https://doi.org/10.1109/FG.2019.8756576