mathematics

MDPI

Article

A Hybrid Image Augmentation Technique for User- and Environment-Independent Hand Gesture Recognition Based on Deep Learning

Baiti-Ahmad Awaluddin [1,2], Chun-Tang Chao [1] and Juing-Shian Chiou [1,*]

[1] Department of Electrical Engineering, Southern Taiwan University of Science and Technology, 1, Nan-Tai St., Yongkang District, Tainan City 71005, Taiwan; da82b207@stust.edu.tw (B.-A.A.); tang@stust.edu.tw (C.-T.C.)
[2] Department of Electronics Engineering Education, Universitas Negeri Yogyakarta, Yogyakarta 55281, Indonesia
* Correspondence: jschiou@stust.edu.tw; Tel.: +886-916-221-152; Fax: +886-6-3010-069

Abstract: This research stems from the increasing use of hand gestures in various applications, such as sign language recognition to electronic device control. The focus is the importance of accuracy and robustness in recognizing hand gestures to avoid misinterpretation and instruction errors. However, many experiments on hand gesture recognition are conducted in limited laboratory environments, which do not fully reflect the everyday use of hand gestures. Therefore, the importance of an ideal background in hand gesture recognition, involving only the signer without any distracting background, is highlighted. In the real world, the use of hand gestures involves various unique environmental conditions, including differences in background colors, varying lighting conditions, and different hand gesture positions. However, the datasets available to train hand gesture recognition models often lack sufficient variability, thereby hindering the development of accurate and adaptable systems. This research aims to develop a robust hand gesture recognition model capable of operating effectively in diverse real-world environments. By leveraging deep learning-based image augmentation techniques, the study seeks to enhance the accuracy of hand gesture recognition by simulating various environmental conditions. Through data duplication and augmentation methods, including background, geometric, and lighting adjustments, the diversity of the primary dataset is expanded to improve the effectiveness of model training. It is important to note that the utilization of the green screen technique, combined with geometric and lighting augmentation, significantly contributes to the model's ability to recognize hand gestures accurately. The research results show a significant improvement in accuracy, especially with implementing the proposed green screen technique, underscoring its effectiveness in adapting to various environmental contexts. Additionally, the study emphasizes the importance of adjusting augmentation techniques to the dataset's characteristics for optimal performance. These findings provide valuable insights into the practical application of hand gesture recognition technology and pave the way for further research in tailoring techniques to datasets with varying complexities and environmental variations.

Keywords: hand gesture recognition; hybrid augmentation; environment independent; green screen technique

MSC: 68T07

Citation: Awaluddin, B.-A.; Chao, C.-T.; Chiou, J.-S. A Hybrid Image Augmentation Technique for User- and Environment-Independent Hand Gesture Recognition Based on Deep Learning. *Mathematics* **2024**, *12*, 1393. https://doi.org/10.3390/math12091393

Academic Editor: Konstantin Kozlov

Received: 8 March 2024
Revised: 13 April 2024
Accepted: 30 April 2024
Published: 2 May 2024

1. Introduction

Hand Gesture Recognition (HGR) plays an essential role in various interactive systems, including signaling systems that rely on gestures [1,2], recognition of sign language [3,4], sports-specific sign language recognition [5,6], human gesture recognition [7,8], pose and posture detection [9,10], physical exercise monitoring [11,12], and control of smart

home/assisted living applications [13]. Computer scientists have harnessed diverse computational techniques and methodologies to optimize human-computer interactions [14,15], integrating hand gestures into software programs that enhance computer and human communication [16]. The advancement of gesture recognition systems significantly enhances the interaction between computers and humans, with hand gestures becoming increasingly prevalent across various sectors. Hand gestures are now used in diverse applications such as gaming [17,18], virtual reality and augmented reality [19,20], assisted living [21,22], and more. Moreover, the recent proliferation of hand gesture recognition in industries like human-robot interaction in manufacturing [23,24] and autonomous vehicle control [25] has spurred considerable interest. Against the backdrop of the ongoing COVID-19 pandemic from 2020 to 2023 [26], where social distancing remains a top priority, the scope for implementing hand gestures is expanding [27], rendering it an intriguing topic for further exploration and discussion.

Although there are several techniques for hand gesture recognition, deep learning—a part of machine learning—has become the most advanced way. Deep learning research has made remarkable strides in solving complex image recognition and related challenges. The development of deep learning was greatly influenced by the significant use of (CNNs) for image classification in 2012. AlexNet performed better than traditional shallow approaches [28]. The popularity of deep learning in this field can be credited to advancements in deep network structures, significant processing capability, and the availability of extensive training datasets. ImageNet [29] is a prominent huge dataset that has significantly stimulated additional advancements. The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) was responsible for coordinating and monitoring the progress of many CNN models, leading to the creation of well-known architectures, including VGG [30], Google Net [31], and ResNet50 [32]. The classification top-5 error rate has significantly decreased throughout the years. 2010–2011, when shallow methods were often used, the error rate was over 25%. However, with the introduction of deep learning in 2015, the error rate dropped to less than 5% [33].

However, despite these significant achievements, deep neural networks and their associated learning algorithms confront several pertinent challenges. One of the most frequently cited issues is the lack of training data [34–36] or the imbalance of classes within datasets [31,32]. To resolve this matter, data augmentation has emerged as the favored technique to enlarge datasets and mitigate the problem of insufficient data for training models [37,38]. Data augmentation involves applying various transformations or manipulations to existing data to generate new samples that resemble the original data. This technique extends the existing dataset by generating new variations from the current data. Yet, despite the efficacy of data augmentation in alleviating data shortages, there are still constraints on the number of variations that can be generated from the original data. This limitation has spurred the exploration of methods to generate an almost limitless volume of new data for use in datasets. Various methods of augmenting image-based data have been developed [39], including geometric changes, color tweaks, random occlusions, and deep learning techniques such as Generative Adversarial Networks (GANs).

Notably, hand gestures in applications demand high robustness and accuracy to preclude misinterpretation and instruction errors. A review of experiments on hand gesture recognition conducted by Noraini et al. [40] revealed that over 47 articles had been executed in constrained laboratory settings. Lim et al. [41] attribute this to the ideal backdrop for gesture recognition, which includes only the signer, devoid of any background, as background clutter can impair gesture recognition accuracy. This limitation underscores the necessity for conducting tests beyond the confines of controlled environments. It is a pressing concern because the domain of hand gesture recognition extends far beyond the laboratory setting. Hand gestures now find applications in many real-world scenarios, each characterized by unique environmental conditions, including varying background colors, lighting conditions, and hand gesture positions [40]. These challenges are compounded by limitations in available datasets for real-world HGR, which often lack the variety

and diversity essential for training robust models [42]. The scarcity of comprehensive and diverse real-world datasets hinders the development of accurate and adaptable hand gesture recognition systems. Training GANs to produce new synthetic images is challenging due to mode collapse, non-convergence, and oscillatory behavior [42], despite its potential for data augmentation [42].

Data augmentation may also generate new data for testing classifiers in some cases [43]. In comparison to GAN-based augmentation, classical augmentation techniques, such as background and brightness variations and geometric transformations, are more straightforward, efficient, and effective for enhancing the performance of CNNs in image classification tasks. For instance, Dundar et al. [38] investigated the impact of background variations on image classification. Their study demonstrated that altering the backgrounds of the training dataset could significantly affect testing accuracies. The study enhanced existing augmentation techniques with foreground segmented objects, and its findings are instrumental in improving accuracies, mainly when working with limited datasets by creating synthetic images.

Kandel [44] adopted a meticulous approach to brightness augmentation techniques, which closely matched the original brightness levels of images. This approach yielded optimal performance. Previous studies on HGR have employed geometric transformations as a data augmentation technique. For example, in [45], the implementation of geometric augmentation significantly enhanced the performance of CNNs by a maximum of 4%. In another study, an HGR system that utilized Capsule Networks exhibited improved performance when combined with geometric augmentation involving rotation and translation operations [46]. Similarly, ref. [47] employed Adapted CNN and image translation (vertical and horizontal) to augment the original data, resulting in a 4% improvement in classification accuracy. Furthermore, ref. [48] utilized random scaling and horizontal/vertical translation to augment training data diversity for HGR applications.

The paper presented by Luo, Cui, and Li in 2021 [49] established a CNN model for recognition. Hand gesture detection is hindered by movements' intricate and varied nature, which are affected by contextual circumstances, lighting conditions, and occlusion. The initial step in improving recognition involves the application of a filter to the skin color of the hand inside the YCbCr color space, thereby isolating the region of motion. A Gaussian filter is employed to mitigate edge noise, followed by morphological gray opening operations and the watershed method for contour segmentation. The eight-connected filling algorithm improves the characteristics of motion. The model in this experiment identifies ten movements ranging from 0 to 9. The experimental findings illustrate the prompt and precise identification of the proposed approach, with an average success rate of 96.46%, without a substantial augmentation in the recognition duration.

However, Rahmat et al. [50] found difficulties when attempting to recognize hand movements in complex backgrounds and without objects using computer vision in human-computer interaction. Problems with skin and background detection needed a potent remedy. The suggested method comprised several stages: acquiring images, resizing them, converting the color space, utilizing the HS-CbCr format for skin recognition, and applying averaging to overcome background difficulties. Grayscale image processing, background accumulation, thresholding, inversion, frame differencing, and picture enhancement were further processes. Contour, convex-hull, and convexity flaws were used to extract features, which were then counted on fingers, and hand direction was determined. The generated instruction-controlled applications like PDF readers, video players, audio players, and slideshow presentations. The method's efficiency was proven by experimental findings, which showed up to 98.71% accuracy under well-lit situations. Lighting highly influenced accuracy, with 95.03% accuracy recorded under lesser illumination. Future improvements included considering machine learning for better object detection accuracy and integrating a hand-tracking approach for dynamic gesture recognition. It was suggested that future studies in increasing hand gesture recognition address skin detection issues related to lighting.

Yi Yao and Chang-Tsun Li's research [51] focuses on addressing the formidable task of recognizing and tracking hand movements in uncontrolled environments. They identify several critical challenges inherent in such environments, including multiple hand regions, moving background objects, variations in scale, speed, trajectory location, changing lighting conditions, and frontal occlusions. They propose an appearance-based method that leverages a novel classifier weighting scheme to tackle these challenges. Their method demonstrates promising performance in effectively handling uncontrolled environments' complexities without prior knowledge. They utilize the Palm Graffiti Digits Database and the Warwick Hand Gesture Database to evaluate their approach. Through extensive experimentation, they illustrate their method's capability to overcome environmental obstacles and enhance the performance of the initial classifier. Our research methodology presents significant advancements over previous works, such as those by Luo, Cui, and Li (2021) and Yi Yao and Rahmat et al., by incorporating cutting-edge data augmentation techniques and optimizing neural network architectures specifically designed for hand gesture recognition. Unlike the former, which relies on traditional image pre-processing and CNN models, and the latter, which focuses on hand gesture recognition under well-lit conditions, our approach is engineered to robustly adapt to a wider range of environmental variations, including dramatic lighting changes and complex backgrounds. We employ advanced data augmentation strategies, including geometric manipulations and lighting variations, to significantly enhance model resilience against external condition fluctuations. Furthermore, our specialized neural network architecture is optimized to capture essential features of hand gestures more efficiently, aiming for superior accuracy while maintaining or even reducing recognition time. This innovative methodology broadens the spectrum of recognizable hand gestures under challenging conditions and increases the model's applicability and flexibility across diverse domains, setting a new benchmark for future hand gesture recognition research.

This innovative research study explores brightness's function as an augmentation method in training deep learning networks. This work aims to expand current knowledge and provide a more thorough understanding of color distortion approaches, geometric augmentation, noise relevance, and image quality in deep learning architectures, in addition to complementing previous research in these areas. Moreover, the paper explores the significance of augmenting background variation to enhance the performance and robustness of deep learning models. By considering the intricate interplay between foreground objects and their surrounding environments, the researchers aim to uncover the potential benefits and challenges of incorporating background variation as an augmentation strategy. In conjunction with all these strategies, another objective is to ascertain the possibility of creating virtually limitless datasets using classical augmentation techniques, resulting in more diversified datasets to address data insufficiency. In brief, the main contributions of this study are:

(a) Introduce a new model augmentation that combines geometric transformation, background, brightness, temperature, and blurriness difference to train deep learning networks to improve hand gesture recognition.
(b) Offering a strategy change background using green screening as a data augmentation technique eliminates the need for manual object annotations. It enhances training performance with limited data by replacing the background using computer vision algorithms.
(c) Depict the proposed green screen technique hand gesture dataset that can be used for training hand gesture recognition to know the effect of background distortion to simulate HGR in real-world uncontrolled environments.
(d) Exploring the potential of classical augmentation techniques to generate unlimited data variations and maintain accuracy.

This research evaluates the accuracy of gesture classification experiments using a primary dataset that employs the proposed green screen technique to replace the background with complex backgrounds, such as images from indoor and outdoor locations

and backgrounds experiencing color distortion. Additionally, classic augmentations like geometric transformations, brightness adjustments, temperature changes, and blurriness are applied to simulate an uncontrolled environment. The primary dataset is then tested with a secondary dataset of existing public datasets to assess its accuracy.

This article is structured as follows: Section 2 briefly introduces the theory of surrounding environmental conditions, which is the focus of the research, as well as the dataset used, and the various augmentation techniques applied to increase the training dataset and simulate real-world conditions. One of the augmentation techniques used is the proposed green screen technique to replace the background image with an appropriate one. Additionally, geometric augmentation is used to represent positions, brightness augmentation to mimic real-world lighting conditions and blur to improve image clarity. Furthermore, this section also explains the use of pre-trained models and CNN. Section 3 presents the experimental setup, programming tools, Data Preparation, and the result of the experiment. Section 4 analyzes the performance of various augmentation techniques on model accuracy. Additionally, the paper explores implications and potential future research opportunities in this field. Section 5 summarizes the key findings, underscoring the research significance and outlining potential future directions, including developing an image augmentation framework for static hand gesture recognition.

2. Materials and Methods

This chapter explains the basic concepts used in this article, including the environment surrounding and simulation using the augmentation technique, deep learning, and pre-trained, which we will use. We will construct the model approach to simulate hand gesture recognition in real environments.

2.1. Material

2.1.1. Environmental Surrounding

Recognizing hand gestures in uncontrolled environmental conditions involves several key factors. In this context, complex backgrounds are the primary factors affecting the accuracy of hand gesture recognition. Research conducted by Fuchang et al. [52]. Highlights that lighting, rotation, translation, and environmental scaling changes can challenge separating the hand from the background. They address this issue by using depth data to separate the hand and achieve synchronized color and depth images. However, their research should have explicitly explored the impact of optical blur on recognition accuracy, leaving a significant gap in our understanding.

It's important to note that another limitation of models utilizing depth images is the requirement for specialized cameras capable of capturing depth information. This limitation may restrict the practicality of depth-based models, as such cameras are often more expensive and less common than regular cameras.

In their study, Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich [53] provide insights into the specific influence of optical blur, resulting from defocus or subject and camera motion, on network model performance. Their research delves into the detrimental effects of blur. It emphasizes the importance of considering blurriness as a significant factor that can challenge the recognition of hand gestures in uncontrolled environmental conditions. Therefore, addressing the impact of optical blur, along with other environmental factors, is crucial for enhancing the accuracy of hand gesture recognition systems.

Additionally, hand gesture recognition in uncontrolled environments faces various challenges beyond complex backgrounds. One of these challenges is the variations in lighting conditions, as examined in a study by Salunke [54]. Their findings highlight that optimal accuracy is achieved under bright lighting conditions. To address the issues related to changes in lighting, researchers have employed techniques such as brightness augmentation. Temperature sensors, a form of temperature augmentation, have also been utilized to compensate for temperature fluctuations that can impact image quality, as demonstrated in the study by Oinam et al. [55] Temperature augmentation ensures that

machine learning models used for hand gesture recognition perform consistently across diverse lighting conditions.

In image processing, noise reduction and contrast enhancement under varying lighting conditions become crucial. Jose et al. [56] use digital image processing techniques to eliminate noise, separate the hand from the background, and enhance contrast. In cases of complex backgrounds with significant background variations, hand localization becomes a challenging task. The study by Peijun et al. [57] shows that accuracy varies depending on the complexity of the background.

Additionally, geometric transformations such as rotation and scaling are essential, and several studies, as explained by Yu et al. [58], have successfully addressed this issue. In situations where devices like Kinect detect a larger object, incorrect results may appear, as found in the study by Atharva et al. [59]. Environmental noise reduction needs to be considered in noisy background situations, as seen in the research by Hafız et al. [60]. Multi-modal approaches using depth data and color images enhance hand gesture recognition accuracy in various backgrounds and lighting situations, as shown in the study by Yifan et al. [61].

Furthermore, addressing geometric transformations is a primary focus. The study by Yiwen et al. [62] demonstrates that their method can adapt to geometric transformations. Time of Flight (ToF) data is used in several studies, as presented by Sachara et al. [63]. However, this data can be noisy depending on lighting conditions and other factors.

Technological approaches in the form of sensor systems have also been used to address various environmental constraints. Washef [64] successfully overcame the limitations of glove-based and vision-based approaches in situations with varying lighting challenges, complex backgrounds, and camera distance. Finally, in handling variations in rotation, scale, translation, and other transformations, the system in the study by Lalit & Pritee [65] has proven to handle various variations, including digits, letters, and symbols.

In conclusion, the representation of environmental surroundings in gesture recognition encompasses factors such as complex background, geometric transformation, brightness augmentation, temperature augmentation, and blurriness augmentation. Researchers have devised various strategies and techniques to address these challenges and enhance the accuracy of hand gesture recognition in uncontrolled environments. These insights can significantly contribute to developing robust and adaptable gesture recognition systems.

2.1.2. Simulate Environment as Real World

Based on the theory of environmental surroundings, we infer that the ecological environment representation in sign recognition encompasses factors such as complex backgrounds, geometric transformations, brightness enhancements, temperature increases, and blurriness levels. Furthermore, the approach to simulate these conditions is elaborated on.

Augmentation Technique

Augmentation is a technique used to address data scarcity in neural network training by applying specific transformations to the original data or images. It is precious in domains like the medical field, where data availability may be limited. These transformations generate new images, facilitating the generalization of knowledge learned by classifiers such as neural networks. Generalization is crucial as it enables models to perform well on unseen or new data, ensuring their reliability in real-world scenarios. This paper uses augmentation to construct a model approach to simulate conditions in real-world scenarios. This technique applies an alternative "on-the-fly" augmentation that dynamically augments during the training process, generating new data in each training epoch and improving knowledge generalization, as shown in Figure 1.

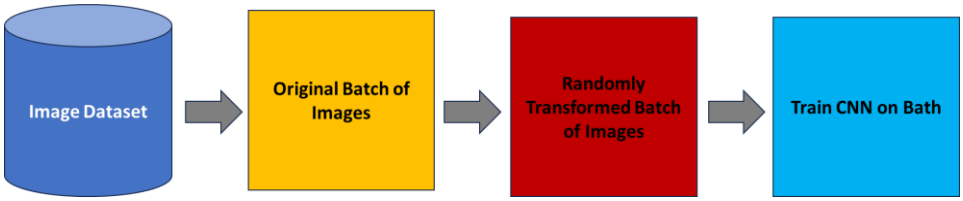

Figure 1. On-The-Fly Augmentation Approach.

In Figure 1, the original image dataset is divided into smaller batches. Random transformations are applied to each batch before feeding them into the machine learning or deep learning algorithm for training. This process generates distinct augmented images within each batch and epoch, promoting better knowledge generalization. Implementing on-the-fly augmentation in the hand gesture recognition system resulted in improved performance. This approach eliminates the need for separate storage of augmented images, making it more memory efficient. By effectively leveraging image augmentation techniques, models can overcome data scarcity, avoid overfitting, and achieve better generalization capabilities. Implementing on-the-fly augmentation in the hand gesture recognition system resulted in improved performance. This approach eliminates the need for separate storage of augmented images, making it more memory efficient. By effectively leveraging image augmentation techniques, models can overcome data scarcity, avoid overfitting, and achieve better generalization capabilities.

In this paper, the augmentation process to simulate conditions in the real world is illustrated in Figure 2. This figure describes the main steps in the image augmentation process in the hand gesture recognition system. The first stage involves receiving the original hand gesture image, which undergoes transformations and modifications, including background changes, geometric adjustments, and brightness and contrast alterations. This augmentation process aims to generate the necessary image variations for training the hand gesture recognition model. Various augmentation techniques enrich the original image with background variations, different perspectives, and changes in brightness. This augmentation process results in an inset of images used in the model training. This allows the model to become more robust and recognize various conditions and environments.

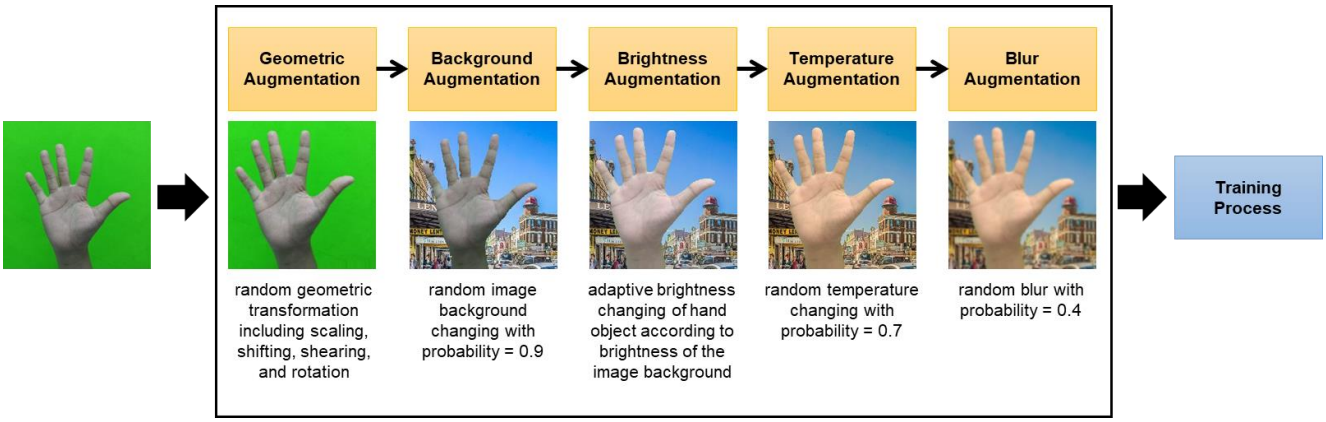

Figure 2. Process of Augmentation to Simulate Condition in the Real World.

Geometric Transformation

This work uses this method to simulate hand gesture poses in uncontrolled environments. Nevertheless, it is essential to acknowledge that certain adjustments, such as inversion, may not be appropriate for image categories, such as digit images, as it can lead to ambiguity between the digits 6 and 9. This research will employ a specific technique:

Image Scaling

Image scaling is a geometric transformation technique used to enlarge an image by multiplying it with distinct scaling factors on the (x) and (y) axes. This enables us to resize the image to a larger or smaller size as required while preserving its dimensions and intricate features. Equations (1) and (2) can be utilized for picture scaling. In these equations, (x,y) represents the coordinates of a pixel in the original image, (x',y') represent the coordinates of the pixel in the scaled image, and (s_x) and (s_y) represent the scale factors for the rows and columns of the image, respectively.

$$x' = x \cdot s_x \tag{1}$$

$$y' = y \cdot s_y \tag{2}$$

Interpolation can enhance the smoothness of the object's edges in the scaled image while preserving the aspect ratio when the scaling factors (s_x) and (s_y) are equal. Resizing images enhances the model's effectiveness and resilience against input size variations. When an image is enlarged, it is trimmed to its original dimensions; however, when it is reduced in size, the original dimensions are preserved, and the empty area is filled using the nearest pixel neighbor technique. Figure 3 illustrates the description of image scaling for augmentation.

Figure 3. Image Scaling Illustration.

Image Rotation

Image rotation is a commonly used technique for improving data in computer vision applications. In order to increase the amount of training data, it is necessary to rotate an image by a predefined angle, often between 0 and 360 degrees. Equations (3) and (4) are used to rotate an image. They take the original pixel coordinates (x, y) and calculate the corresponding pixel coordinates (x', y') in the rotated image. The rotation angle in radians is denoted by θ, and (cx, cy) represents the coordinates of the picture center.

$$x' = (x - cx)cos\theta - (y - cy)sin\theta + cx \tag{3}$$

$$y' = (x - cx)sin\theta + (y - cy)\cos\theta + cy \tag{4}$$

Like picture scaling, image rotation can create empty spaces that require filling using interpolation methods, such as the nearest pixel technique employed in this study. Figure 4 illustrates the description of image rotation for augmentation.

Figure 4. Image Rotation illustration.

Image Translation

Image translation refers to the movement of an image in both the horizontal (x-axis) and vertical (y-axis) directions by a defined number of pixels. This shift in position creates additional training data, improving the model's ability to handle changes in position. Equation (5) is employed for translating the x-axis, while Equation (6) is utilized for translating the y-axis. In these equations, (x,y) represents the coordinates of a pixel in the original image, (x',y') represents the coordinates of the corresponding pixel in the translated image, and (dx, dy) represents the translation offsets.

$$x' = x + dx \tag{5}$$

$$y' = y + dy \tag{6}$$

The nearest pixel interpolation algorithm additionally populates empty regions in the translated images. The depiction of the picture translation for augmentation may be seen in Figure 5.

Figure 5. Image Translation Illustration.

Image Shearing

Image sharing is a technique used to alter an image by shifting each row or column of pixels along the x-axis or y-axis. The amount of displacement is controlled by the y-coordinate or x-coordinate of each pixel. This methodology simplifies the administration of input photographs acquired from different views or angles. Equations (7) and (8) can be used to shear an image along the x-axis and y-axis. In these equations, (x,y) represent the coordinates of a pixel in the original image, (x',y') represent the coordinates of the corresponding pixel in the sheared image, and shx and shy represent the shear factors along the x-axis and y-axis, respectively.

$$x' = x + shx * y \tag{7}$$

$$y' = y + shy * x \tag{8}$$

To fill in the space in the sheared images, nearest pixel neighbor interpolation is used. Figure 6 shows an illustration of how image shearing is described for augmentation.

Figure 6. Image Shearing Illustration.

Image Flipping

Image flipping is a method that alters the image by inverting its left and right sides. HGR only employs horizontal flipping. Equation (9) is the formula for horizontal flipping.

In this equation, (x,y) represents the coordinates of a pixel in the original image, x' represents the matching pixel index in the flipped image, and W represents the image's width.

$$x' = (W - 1) - x \tag{9}$$

The order of pixels in each row is reversed from left to right to invert the entire image. Flipping an image horizontally does not need interpolation. Figure 7 shows how to describe Image Flipping Horizontally

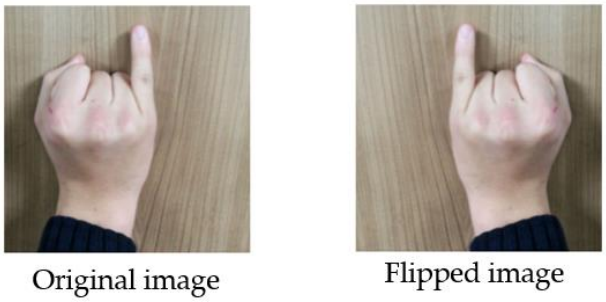

Figure 7. Image Flipping Horizontally Illustration.

Background Augmentation

This research project aims to develop a system capable of accurately recognizing hand movements, especially in various environmental conditions. We use a technique that replaces the proposed green screen technique background with various pre-prepared backgrounds to simulate such ecological variations. This creates a consistent and uniform background so the computer can focus more on correctly recognizing hand gestures.

The approach we used was the proposed green screen technique, also known as Chroma Key, which has become an essential tool in the field of visual effects. When creating hand gesture images, we use this technique as a backdrop to make it easy to replace the background to simulate a natural environment. As highlighted in the various abstracts, green screens have evolved where previously impossible backgrounds, such as alien planets and outer space, can now be easily realized.

Several studies further outline techniques and advances related to green screens. For example, Raditya et al. [66] researched using various background colors in the Chroma Key process to measure efficiency and fatigue. Jin Zhi et al. [67] explore alternatives to green, incorporating complex mathematical concepts to improve the quality of results. In addition, Soumyadip Sengupta et al. [68] introduce a technique to produce mattes without relying on a green screen or hand manipulation, making the background replacement process more convenient. By referring to these studies, we gain insight into the evolutionary nature of green screen techniques and their diverse applications. In the context of the hand-gesture images we create, green screen techniques offer creative freedom to build dynamic and immersive visual environments, bridging the gap between technology and visual arts to develop realistic environmental simulations.

Backgrounds used to replace the original background in hand gesture images include a variety of indoor and outdoor settings, from busy airport scenes and dimly lit basements to quiet library environments, cozy rooms, and streets. Busy city street. We also incorporate various backgrounds, such as forests, subtle color gradients, and vibrant environments. All these background variations enable computers to effectively learn and recognize hand movements in various real-world scenarios, improving their operational capabilities in diverse conditions. The following are examples of the images we use as background variations in Figure 8:

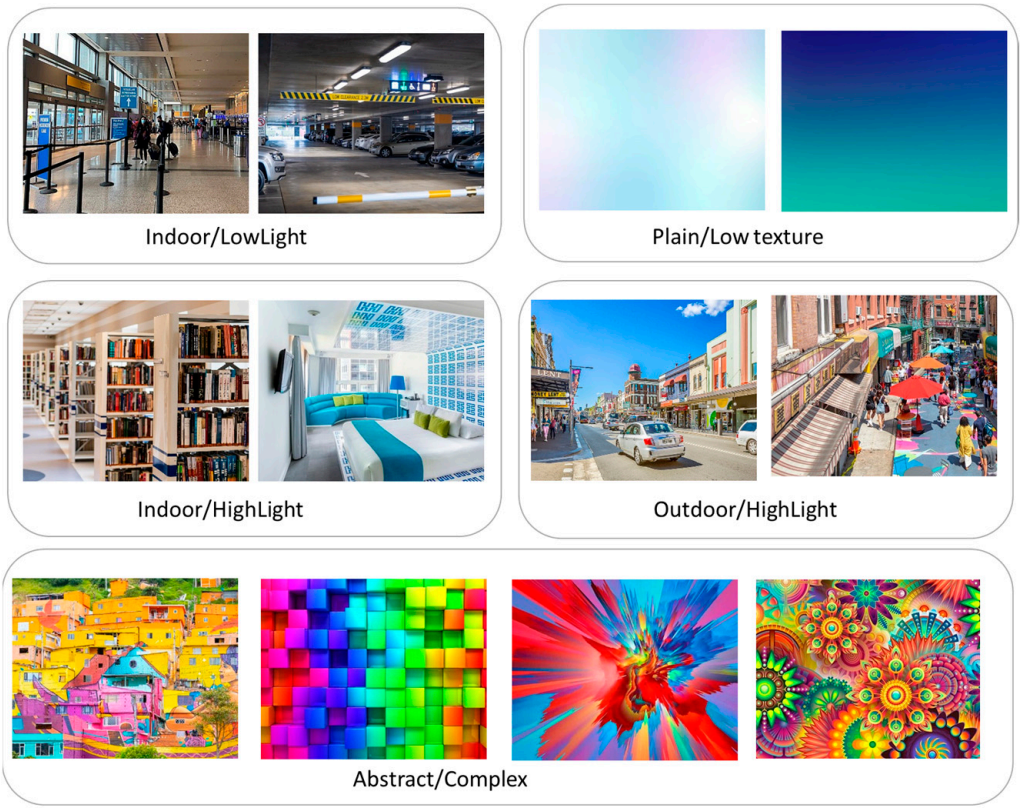

Figure 8. Sample Image of Background.

Background replacement introduces environmental variation and increases the capacity of our system to recognize hand gestures in more challenging scenarios. This diverse background helps train our system to adapt to unexpected ecological changes, better preparing it for real-world deployment challenges. Here is an overview of the background replacement process in Figure 9:

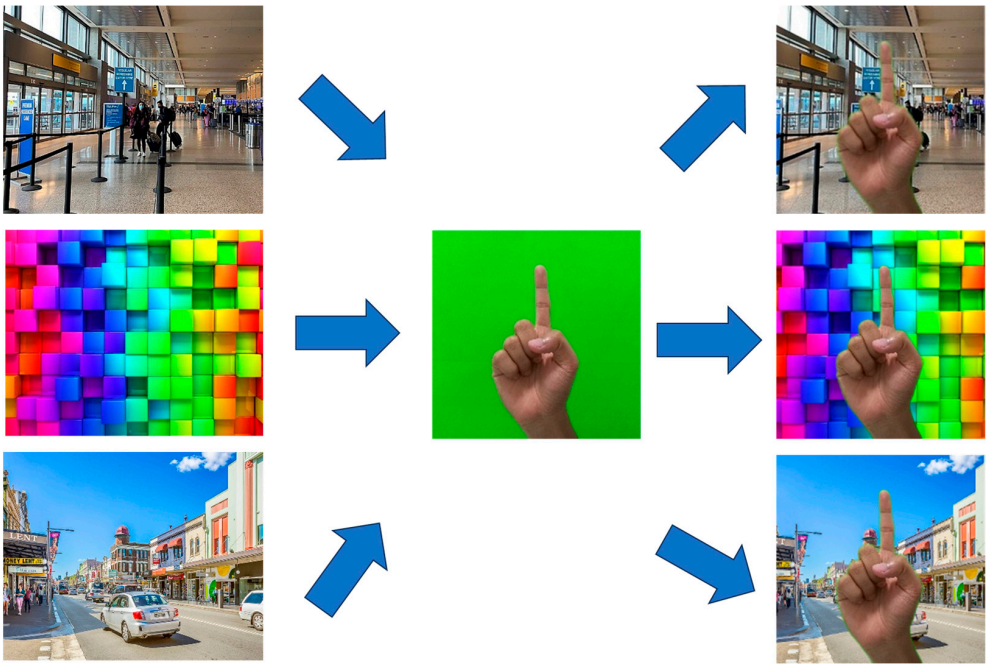

Figure 9. Background changing to make the variation.

Temperature

As a part of augmentation techniques to simulate the real world for handling accuracy in recognition, temperature augmentation modifies captured images to represent diverse lighting conditions and color temperature changes. By applying controlled adjustments to the color channels, such as increasing or decreasing red or blue intensity, images can be transformed to mimic shifts in color temperature. This process aids in training models to recognize objects or gestures accurately in environments with varying color temperatures, ensuring that they perform reliably across different lighting conditions. Temperature augmentation is essential in the arsenal of techniques to enhance machine-learning models' robustness and improve their performance in real-world scenarios. The following is an illustration of temperature augmentation as shown in Figure 10:

Figure 10. Sample of an image using Temperature Augmentation.

Blurriness

Blurriness is one of the image augmentation techniques utilized in image processing and machine learning. This technique aims to simulate real-world conditions by introducing an element of fuzziness or blur to images. Image blur occurs when the details of an image become less sharp or reduced, often resembling a slightly out-of-focus image. The application of blur introduces factors that might exist in everyday environments, affecting image sharpness.

There are several crucial aspects to consider in the application of blurriness as an augmentation technique:

(a) Simulation of Movement or Vibration: In real-world scenarios, images can often become blurred due to camera shake, hand movements, or object motion. By introducing elements of blurriness, we can simulate this effect and train the model to recognize images that might not always be perfectly sharp.
(b) Variations in Lighting: Uneven or changing lighting conditions can make images less sharp. By applying blur, we can create variations in lighting in the training dataset, allowing the model to understand objects under different lighting conditions.
(c) Distance Effects: Images taken from a distance are often blurry. By adding blur elements, we can depict distant objects more realistically, enabling the model to recognize these objects in real-life conditions.
(d) Visual Uncertainty: Not all objects in an image will always be sharp in real-world situations. Some image elements may appear blurry or less sharp, and the model should be capable of identifying these objects in everyday conditions.

Implementing blurriness as a dataset augmentation technique helps the model become more tolerant of variations in real-world conditions. This makes the model more adaptive and reliable in recognizing objects or situations that may not always be perfect or sharp in images. Blurriness augmentation is another valuable tool in training models to face more realistic and dependable conditions in real-world applications. For an overview of blurriness augmentation is depicted in Figure 11.

Figure 11. Blurriness Augmentation.

2.1.3. Dataset

In this study, we adopt a highly inclusive comprehensive approach to evaluate our Hand Gesture Recognition (HGR) model. This approach involves the combination of various datasets, including a meticulously created custom dataset. This custom dataset has undergone several treatments to simulate real-world conditions, creating a more representative testing environment. It is also essential to understand the concept of a public dataset, which researchers created openly. Public datasets encourage collaboration, standardization, and advancement within a specific domain. We aim to create a holistic approach to HGR model development by merging custom and public datasets.

In the hand gesture recognition (HGR) context, most models have historically operated in a user-dependent mode. This implies that the training and testing data are derived from a single dataset. However, in this research, we aim to develop a user-independent model. In other words, training and testing data will involve various individuals with diverse hand characteristics. This is highly relevant for practical applications because many individuals with varying hand features will use the HGR system we are developing. Combining these two dataset types, we aim to create a robust HGR model that addresses challenges arising from real-world environmental variations. Our comprehensive approach allows our research to bridge the gap between controlled laboratory conditions and the complexity of unpredictable real-world scenarios. Consequently, our research findings are expected to be relevant in real practical situations.

It is essential to note that both dataset types used in this research adhere to applicable privacy and usage guidelines. Throughout this research, we are highly committed to ethical considerations, including protecting individual privacy.

Primary Dataset/Custom Dataset

Our created dataset is integral to our research and has been meticulously crafted with specific attributes. These hand gesture images were captured in a controlled indoor environment, illuminated by commonly used artificial lighting. We employed commonly available smartphone cameras to ensure realistic representations of real-world scenarios. This dataset includes hand images from individuals with unique hand characteristics and variations. This diversity is essential for training the hand gesture recognition model to perform effectively in practical applications where user hand characteristics vary.

Our data collection process was conducted against a consistent green screen background, which is crucial for the background replacement technique later. This green screen background ensures the model's versatility in various real-world environments.

Throughout the dataset creation, we applied various data augmentation techniques. These techniques encompass background changes to simulate multiple environmental conditions, geometric transformations for hand positions and orientation variations, brightness adjustments to accommodate lighting changes, and image temperature modifications to emulate various temperature settings. Each action aims to enrich our training dataset, equipping our model with the flexibility needed to tackle challenges that arise in everyday situations. An example of sample images from the dataset we created is shown in Figure 12.

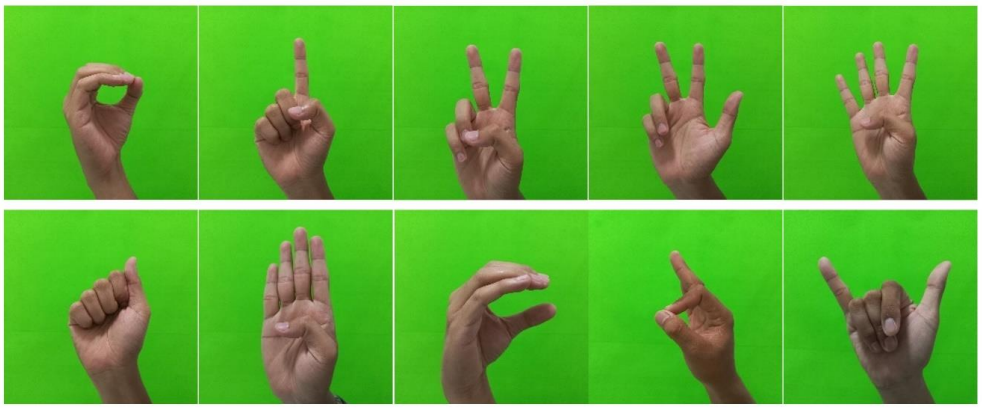

Figure 12. Sample image of Custom Dataset using Greenscreen.

Public Datasets/Secondary Dataset

We utilize a public dataset comprising several sub-datasets to assess our model's reliability in various real-world scenarios. These sub-datasets include Massey Hand Image ASL, Digit and Alphabet, NUS-II, Sebastian Marcel, and Hand Gesture (HG14) 14, each with distinct characteristics and data sources. Each sub-dataset contains diverse images displaying unique hand gestures. The public dataset is utilized as the testing data to assess the efficacy of our model, which has been trained using a custom-created dataset subjected to diverse augmentation to replicate real-world scenarios. The explanation of each sub-dataset will be described in the paragraph below:

Massey University (MU) HandImages American Sign Language (A.S.L)

Barczak and colleagues conducted a study at Massey University (M.U.) in New Zealand, where they created the MU HandImages A.S.L. dataset. This dataset comprises 2425 images captured from five individuals, each displaying various hand gestures. The photographs were taken in different lighting conditions and against a green screen background. The dataset comprises 26 classes representing fundamental American Sign Language (ASL) movements. The images in the dataset have a black backdrop, and their pixel sizes may vary depending on the hand posture. Figure 13 [69] visually represents some images from this dataset. This dataset includes gestures representing digits and alphabets, making it a valuable resource for research and applications.

Figure 13. The MU HandImages A.S.L. dataset is a collection of 26 unique hand gestures that are frequently employed in American Sign Language (ASL).

Sebastian Marcel Static Hand Gesture Dataset

The Sebastian Marcel Static Hand Gesture Dataset [70] was used to train a neural network model to detect hand positions in photographs. Space discretization was used to separate hand movements based on facial position and body anthropometry. The dataset has ten persons demonstrating six hand postures (a, b, c, point, five, and v) in uniform and complicated backdrops with different picture sizes based on the hand gesture. Figure 14 [70] depicts several photos from this collection.

Figure 14. Sample images from the Sebastian Marcel Static Hand Gesture Dataset featuring six distinct hand gestures demonstrated by ten individuals.

The NUS Hand Posture II

NUS Hand Posture Dataset includes hand posture images captured in and around the National University of Singapore (NUS) against complex backgrounds. It consists of 10 classes of hand postures performed by 40 subjects of different ethnicities, genders, and ages (22 to 56 years). Each subject demonstrated the ten hand postures five times, incorporating natural variations.

The dataset is divided into three folders: "Hand Postures" (2000 images), "Hand Postures with human noise" (750 images), and "Backgrounds" (2000 images). The hand posture images are available in grayscale and color, with resolutions of 160 × 120 and 320 × 240 pixels. The images in the "Hand Postures with human noise" folder include additional elements like the face of the poser and humans in the background.

All images are in RGB format and saved as JPEG files. The dataset has been used in academic research, and the results of hand posture detection and recognition are reported in the paper titled "Attention Based Detection and Recognition of Hand Postures Against Complex Backgrounds" by Pramod Kumar Pisharady, Prahlad Vadakkepat, and Ai Poh Loh [71]. Figure 15 [71] shows sample images of this dataset.

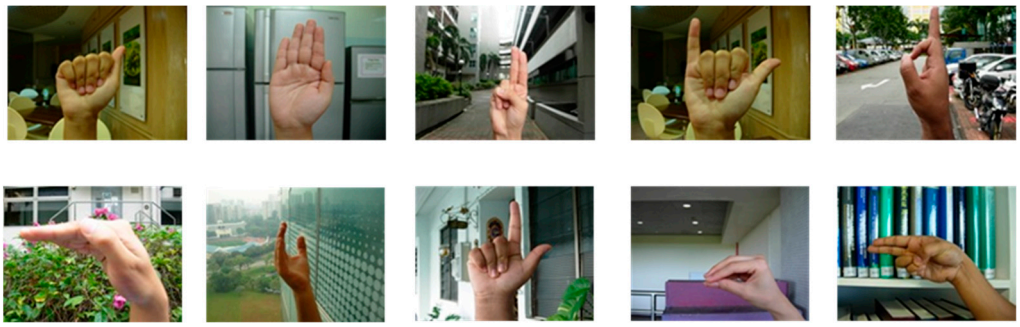

Figure 15. Sample image of NUS Dataset consists of Hand Posture, hand Posture with noise and background.

HG 14

The Hand Gestures 14 (HG14) dataset, developed by Guler et al. [72], comprises 14 hand gestures that are appropriate for hand interaction and application control in aug-

mented reality. The dataset comprises 14,000 photographs, each containing RGB channels and occupying a 256 by 256 pixels resolution. Every image is accompanied by a background that is both simple and uniformly colored, as depicted in Figure 16 [72].

Figure 16. Sample images from the HG14 dataset showcasing the 14 distinct hand gestures.

2.1.4. Deep Learning and Neural Network

Deep learning is a cutting-edge method in the machine learning domain that aims to understand and manage complex information from large data sets automatically. Inspired by how the human brain works, this approach leverages artificial neural networks to handle complex data analysis tasks. Neural networks consist of a series of interconnected layers of neurons, where each layer has its role in processing and interpreting data. At the peak of deep learning progress, there are various architectures, such as forward neural networks (FNNs), convolutional neural networks (CNNs), recurrent neural networks (RNNs), and generative adversarial networks (GANs). These models continue to evolve, adapt to various problems, and demonstrate the changing dynamics of research and applications in deep learning. The following is the architecture and approach used in this research.

Convolutional Neural Network

Deep Learning (DL) has various architectures, one of which is Convolutional Neural Networks (CNN), known for its effectiveness in image recognition compared to traditional machine learning approaches [32]. The basic idea of CNN is the technique of image convolution, which combines an input matrix and a kernel matrix to produce a third matrix that represents how one matrix is modified by the other.

A CNN architecture generally consists of two parts: feature extraction and classification [73], as depicted in Figure 17. The feature extraction part applies image convolution to the input image to produce a series of feature maps. These features are then used in the classification part to classify the label of the input image.

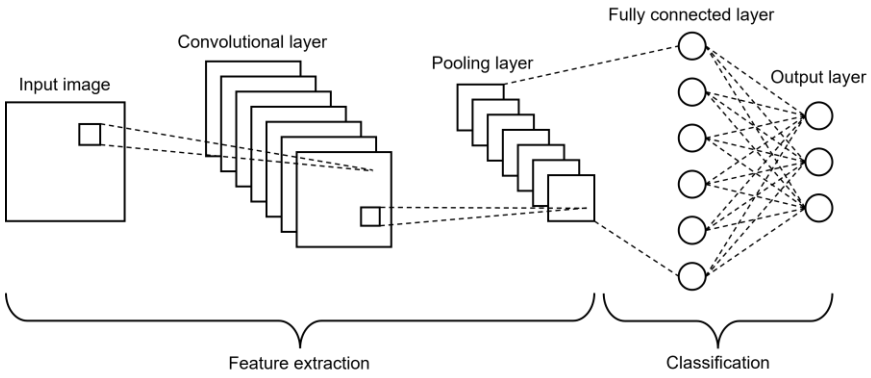

Figure 17. CNN architecture with feature extraction and classification parts.

CNN utilizes convolution to generate a collection of feature maps. Each filter can identify distinct patterns within the input image, including but not limited to edges, lines, and corners. To incorporate non-linearity and acquire intricate representations of the input

image, the output of the convolutional layer is subjected to a non-linear activation function, such as the Rectified Linear Unit (*ReLU*). The *ReLU* function is defined in Equation (10) [74].

$$ReLU(x) = max(0,\ x) \tag{10}$$

The pooling layer is employed after the convolutional layer to decrease the dimensionality of the feature maps and introduce translational invariance. Maximum pooling selects the highest value inside a specific local area, whereas average pooling computes the average value.

Pre-Trained Neural Network

The primary challenge in training CNN models is the need for powerful computational resources, such as Graphics Processing Units (GPUs), and sufficient memory capacity to store datasets and CNN parameters during training [73]. To overcome these constraints, researchers have developed pre-trained CNN models on large public datasets such as ImageNet or Microsoft COCO [75]. Research has shown that these pre-trained models can be a solid foundation for building image classification systems with a deep understanding of visual content. Transfer learning or fine-tuning of pre-trained models has become a common approach in current research due to its advantages. Several model optimization techniques have been discussed in the literature, including the use of model optimization methods like Stochastic Gradient Descent (SGD), Adam, and RMSProp. In several fields of image recognition and classification research, the usage of pre-trained Convolutional Neural Network (CNN) models such as ResNet-50 has become a strategically important decision. ResNet-50, with its wide and deep architecture, has proven to provide an impressive level of accuracy in various tasks [76]. In the context of achieving effectiveness, a CNN model needs to have adequate depth and be trained on a large dataset, allowing the model to understand various patterns in image data [73].

In a study that focused on using ResNet50 and InceptionV3 models, model optimization based on SGD, Adam, and RMSProp was used to train fine-tuned CNN models on a cat vs. dog dataset. The results showed that the SGD optimizer outperformed the other two for ResNet50, achieving a training accuracy of around 99% [77]. In another context of hand gesture recognition using electromyography (EMG) signals, the deep learning-based approach has yielded impressive results. This study, the ResNet-50 model was used to classify various hand gestures with a testing accuracy of approximately 99.59% [78].

Furthermore, the use of ResNet-50 in image recognition has been the focus of another research, such as in emotion detection [79]. Emotion detection is a key component in developing intelligent human-computer interface systems. Current emotion detection systems typically assume the availability of full, unobstructed facial images. In this study, convolutional neural networks (CNNs) with transfer learning are used to recognize seven basic emotional states, such as Anger, Contempt, Disgust, Fear, Happiness, and Sadness. The research compares the performance of three pre-trained networks: VGG16, ResNet50, and a modified version of ResNet50 that integrates a new architectural component called squeeze and excitation (SE-ResNet50). These networks are trained using a dataset of facial images, and their results are carefully evaluated and compared. The achieved validation accuracies are quite impressive, with VGG16 reaching 96.8%, ResNet50 reaching 99.47%, and SE-ResNet50 reaching 97.34%. In addition to accuracy, precision and recall are also considered as performance metrics in this study. The evaluation shows that all three networks can detect emotions accurately, but ResNet50 emerges as the most appropriate and reliable choice among them. This highlights the superiority of the pre-trained ResNet50 model in accurately recognizing emotions from facial images.

The use of the ResNet-50 model has been a focal point in research on Masked Face Recognition [80]. This phenomenon has emerged since the coronavirus pandemic in December 2019, triggering a significant increase in interest in enhancing facial recognition systems. The urgent need to protect the public from virus transmission has strengthened this drive. However, the preventive measures implemented to control virus spread present

challenges for security and surveillance systems, particularly in recognizing faces wearing masks. To address this issue, the research faces constraints related to the lack of adequate datasets for masked face recognition. Available datasets prioritize faces with Caucasian features, while faces with Ethiopian racial features are often overlooked. Therefore, this study formulated a specific dataset to overcome these limitations. This study conducted a comparative analysis among three leading neural network models: AlexNet, ResNet-50, and Inception-V3. They underwent testing to determine their ability to identify faces covered by surgical, textile, and N95 masks, among other forms of masks. The research findings demonstrate that CNN models can achieve very high levels of recognition accuracy, both for faces wearing masks and those without. Furthermore, model performance analysis indicates that ResNet-50 stands out by achieving the highest accuracy, reaching 97.5%. This finding underscores the superiority of ResNet-50 in recognizing faces wearing masks compared to other models, such as Inception-V3 and AlexNet. From the results of this study, it can be concluded that the use of the ResNet-50 model makes a significant contribution to improving the accuracy of masked face recognition, making it the preferred choice in addressing the challenges of face recognition in the pandemic era.

Furthermore, research related to brain tumor classification also supports the use of the ResNet-50 model. A framework that utilizes optimal deep learning features has been proposed for brain tumor classification. In this experiment, the ResNet-50 model was used with transfer learning, and the results showed significant accuracy for brain tumor classification [81].

Thus, the use of pre-trained ResNet-50 models supports the success of these studies and demonstrates their enormous potential in realizing intelligent and accurate solutions for various challenges in various disciplines. With the continuous advancement of knowledge and technology, it is expected that the role of this model in future research will become even more prominent and provide increasingly significant benefits to the broader society.

2.1.5. Evaluation Metric

To assess the performance of our model, we utilized accuracy as the primary evaluation metric. In this paper, accuracy is used to measure the result, and accuracy refers to the overall performance of the classifier. The classifier's capacity to accurately differentiate between genuine labels can be described. Nevertheless, a significant limitation of employing accuracy arises when there is an imbalance, when the positive and negative classifications are not evenly distributed. Equation (2) provides a formal definition of accuracy [82].

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{11}$$

The variable *LP* represents the count of correctly categorized true positive labels, *LN* represents the count of correctly classified true negative labels, *FP* represents the count of incorrectly classified false positive labels, and *AN* represents the count of incorrectly classed false negative labels.

2.2. Research Methodology

As this research explores the efficiency of the green screening technique in replacing backgrounds to simulate diverse environments and assess its impact on hand gesture recognition accuracy, several key research questions have been posed to guide our investigation. These questions serve as the foundational pillars upon which our study is constructed, enabling us to delve into the intricacies of image augmentation strategies and their practical implications. The following research inquiries define the scope of our exploration:

(a) Integration of green screening technique to replace backgrounds for simulating diverse environments.
(b) Analysis of the impact of hybrid image augmentation on Hand Gesture Recognition Accuracy:
(c) Quantitative Assessment of Test Accuracy Post-Augmentations with a Public Dataset.

(d) Exploration of the extent of Classical Augmentation for Generating Varied Data while Maintaining Accuracy
(e) Investigation into the Contribution of Green Screen Dataset for Implementing Hybrid Image Augmentation.

The research methodology designed to address these inquiries is visually illustrated Figure 18. This multi-step approach begins with creating a base dataset featuring hand gestures performed against a consistent green screen background. To ensure an ample volume of data for analysis, additional datasets are procured and meticulously replicated 30 times, expanding the dataset's size significantly.

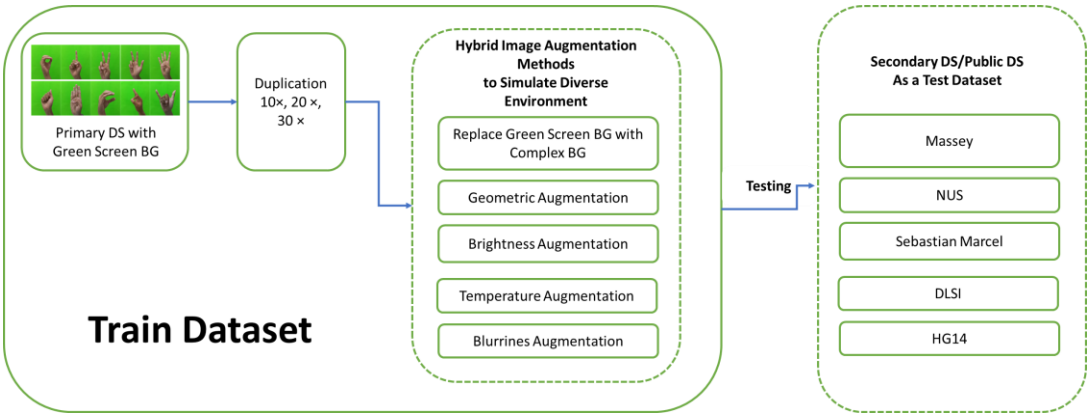

Figure 18. Overview of the Research Methodology in this Study.

In the subsequent phases, critical augmentations are introduced. The green screen background, the constant backdrop in the initial dataset, is thoughtfully substituted with backgrounds that span a spectrum of colors and settings. These backgrounds include walls, wooded areas, gradient patterns, airports, basements, vibrant and colorful locations, libraries, rooms, and urban streets. The deliberate inclusion of these diverse backgrounds is pivotal to mimicking a real-world environment characterized by many potential scenarios.

Continuing to steps 3 and 4, classical augmentation techniques come into play. Geometric transformations, brightness adjustments, temperature shifts, the introduction of blurriness, and a pretrained ResNet50 are applied systematically to replicate the unpredictability and uncontrolled aspects of real-world environments. These alterations significantly enrich the dataset and prepare it for testing.

The final dataset, a product of these meticulous augmentations and background substitutions, is subjected to rigorous testing. The testing phase involves an examination of its performance against publicly established datasets obtained from Massey, NUS, Sebastian Marcel, DLSI, and HG14. Each public dataset is assessed independently to gauge the impact and effectiveness of the hybrid image augmentation strategy. This approach ensures that the accuracy of each dataset is rigorously evaluated and contributes to a comprehensive understanding of the method's performance across various datasets.

3. Results

3.1. Experimental Setup

This study aims to simulate hand gesture recognition in uncontrolled environments or real-world conditions. It involves treating the initial dataset, which uses a green screen background, by replacing or adding backgrounds that reflect various other locations. Classical augmentation techniques such as geometry, brightness, temperature, and blurriness enhance this simulation.

The study adopts an "on the fly" augmentation strategy to achieve this goal and defines various parameters for each augmentation technique, as shown in Table 1. Augmentations are performed in a series of steps, starting with background augmentation (1), followed by

geometry transformation (2), brightness (3), temperature (4), and blurriness (5). The deep learning algorithm's training phase involves each stage's application.

Table 1. Parameter setting for all Augmentation Techniques.

Category	Technique	Parameter Setting	Value/ Range	Direction	Description
Geometric Transformation	Rotation	rotation_ range	10°	Positive (Clockwise)/Negative (Counterclockwise)	Rotation of the image within the range of −10 degrees to +10 degrees.
	Translation	width_shift range	0.1	Positive (Rightward)/Negative (Leftward)	Shifting the image width within the range of −10% to +10% of the width.
		height_shift_ range	0.1	Positive (Downward)/Negative (Upward)	Shifting the image height within the range of −10% to +10% of the height.
	Shearing	shear_range	10°	Positive (Right Shear)/Negative (Left Shear)	Shearing the image within the range of −10 degrees to +10 degrees.
	Scaling	zoom_range	[1, 1.5]	Positive (Zoom In)/Negative (Zoom Out)	Scaling the image within the range of 1 to 1.5 times the original size.
	Flipping	Horizontal Flip	Enabled/ Disabled	Horizontal Flip	This enables horizontal flipping. It means the image can be horizontally flipped (resulting in a mirrored image).
Brightness	Adjustment	brightness	20	Positive (Brightening)/Negative (Darkening)	Changing the brightness level within the range of −20 to +20.
Temperature	Adjusting	temperature	20	Positive (Warming–Red)/Negative (Cooling–Blue)	Adjusting the image's color temperature within the range of −20 to +20.
Blurriness	Randomly	blurriness	Random	Blurring (No explicit direction)	Randomly adding blur to the image.

Before augmentation, the initial dataset is duplicated 10, 20, 30 times. Each copy of the dataset then undergoes a series of augmentations. The result is a training dataset with increased variability. This dataset will be tested to evaluate its impact on model accuracy. Thus, we hope this research can provide valuable insights into using augmentation in the context of hand gesture recognition in real-world environments.

Transfer Learning uses pre-trained weights from ImageNet as the initial weights for the network to circumvent the need for intricate and computationally intensive learning processes. Weight optimization is limited to the classification layer of the pre-trained model, where it is solely used to optimize the networks in the fully connected layers. The Optimizing technique known as Adaptive Moment Estimation (ADAM) is used to improve the training process and prevent gradient vanishing during training [83]. The pre-existing network undergoes retraining for a total of 50 epochs, employed with a batch size of 32. To preserve the pre-trained weights of ImageNet, the process of network training involves the freezing of all layers throughout the feature extraction phase. The evaluation of network performance is conducted by employing the "accuracy" metric.

The experimental setup employs the Python programming language, with various libraries like TensorFlow, Matplotlib, and NumPy. The experiment is conducted on a personal computer, with the specifications specified in Table 2.

Table 2. Hardware and software specifications for the experiment.

Hardware/Software	Specification
Processor (CPU)	Intel Core i5-9300H @2.40 GHz
Memory (RAM)	32 GB DDR4
Graphical Processing Unit (GPU)	Nvidia GTX 1660 Ti–6 GB vRAM
Operating System	Windows 11 Home Edition
Python version	3.6.13
Cuda/CuDNN version	11.0/8.0

3.2. Programming Tools

In the research methodology depicted in above Figure 18, there are instructions that state, This research evaluates the accuracy of gesture classification experiments using a primary dataset that employs the proposed green screen technique to replace the background with complex backgrounds, such as images from indoor and outdoor locations, along with backgrounds experiencing color distortion. Additionally, classic augmentations like geometric transformations, brightness adjustments, temperature changes, and blurriness are applied to simulate an uncontrolled environment. The primary dataset is then tested with a secondary dataset of existing public datasets to assess its accuracy".

These instructions are then translated using a computer with an Intel Core i5 9th generation processor, 32 GB RAM, Nvidia 1660Ti Graphics Card, programmed with Python 3.6.13, and utilizing tools from TensorFlow 2.6.2 and Keras 2.6.0 by importing various libraries and modules needed for model development and image augmentation. Although this code leverages Keras functionality, the focus is on TensorFlow as a broader deep-learning framework. Subsequently, the definition of training, testing, and background directories is carried out, forming the foundation for loading and processing data.

```
"'python
# Defining dataset directories
train_dir = "greenscreen dataset"
test_dir = "public dataset"
bg_dir = "background image"
"'
```

Next, the training dataset is duplicated to enhance data variation, where the number of duplicates is determined by the "num_duplicates" variable. The preprocessing function is then defined to prepare the background and convert images into NumPy arrays for 30 iterations as needed.

```
"'python
num_duplicates = 30
input_folder = train_dir
duplicate_folder = train_dir + "_" + str(num_duplicates) + "x"
for root, _, filenames in os.walk(input_folder):
# . . .
"'
```

Furthermore, the "**preprocess_image**" and "**change_background_v2**" functions play a crucial role in preparing images and replacing the green background with randomly selected background images, introducing variation in hand gesture backgrounds. These functions handle the preprocessing of input images during data augmentation. The "**preprocess_image**" function selects a random background from a predefined list, adjusts brightness and temperature, replaces the background using "**change_background_v2**", and applies a blur effect using **"blur_image**". The "change_background_v2" function replaces the green background in the image with a randomly selected background. The "**adjust_brightness_temperature**" function adjusts the brightness and temperature of the hand object in the image.

```
“‘python
def preprocess_image(image):
# . . .
return img_array
def blur_image(image):
# . . .
return image
def change_background_v2(img, bg):
# . . .
return img_edit
def adjust_brightness_temperature(image, bg):
# . . .
return result_image
”’
```

Next, the configuration of the “ImageDataGenerator” is set for training data with various augmentation techniques, including rotation, horizontal and vertical shifts, shear, zoom, and custom preprocessing using the “**preprocess_image**” function.

```
“‘python
train_datagen = ImageDataGenerator(
rotation_range = 10,
width_shift_range = 0.1,
height_shift_range = 0.1,
shear_range = 10,
zoom_range = [1, 1.5],
fill_mode = ‘nearest’,
preprocessing_function=preprocess_image
)
”’
```

(a) Finally, a CNN model is built using the ResNet50 architecture as the base, with additional layers suitable for the hand gesture recognition task. Here’s a detailed explanation of each step in building a CNN model using the ResNet50 architecture:

(b) Building the Base Model with ResNet50:

- ResNet50 is a Convolutional Neural Network (CNN) architecture developed by Microsoft Research. It consists of 50 layers (hence the “50” in the name) and has been proven highly effective in various computer vision tasks, especially image classification.
- ResNet50 (include_top = False, weights = ‘imagenet’, input_shape = (224, 224, 3))”: This function creates the base ResNet50 model. The argument “include_top = False” indicates that the fully connected layers at the top of the model will not be included, allowing us to customize the top layers according to our task. “Weights = ‘imagenet’ initializes the model with weights learned from the “imagenet” dataset, enabling the model to have an initial understanding of various features present in images. The argument “input_shape = (224, 224, 3)” specifies the size and color channels (RGB) of the input images that the model will receive.

(c) Setting Trainable Layers

After creating the base ResNet50 model, we set all its layers to be non-trainable (“layer.trainable = False”). This step ensures that the weights learned by the ResNet50 model are not altered during the new model’s training process. We want to leverage the features learned by the ResNet50 model on the “imagenet” dataset without modifying them.

(d) Adding Additional Layers:

- After the base ResNet50 model, we add several additional layers on top of it to tailor the model to the hand gesture recognition task.

- Flatten(): This layer flattens the output of the base model into one dimension. This is necessary because the subsequent Dense layers require input in the form of a one-dimensional vector.
- Dense(512, activation = 'relu'): This Dense layer consists of 512 neuron units with the ReLU activation function. Dense layers like this aim to learn more abstract feature representations from the image data.
- Dropout(0.5): The Dropout layer is used to reduce overfitting by randomly deactivating some neuron units during the training process.
- Dense(train_generator.num_classes, activation = 'softmax'): The final Dense layer has the same number of neuron units as the number of classes in the training dataset, and it uses the softmax activation function to generate probabilities of possible classes.

(e) Compiling the Model:

- After adding the additional layers, the model needs to be compiled before it can be used for the training process.
- model.compile(optimizer = 'adam', loss = 'categorical_crossentropy',metrics = ['accuracy']): In this step, we specify the optimizer to be used (in this case, the Adam optimizer), the loss function appropriate for the multi-class classification task (categorical cross-entropy), and the metrics to be monitored during training (in this case, accuracy).

With the above steps, we have successfully built a CNN model using the ResNet50 architecture as the base, ready to be trained for the hand gesture recognition task.

```
'''python
base_model = ResNet50(include_top = False, weights = 'imagenet', input_shape = (224, 224, 3))
for layer in base_model.layers:
layer.trainable = False
x = base_model.output
x = Flatten()(x)
x = Dense(512, activation='relu')(x)
x = Dropout(0.5)(x)
predictions = Dense(train_generator.num_classes, activation='softmax')(x)
model = Model(inputs = base_model.input, outputs = predictions)
model.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = ['accuracy'])
'''
```

This experiment is directed towards each public dataset to understand the extent of the impact of hybrid augmentation on accuracy results.

3.3. Data Preparation

This chapter will explain the data preparation steps we have undertaken to support this research. Data preparation is a crucial stage in ensuring the quality and integrity of the data before using it for model training and evaluation in sign language recognition.

In this research, we used various data sources, including data we created as training data, publicly available datasets as testing data, and a collection of background images depicting various outdoor scenarios, gradients, and colors. There are five public datasets, and each has its dataset with a greenscreen as the background. The greenscreen dataset underwent a background replacement process and the addition of classical augmentation to simulate diverse environments. Subsequently, the prepared datasets are used to evaluate the effectiveness of various experimental scenarios, as summarized in Table 3.

Table 3. Description of Dataset.

Category	Name of Dataset	Number of Data	Number of Classes	Images Size	Image Background/Image Consist
Public (As a Testing Data)	HG14	14,280	14	256 × 256	uniform
	MU HandImages ASL (Digit 0–9)	700	10	vary	uniform
	MU HandImages ASL (Alphabet)	3490	26	vary	uniform
	Sebastian Marcel	659	6	vary	uniform & complex
	NUS-II	2000	10	160 × 120	complex
Custom Dataset (using Green Screen B.G. for each Public Dataset as Training Data)	HG14	280	14	224 × 224	greenscreen
	MU HandImages ASL (Digit 0–9)	201	10	224 × 224	greenscreen
	MU HandImages ASL (Alphabet)	781	26	224 × 224	greenscreen
	Sebastian Marcel	120	6	224 × 224	greenscreen
	NUS-II	210	10	224 × 224	greenscreen
Image Background for replacing B.G. Greenscreen	-	90	-	400 × 320	various outdoor scenarios, gradients, and different colors

In this study, we applied a series of data preprocessing techniques to ensure consistent and reliable characteristics in the dataset. These techniques include:

(a) Image Resizing: All images in our dataset were resized to 224 × 224 pixels with three color channels (RGB). This step is essential to match the image format commonly used in Convolutional Neural Network (CNN) models like ResNet and Inception.

(b) Pixel Normalization: We performed pixel normalization to ensure that pixel values have a uniform scale. This allows the model to understand patterns without being affected by variations in pixel values.

(c) Data Augmentation: Data augmentation techniques enhanced the dataset's variability. This includes image rotation, horizontal and vertical shifts, shear transformations, and image zooming. Additionally, geometric-based data augmentation, brightness, temperature adjustments, and image blurring were applied to some images. These techniques enrich the dataset with more variations to help the model understand sign language gestures better.

(d) Data Duplication: Before we replaced the background, each image in our training dataset was duplicated 1, 10, 20, or even 30 times. This duplication served a specific purpose: significantly increasing the volume of training data and introducing a more comprehensive array of variations into our dataset. By duplicating the images, we took an essential initial step before altering the background, ensuring our dataset was diverse. This approach was imperative given the relatively small size of our original training data.

In this context, the processed data includes our created dataset, consisting of gesture images with a green background. Each image in this dataset was duplicated 30 times before the background replacement. This data underwent various augmentation steps, including brightness and temperature adjustments and blurring.

Furthermore, the prepared datasets are trained using the transfer learning method with a pre-trained ResNet50 model to optimize the training process. After training, testing

is carried out on the relevant public dataset for each specific dataset. The results of these tests provide information about the accuracy of each dataset.

Through this approach, we can conduct a more in-depth analysis to measure the extent to which our applied approach succeeds when tested on various public datasets.

3.4. Experimental Results

This experiment is aimed at explaining based on the scenarios in the previous section, which are divided into the testing results for each dataset. Along with the differences in the number of classes in each dataset and the balancing performed, the only performance metric used is accuracy. We have conducted a series of experiments to explain the role of the green screening technique, hybrid image augmentation methods, and using datasets with green screen backgrounds to improve the accuracy of sign language recognition in various situations. The results of these experiments provide a deeper understanding of the impact of combining multiple techniques and the extent to which they can achieve high accuracy.

Diving deeper into the test results, let's correlate our findings with the quantitative data, providing a more comprehensive understanding of each test's implications and the varying outcomes across the different datasets.

In a comprehensive analysis of various datasets, significant diversity in outcomes with complex patterns emerged. This is associated with the effectiveness of this study's augmentation and data duplication strategies in exploring the accuracy of hand gesture recognition models with diverse datasets. Each dataset exhibits unique characteristics that respond to the applied techniques. Here are the results from each dataset:

Firstly, in the Sebastian Marcel (SM) dataset, we observed a relatively stable increase in accuracy with each augmentation level, ranging from single augmentation to all augmentations applied. As given in Table 4, initially, accuracy was consistently improved from no duplication (original) to 10× data duplication. However, at 20× duplication, there was a decrease in accuracy before increasing again at 30× duplication. From the analysis of augmentation stages, we found that a Single Augmentation Change Background on original data resulted in the lowest accuracy. This indicates that applying Change Background augmentation without duplication in the SM dataset is considered ineffective in improving accuracy. However, when we used data duplication, a significant increase in accuracy was observed, especially up to 10× duplication. This suggests that data duplication can overcome the weaknesses of single augmentation strategies. Fluctuations at 20× duplication may be caused by information overload or noise. However, the increase again at 30× duplication indicates that the excess data still provides informative value that can improve the model's accuracy. Additionally, changes in accuracy at each duplication stage may reflect the model's sensitivity to the quantity and variation of the data used. However, the decrease in accuracy at 20× duplication can also indicate a potential for overfitting, where the model becomes too biased towards the training data and loses the ability to generalize well to test data. Therefore, it is essential to carefully evaluate the model's performance on test data and make appropriate adjustments to the level of data duplication to avoid overfitting and ensure optimal model performance.

Table 4. Accuracy Results on the Sebastian Marcel Dataset with Various Data Augmentation Techniques for Each Duplication.

Duplication	Change Background	Change Background + Geometric	Change Background + Geometric + Brightness	Change Background + Geometric + Brightness + Temperature	Change Background + Geometric + Brightness + Temperature + Blur
Original	0.8027	0.8467	0.8346	0.8574	0.8741
10× Original	0.8801	0.8832	0.8923	0.9059	0.9272
20× Original	0.8270	0.8741	0.8771	0.8877	0.9241
30× Original	0.8528	0.8786	0.8816	0.8786	0.9272

Secondly, based on Table 5, the NUS II dataset demonstrates a dynamic response to augmentation, with almost all sequences of augmentation experiencing an increase in accuracy, except for background change augmentation without data duplication. For example, when using the original data, the accuracy is 0.815. However, as the duplication level increases, such as 10× (0.7875), 20× (0.7805), and 30× (0.8035), there is a consistent decrease. Although there is a slight increase at 30× duplication, the increase is insignificant. Additionally, for change background, geometric, and brightness augmentations, although there is an increase at the 10× duplication level, there is generally a decrease in accuracy. For instance, the original data's accuracy of changing background is 0.836. However, at 10× duplication, the accuracy increases to 0.8989 before decreasing at 20× (0.8974) and 30× (0.8874) duplications. Despite the significant increase in accuracy when all augmentations are applied, it is essential to consider the possibility of overfitting in the model. Overfitting occurs when the model becomes too "memorized" to the training data and loses the ability to generalize to new test data. The significant increase may indicate that the model has overly adapted to the training data, thus unable to generalize well to new data. Specifically, for change background augmentation, it is observed that the results tend to decrease at each duplication level, especially at 10× duplication. This suggests that background change without data duplication does not significantly improve the model's accuracy in the NUS II dataset. Furthermore, the increase observed at 30× duplication may result from random adjustments or luck rather than a natural effect of the augmentation.

Table 5. Accuracy Results on the NUS_II Dataset with Various Data Augmentation Techniques for Each Duplication.

Duplication	Change Background	Change Background + Geometric	Change Background + Geometric + Brightness	Change Background + Geometric + Brightness + Temperature	Change Background + Geometric + Brightness + Temperature + Blur
Original	0.8135	0.8290	0.8346	0.8535	0.8690
10× Original	0.7875	0.8905	0.8989	0.8830	0.9040
20× Original	0.7805	0.8915	0.8974	0.8895	0.9095
30× Original	0.8035	0.8924	0.8874	0.9059	0.9271

Thirdly, the Massey dataset demonstrates complex fluctuation patterns in accuracy at each level of augmentation duplication. This highlights the complexity of selecting appropriate augmentation and underscores that choosing the right augmentation strategy is complex. According to the results in Tables 6 and 7, the dataset's response to augmentation also confirms that not all augmentations contribute equally to accuracy improvement, highlighting sensitivity variations depending on the type of augmentation applied. The importance of selecting the proper augmentation is closely related to the issue of overfitting, where the model becomes too biased towards the training data and loses the ability to generalize patterns to new data. Using appropriate augmentation, we can help prevent overfitting by introducing additional diversity into the training data, allowing the model to learn more general patterns rather than just memorizing the training data. The results from the dataset show that there are specific duplication levels where certain combinations of augmentation yield the highest accuracy. For example, at the 30× duplication level, the combination of background, geometric, and brightness changes produces the highest accuracy. This suggests that by introducing more significant variation in the training data, the model can better understand underlying patterns, thus reducing the risk of overfitting. However, it should be noted that complex fluctuation patterns in accuracy can also be potential signs of overfitting. For instance, a drastic increase in accuracy at certain duplication levels may indicate that the model has "memorized" patterns in the training data rather than genuinely learning more general features. Therefore, careful evaluation of the model's response to augmentation is necessary to ensure that accuracy improvements are not solely due to overfitting.

Table 6. Accuracy Results on the Massey (Digit) Dataset with Various Data Augmentation Techniques for Each Duplication.

Duplication	Change Background	Change Background + Geometric	Change Background + Geometric + Brightness	Change Background + Geometric + Brightness + Temperature	Change Background + Geometric + Brightness + Temperature + Blur
Original	0.8043	0.8057	0.8657	0.8729	0.8929
10× Original	0.8229	0.8929	0.8857	0.9014	0.9157
20× Original	0.8057	0.8800	0.9157	0.9143	0.9071
30× Original	0.8086	0.8814	0.8986	0.9029	0.9000

Table 7. Accuracy Results on the Massey (Alphabet) Dataset with Various Data Augmentation Techniques for Each Duplication.

Duplication	Change Background	Change Background + Geometric	Change Background + Geometric + Brightness	Change Background + Geometric + Brightness + Temperature	Change Background + Geometric + Brightness + Temperature + Blur
Original	0.6667	0.7499	0.7322	0.7140	0.7190
10× Original	0.6171	0.7658	0.8006	0.7548	0.7840
20× Original	0.6193	0.7427	0.7168	0.7669	0.7614
30× Original	0.6705	0.7680	0.7972	0.7801	0.7652

Lastly, the HG 14 dataset presents challenges in improving accuracy, with the lowest accuracy values observed for some types of augmentation, as given in Table 8. This indicates that effective augmentation strategies may vary depending on dataset characteristics and require structured approaches and careful exploration.

Table 8. Accuracy Results on the HG 14 Dataset with Various Data Augmentation Techniques for Each Duplication.

Duplication	Change Background	Change Background + Geometric	Change Background + Geometric + Brightness	Change Background + Geometric + Brightness + Temperature	Change Background + Geometric + Brightness + Temperature + Blur
Original	0.4852	0.4740	0.4999	0.4758	0.4880
10× Original	0.5202	0.4964	0.5705	0.5705	0.5682
20× Original	0.5124	0.4902	0.4932	0.5588	0.5604
30× Original	0.5030	0.5064	0.4889	0.5547	0.5599

According to the average performances from each dataset, as shown in Table 9, it is essential to recognize that each dataset exhibits a unique response to data duplication and augmentation techniques. The "HG14" dataset is a prime example, where accuracy remains relatively low even after duplication and augmentation. The distinctive characteristics of this dataset significantly affect its response to these techniques.

The relatively lower accuracy in the HG14 dataset can be attributed to various factors. Despite its extensive image count, the dataset lacks diversity in hand gestures, which could lead to overfitting for specific poses. Additionally, while challenging to recognize hand gestures in various contexts, diverse backgrounds may introduce confusion due to stark background variations among hand gestures. Image quality issues, encompassing variations in contrast and background clarity, may further hinder precise recognition. Moreover, the dataset's augmentation methods might have inherent limitations or adverse effects on accuracy. It is imperative to consider potential algorithmic constraints when dealing with a large and diverse dataset like HG14. To enhance accuracy, further research

and experimentation may be necessary, focusing on augmentation techniques and training algorithms tailored to HG14's unique characteristics.

Table 9. Average each Public Dataset and Total.

Dataset	Change Background	Change Background + Geometric	Change Background + Geometric + Brightness	Change Background + Geometric + Brightness + Temperature	Change Background + Geometric + Brightness + Temperature + Blur
Sebastian Marcel	0.8407	0.8706	0.8714	0.8824	0.9131
NUS–II	0.7962	0.8758	0.8796	0.8830	0.9024
Massey–digit	0.8104	0.8650	0.8914	0.8979	0.9039
Massey–alphabet	0.6434	0.7566	0.7617	0.7540	0.7574
HG14	0.5052	0.4918	0.5131	0.5400	0.5441
Total Average	**0.719175**	**0.77197**	**0.783445**	**0.79143**	**0.8042**

Furthermore, the significance of establishing a baseline is evident in the "Massey—the alphabet" dataset. Despite a notable increase in accuracy when transitioning from "Original" to "10× Original Data" (0.6667 to 0.6171), the initial baseline accuracy was lower compared to other datasets. This improvement effectively brings it closer to the average accuracy of different datasets.

An essential aspect to assess is the potential for overfitting. Substantial accuracy increases from "Original" to "30× Original Data" in some datasets, such as "Sebastian Marcel" and "Massey—digit", may indicate the possibility of overfitting. While accuracy improves the training data, imbalanced results can be observed in the test data. Therefore, achieving the right balance in data duplication and augmentation is crucial to avoid overfitting.

Finally, let's consider the influence of the machine learning model. Based on the data, the ResNet50 model may be better suited for specific datasets. The "NUS–II" dataset consistently produces positive results, even with increased data duplication and augmentation. However, the model's response may vary depending on the dataset's characteristics. Understanding the model's role in the context of hand gesture recognition enables us to make more informed decisions in model development.

By reviewing the results in Table 8, we can correlate them with the unique characteristics of each dataset described earlier. In the "Sebastian Marcel" dataset, with its diverse backgrounds, complex augmentation combinations such as "Change background + geometric + brightness + temperature + blur" yield the best results (0.9131), emphasizing the need for robust augmentation when dealing with intricate backgrounds. Conversely, the "Massey–alphabet" dataset, which exhibits lower overall accuracy, lacks significant augmentation combinations, possibly due to the stable image characteristics in this dataset (0.7540–0.7617). However, in "HG14", which features relatively lower accuracy (0.4918–0.5441), improvements in accuracy are evident when employing more complex augmentation. This improvement may be attributed to the highly varied backgrounds in each hand gesture within the dataset. These insights underscore the necessity for further research and exploration of augmentations tailored to the specific characteristics of each dataset to enhance hand gesture recognition accuracy.

4. Discussion

However, it's essential to remember that there is no one-size-fits-all approach in hand gesture recognition. Each dataset possesses unique characteristics that influence how it responds to augmentation and data duplication techniques. Therefore, the selection and adaptation of techniques should be based on a deep understanding of the specific dataset's characteristics used in the research.

Based on our research findings, we draw several pivotal conclusions. The "green screen" technique substantially boosts hand gesture recognition accuracy across various

environmental conditions. Furthermore, implementing hybrid image augmentation yields a notable positive impact on accuracy. However, it's important to note that the extent of accuracy improvement varies depending on the characteristics of each dataset. This underscores the significance of tailoring techniques based on the unique characteristics of individual datasets. Employing a dataset employing the "green screen" technique also contributes to the success of hybrid image augmentation. Thus, the study highlights the importance of an adaptive approach to data processing techniques in significantly enhancing hand gesture recognition accuracy.

Despite the limitations posed by the available datasets, our research provides concrete evidence that with a careful approach and deep understanding of dataset characteristics, we were able to develop a hand gesture recognition model capable of overcoming various practical challenges. For instance, when faced with significant variations in lighting conditions across different environments, our model demonstrated its ability to recognize hand gestures accurately and consistently. Furthermore, our model could accurately distinguish hand gestures without background interference in complex or unpredictable background situations. Moving forward, we advocate for further research in applying augmentation techniques in hand gesture recognition, particularly on datasets featuring more diverse characteristics and complex environmental variations. Future studies could explore the implementation of adaptive augmentation techniques, where the type and complexity level of augmentation are tailored to the unique characteristics of each dataset. This approach could enhance the model's resilience to more extensive variations in real-world conditions, such as drastic lighting changes or unexpected backgrounds like crowds or moving objects.

Our research demonstrates that employing a series of augmentations, ranging from background changes to blurring, significantly improves hand gesture recognition accuracy. From a single type of augmentation alone, accuracy increased from 0.72 to 0.8042 when all kinds of augmentations were applied simultaneously. Combining various augmentation techniques can enhance the model's overall performance. The strength of our research lies in the comprehensive approach to improving hand gesture recognition accuracy by considering multiple environmental factors. Unlike the study by Luo, Cui, and Li [49], which focused on isolating hand regions based on skin color, our approach provides a more holistic solution by applying various augmentations. While Rahmat et al.'s research [50] was limited to improving object detection and lighting, we offer a broader approach by considering background variations, geometry, lighting, and blurring effects.

Compared to Yi Yao and Chang-Tsun Li's study [51], which used appearance-based methods, we substantially improved hand gesture recognition accuracy by applying various augmentations simultaneously. We successfully demonstrated that combining background changes, geometric augmentations, brightness enhancement, color temperature adjustment, and blurring could significantly improve hand gesture recognition accuracy.

Therefore, our research represents a holistic and comprehensive approach to enhancing hand gesture recognition accuracy by considering various environmental factors and implementing an effective series of augmentations.

5. Conclusions

This research has successfully developed a hybrid augmentation framework for enhancing the performance of hand gesture recognition systems in user- and environment-independent scenarios. The augmentation strategy incorporates background replacement, geometric transformations, brightness and temperature changes, and blurriness implementation. All these augmentations aim to generate more training data with variational conditions. From the experiments using several datasets, it is found that this hybrid augmentation strategy improves the classification accuracy by 8.5% on average. Besides that, adding the training data using 10×, 20×, and 30× duplications also helps to increase the recognition performance by up to 6% compared to just using the original training data. It is noted that the experiment was conducted using training data from a single person. Therefore, in the future, it can be observed whether the proposed augmentation strategy

will provide more significant improvement when more volunteers are involved in the image acquisition process.

Based on the discussion above, we can conclude several crucial points that mark a significant step forward in hand gesture recognition. First, using the "green screen" technique has revolutionized the accuracy of hand gesture recognition across various environmental contexts. By isolating the background, the hand gesture recognition model becomes more focused on the gestures, overcoming visual disturbances that may arise from complex or changing backgrounds. Second, innovative approaches in applying hybrid image augmentation have shown that geometric precision, lighting quality, and consistent backgrounds are key to improving accuracy. However, this improvement is not uniform and needs to be tailored to the unique characteristics of each dataset, emphasizing the need for dataset-based technique adjustments.

Moreover, the findings offer significant insights into the significance of tailoring augmentation strategies to the specific dataset employed. Datasets using the "green screen" technique demonstrate more substantial success in implementing hybrid image augmentation, reaffirming the need for adaptive strategies in data processing to achieve optimal accuracy. This conclusion paves the way for more detailed approaches in using augmentation techniques, especially on datasets with diverse characteristics and complex environmental variations. By providing a solid foundation, this research encourages further exploration into the potential application of hand gesture recognition technology in various broader application contexts.

Author Contributions: Conceptualization, B.-A.A.; methodology, B.-A.A.; software, B.-A.A.; validation, B.-A.A. and C.-T.C.; formal analysis, B.-A.A.; investigation, B.-A.A.; resources, B.-A.A.; data curation, B.-A.A.; writing—original draft preparation, B.-A.A.; writing—review and editing, B.-A.A. and C.-T.C.; visualization, B.-A.A.; supervision, C.-T.C. and J.-S.C.; project administration, J.-S.C.; funding acquisition, J.-S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Science and Technology Council, Taiwan, grant number NSTC 112-2221-E-218-017.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All datasets in this work can be accessed through the link provided in Section 2.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Sun, J.-H.; Ji, T.-T.; Zhang, S.-B.; Yang, J.-K.; Ji, G.-R. Research on the Hand Gesture Recognition Based on Deep Learning. In Proceedings of the 2018 12th International Symposium on Antennas, Propagation and EM Theory (ISAPE), Hangzhou, China, 3–6 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–4.
2. Oudah, M.; Al-Naji, A.; Chahl, J. Hand Gesture Recognition Based on Computer Vision: A Review of Techniques. *J. Imaging* **2020**, *6*, 73. [CrossRef]
3. Muthu Mariappan, H.; Gomathi, V. Real-Time Recognition of Indian Sign Language. In Proceedings of the ICCIDS 2019—2nd International Conference on Computational Intelligence in Data Science, Chennai, India, 21–23 February 2019; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Department of Computer Science and Engineering, National Engineering College: Kovilpatti, India, 2019.
4. Makarov, I.; Veldyaykin, N.; Chertkov, M.; Pokoev, A. Russian Sign Language Dactyl Recognition. In Proceedings of the 2019 42nd International Conference on Telecommunications and Signal Processing, TSP 2019, Budapest, Hungary, 1–3 July 2019; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; National Research University Higher School of Economics: Moscow, Russia, 2019; pp. 726–729.
5. Žemgulys, J.; Raudonis, V.; Maskeliūnas, R.; Damaševičius, R. Recognition of Basketball Referee Signals from Real-Time Videos. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 979–991. [CrossRef]
6. Kong, L.; Huang, D.; Qin, J.; Wang, Y. A Joint Framework for Athlete Tracking and Action Recognition in Sports Videos. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 532–548. [CrossRef]

7. Carfi, A.; Motolese, C.; Bruno, B.; Mastrogiovanni, F. Online Human Gesture Recognition Using Recurrent Neural Networks and Wearable Sensors. In Proceedings of the 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Nanjing, China, 27–31 August 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 188–195.
8. Park, S.; Kim, D. Study on 3D Action Recognition Based on Deep Neural Network. In Proceedings of the 2019 International Conference on Electronics, Information, and Communication (ICEIC), Auckland, New Zealand, 22–25 January 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–3.
9. Badave, H.; Kuber, M. Head Pose Estimation Based Robust Multicamera Face Recognition. In Proceedings of the 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), Coimbatore, India, 25–27 March 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 492–495.
10. Liaqat, S.; Dashtipour, K.; Arshad, K.; Assaleh, K.; Ramzan, N. A Hybrid Posture Detection Framework: Integrating Machine Learning and Deep Neural Networks. *IEEE Sens. J.* **2021**, *21*, 9515–9522. [CrossRef]
11. Wang, Y.; Liu, J. A Self-Developed Smart Wristband to Monitor Exercise Intensity and Safety in Physical Education Class. In Proceedings of the Proceedings—2019 8th International Conference of Educational Innovation through Technology, EITT 2019, Biloxi, MS, USA, 27–31 October 2019; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2019; pp. 160–164.
12. Caviedes, J.E.; Li, B.; Jammula, V.C. Wearable Sensor Array Design for Spine Posture Monitoring during Exercise Incorporating Biofeedback. *IEEE Trans. Biomed. Eng.* **2020**, *67*, 2828–2838. [CrossRef]
13. Arathi, P.N.; Arthika, S.; Ponmithra, S.; Srinivasan, K.; Rukkumani, V. Gesture Based Home Automation System. In Proceedings of the 2017 International Conference On Nextgen Electronic Technologies: Silicon to Software, ICNETS2 2017, Chennai, India, 23–25 March 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Department of Electronics and Instrumentation Engineering, Sri Ramakrishna Engineering College: Coimbatore, India, 2017; pp. 198–201.
14. Abraham, L.; Urru, A.; Normani, N.; Wilk, M.P.; Walsh, M.; O'flynn, B. Hand Tracking and Gesture Recognition Using Lensless Smart Sensors. *Sensors* **2018**, *18*, 2834. [CrossRef]
15. Nascimento, T.H.; Soares, F.A.A.M.N.; Nascimento, H.A.D.; Vieira, M.A.; Carvalho, T.P.; de Miranda, W.F. Netflix Control Method Using Smartwatches and Continuous Gesture Recognition. In Proceedings of the 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), Edmonton, AB, Canada, 5–8 May 2019; IEEE: Piscataway, NJ, USA, 2019; pp. 1–4.
16. Ahmed, S.; Cho, S.H. Hand Gesture Recognition Using an IR-UWB Radar with an Inception Module-Based Classifier. *Sensors* **2020**, *20*, 564. [CrossRef]
17. Lee, C.; Kim, J.; Cho, S.; Kim, J.; Yoo, J.; Kwon, S. Development of Real-Time Hand Gesture Recognition for Tabletop Holographic Display Interaction Using Azure Kinect. *Sensors* **2020**, *20*, 4566. [CrossRef] [PubMed]
18. Ekneling, S.; Sonestedt, T.; Georgiadis, A.; Yousefi, S.; Chana, J. Magestro: Gamification of the Data Collection Process for Development of the Hand Gesture Recognition Technology. In Proceedings of the Adjunct Proceedings—2018 IEEE International Symposium on Mixed and Augmented Reality, ISMAR-Adjunct 2018, Munich, Germany, 16–20 October 2018; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Department of Computer and Systems Sciences, Stockholm University: Stockholm, Sweden, 2018; pp. 417–418.
19. Bai, Z.; Wang, L.; Zhou, S.; Cao, Y.; Liu, Y.; Zhang, J. Fast Recognition Method of Football Robot's Graphics from the VR Perspective. *IEEE Access* **2020**, *8*, 161472–161479. [CrossRef]
20. Nooruddin, N.; Dembani, R.; Maitlo, N. HGR: Hand-Gesture-Recognition Based Text Input Method for AR/VR Wearable Devices. In Proceedings of the Conference Proceedings—IEEE International Conference on Systems, Man and Cybernetics, Toronto, ON, Canada, 11–14 October 2020; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2020; pp. 744–751.
21. Mezari, A.; Maglogiannis, I. An Easily Customized Gesture Recognizer for Assisted Living Using Commodity Mobile Devices. *J. Healthc. Eng.* **2018**, *2018*, 3180652. [CrossRef]
22. Roberge, A.; Bouchard, B.; Maître, J.; Gaboury, S. Hand Gestures Identification for Fine-Grained Human Activity Recognition in Smart Homes. *Procedia Comput. Sci.* **2022**, *201*, 32–39. [CrossRef]
23. Kaczmarek, W.; Panasiuk, J.; Borys, S.; Banach, P. Industrial Robot Control by Means of Gestures and Voice Commands in Off-Line and On-Line Mode. *Sensors* **2020**, *20*, 6358. [CrossRef] [PubMed]
24. Neto, P.; Simão, M.; Mendes, N.; Safeea, M. Gesture-Based Human-Robot Interaction for Human Assistance in Manufacturing. *Int. J. Adv. Manuf. Technol.* **2019**, *101*, 119–135. [CrossRef]
25. Young, G.; Milne, H.; Griffiths, D.; Padfield, E.; Blenkinsopp, R.; Georgiou, O. Designing Mid-Air Haptic Gesture Controlled User Interfaces for Cars. *Proc. ACM Hum. Comput. Interact* **2020**, *4*, 1–23. [CrossRef]
26. Archived: WHO Timeline—COVID-19. Available online: https://www.who.int/news/item/27-04-2020-who-timeline---covid-19 (accessed on 25 October 2023).
27. Katti, J.; Kulkarni, A.; Pachange, A.; Jadhav, A.; Nikam, P. Contactless Elevator Based on Hand Gestures during COVID-19 like Pandemics. In Proceedings of the 2021 7th International Conference on Advanced Computing and Communication Systems, ICACCS 2021, Coimbatore, India, 19–20 March 2021; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Pimpri Chinchwad College of Engineering: Maharashtra, India, 2021; pp. 672–676.
28. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]

29. Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; Fei-Fei, L. ImageNet: A Large-Scale Hierarchical Image Database. In Proceedings of the 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPR Workshops), Miami, FL, USA, 20–25 June 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 248–255.
30. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556.
31. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the Inception Architecture for Computer Vision. In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 2818–2826.
32. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; IEEE: Piscataway, NJ, USA, 2016; pp. 770–778.
33. Shafiq, M.; Gu, Z. Deep Residual Learning for Image Recognition: A Survey. *Appl. Sci.* **2022**, *12*, 8972. [CrossRef]
34. Khosla, C.; Saini, B.S. Enhancing Performance of Deep Learning Models with Different Data Augmentation Techniques: A Survey. In Proceedings of the International Conference on Intelligent Engineering and Management, ICIEM 2020, London, UK, 17–19 June 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 79–85.
35. Mikolajczyk, A.; Grochowski, M. Data Augmentation for Improving Deep Learning in Image Classification Problem. In Proceedings of the 2018 International Interdisciplinary PhD Workshop (IIPhDW), Swinoujscie, Poland, 9–12 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 117–122.
36. Kaur, P.; Khehra, B.S.; Mavi, E.B.S. Data Augmentation for Object Detection: A Review. In Proceedings of the 2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS), Lansing, MI, USA, 9–11 August 2021; pp. 537–543. [CrossRef]
37. Leevy, J.L.; Khoshgoftaar, T.M.; Bauder, R.A.; Seliya, N. A Survey on Addressing High-Class Imbalance in Big Data. *J. Big Data* **2018**, *5*, 42. [CrossRef]
38. Shukla, P.; Bhowmick, K. To Improve Classification of Imbalanced Datasets. In Proceedings of the 2017 International Conference on Innovations in Information, Embedded and Communication Systems, ICIIECS 2017, Coimbatore, India, 17–18 March 2017; pp. 1–5. [CrossRef]
39. Shorten, C.; Khoshgoftaar, T.M. A Survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60. [CrossRef]
40. Mohamed, N.; Mustafa, M.B.; Jomhari, N. A Review of the Hand Gesture Recognition System: Current Progress and Future Directions. *IEEE Access* **2021**, *9*, 157422–157436. [CrossRef]
41. Lim, K.M.; Tan, A.W.C.; Tan, S.C. A Feature Covariance Matrix with Serial Particle Filter for Isolated Sign Language Recognition. *Expert Syst. Appl.* **2016**, *54*, 208–218. [CrossRef]
42. Farahanipad, F.; Rezaei, M.; Nasr, M.S.; Kamangar, F.; Athitsos, V. A Survey on GAN-Based Data Augmentation for Hand Pose Estimation Problem. *Technologies* **2022**, *10*, 43. [CrossRef]
43. Sharma, S.; Singh, S. Vision-Based Hand Gesture Recognition Using Deep Learning for the Interpretation of Sign Language. *Expert Syst. Appl.* **2021**, *182*, 115657. [CrossRef]
44. Kandel, I.; Castelli, M.; Manzoni, L. Brightness as an Augmentation Technique for Image Classification. *Emerg. Sci. J.* **2022**, *6*, 881–892. [CrossRef]
45. Islam, M.Z.; Hossain, M.S.; Ul Islam, R.; Andersson, K. Static Hand Gesture Recognition Using Convolutional Neural Network with Data Augmentation. In Proceedings of the 2019 Joint 8th International Conference on Informatics, Electronics and Vision, ICIEV 2019 and 3rd International Conference on Imaging, Vision and Pattern Recognition, icIVPR 2019 with International Conference on Activity and Behavior Computing, ABC 2019, Spokane, WA, USA, 30 May–2 June 2019; pp. 324–329. [CrossRef]
46. Bousbai, K.; Merah, M. Hand Gesture Recognition Using Capabilities of Capsule Network and Data Augmentation. In Proceedings of the 2022 7th International Conference on Image and Signal Processing and Their Applications, ISPA 2022—Proceedings, Mostaganem, Algeria, 8–9 May 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Mostaganem University, Elctronics and Embededd Systems, Department of Electrical Engineering: Mostaganem, Algeria, 2022.
47. Alani, A.A.; Cosma, G.; Taherkhani, A.; McGinnity, T.M. Hand Gesture Recognition Using an Adapted Convolutional Neural Network with Data Augmentation. In Proceedings of the 2018 4th International Conference on Information Management (ICIM), Oxford, UK, 25–27 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 5–12.
48. Zhou, W.; Chen, K. A Lightweight Hand Gesture Recognition in Complex Backgrounds. *Displays* **2022**, *74*, 102226. [CrossRef]
49. Luo, Y.; Cui, G.; Li, D. An Improved Gesture Segmentation Method for Gesture Recognition Based on CNN and YCbCr. *J. Electr. Comput. Eng.* **2021**, *2021*, 1783246. [CrossRef]
50. Fadillah Rahmat, R.; Chairunnisa, T.; Gunawan, D.; Fermi Pasha, M.; Budiarto, R. Hand gestures recognition with improved skin color segmentation in human-computer interaction applications. *J. Theor. Appl. Inf. Technol.* **2019**, *97*, 727–739.
51. Yao, Y.; Li, C.T. Hand Gesture Recognition and Spotting in Uncontrolled Environments Based on Classifier Weighting. In Proceedings of the International Conference on Image Processing, ICIP 2015, Quebec City, QC, Canada, 27–30 September 2015; pp. 3082–3086. [CrossRef]
52. Yang, F.; Shi, H. Research on Static Hand Gesture Recognition Technology for Human Computer Interaction System. In Proceedings of the 2016 International Conference on Intelligent Transportation, Big Data and Smart City, ICITBS 2016, Changsha, China, 17–18 December 2016; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2017; pp. 459–463.

53. Vasiljevic, I.; Chakrabarti, A.; Shakhnarovich, G. Examining the Impact of Blur on Recognition by Convolutional Networks. *arXiv* **2016**, arXiv:1611.05760.
54. Salunke, T.P.; Bharkad, S.D. Power Point Control Using Hand Gesture Recognition Based on HOG Feature Extraction and K-Nn Classification. In Proceedings of the International Conference on Computing Methodologies and Communication, ICCMC 2017, Erode, India, 18–19 July 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Dept. of EandTC Engineering, Government College of Engineering: Aurangabad, India, 2018; pp. 1151–1155.
55. Chanu, O.R.; Pillai, A.; Sinha, S.; Das, P. Comparative Study for Vision Based and Data Based Hand Gesture Recognition Technique. In Proceedings of the ICCT 2017—International Conference on Intelligent Communication and Computational Techniques, Jaipur, India, 22–23 December 2017; pp. 26–31. [CrossRef]
56. Flores, C.J.L.; Cutipa, A.E.G.; Enciso, R.L. Application of Convolutional Neural Networks for Static Hand Gestures Recognition under Different Invariant Features. In Proceedings of the 2017 IEEE 24th International Congress on Electronics, Electrical Engineering and Computing, INTERCON 2017, Cusco, Peru, 15–18 August 2017; pp. 5–8. [CrossRef]
57. Bao, P.; Maqueda, A.I.; Del-Blanco, C.R.; Garciá, N. Tiny Hand Gesture Recognition without Localization via a Deep Convolutional Network. *IEEE Trans. Consum. Electron.* **2017**, *63*, 251–257. [CrossRef]
58. Qiao, Y.; Feng, Z.; Zhou, X.; Yang, X. Principle Component Analysis Based Hand Gesture Recognition for Android Phone Using Area Features. In Proceedings of the 2017 2nd International Conference on Multimedia and Image Processing, ICMIP 2017, Wuhan, China, 17–19 March 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; School of Information Science and Engineering, University of Jinan: Jinan, China, 2017; pp. 108–112.
59. Kadethankar, A.A.; Joshi, A.D. Dynamic Hand Gesture Recognition Using Kinect. In Proceedings of the 2017 Innovations in Power and Advanced Computing Technologies, i-PACT 2017, Vellore, India, 21–22 April 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Electronics and Telecommunication, Shri Guru Gobind Singhji Institute of Engg. and Tech.: Maharashtra, India, 2017; pp. 1–3.
60. Abdul-Rashid, H.M.; Kiran, L.; Mirrani, M.D.; Maraaj, M.N. CMSWVHG-Control MS Windows via Hand Gesture. In Proceedings of the Proceedings of 2017 International Multi-Topic Conference, INMIC 2017, Lahore, Pakistan, 24–26 November 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; National University of Computer and Emerging Sciences, FAST-NU: Islamabad, Pakistan, 2018; pp. 1–7.
61. Zhang, Y.; Cao, C.; Cheng, J.; Lu, H. EgoGesture: A New Dataset and Benchmark for Egocentric Hand Gesture Recognition. *IEEE Trans. Multimed.* **2018**, *20*, 1038–1050. [CrossRef]
62. He, Y.; Yang, J.; Shao, Z.; Li, Y. Salient Feature Point Selection for Real Time RGB-D Hand Gesture Recognition. In Proceedings of the 2017 IEEE International Conference on Real-Time Computing and Robotics, RCAR 2017, Okinawa, Japan, 14–18 July 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; School of Urban Rail Transportation, Soochow University: Suzhou, China, 2017; pp. 103–108.
63. Sachara, F.; Kopinski, T.; Gepperth, A.; Handmann, U. Free-Hand Gesture Recognition with 3D-CNNs for in-Car Infotainment Control in Real-Time. In Proceedings of the IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, Yokohama, Japan, 16–19 October 2017; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA; Computer Science Institute, Hochschule Ruhr West: Bottrop, Germany, 2018; pp. 959–964.
64. Ahmed, W.; Chanda, K.; Mitra, S. Vision Based Hand Gesture Recognition Using Dynamic Time Warping for Indian Sign Language. In Proceedings of the 2016 International Conference on Information Science, ICIS 2016, Kochi, India, 12–13 August 2016; pp. 120–125. [CrossRef]
65. Kane, L.; Khanna, P. Vision-Based Mid-Air Unistroke Character Input Using Polar Signatures. *IEEE Trans. Hum. Mach. Syst.* **2017**, *47*, 1077–1088. [CrossRef]
66. Raditya, C.; Rizky, M.; Mayranio, S.; Soewito, B. The Effectivity of Color for Chroma-Key Techniques. *Procedia Comput. Sci.* **2021**, *179*, 281–288. [CrossRef]
67. Zhi, J. An Alternative Green Screen Keying Method for Film Visual Effects. *Int. J. Multimed. Its Appl.* **2015**, *7*, 1–12. [CrossRef]
68. Sengupta, S.; Jayaram, V.; Curless, B.; Seitz, S.; Kemelmacher-Shlizerman, I. Background Matting: The World Is Your Green Screen. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 2288–2297. [CrossRef]
69. Barczak, A.L.C.; Reyes, N.H.; Abastillas, M.; Piccio, A.; Susnjak, T. *A New 2D Static Hand Gesture Colour Image Dataset for ASL Gestures*; Massey University: Palmerston North, New Zealand, 2011; Volume 15, Available online: https://mro.massey.ac.nz/server/api/core/bitstreams/09187662-5ebe-4563-8515-3d7e5e1d2a33/content (accessed on 2 February 2023).
70. Marcel, S. Hand Posture Recognition in a Body-Face Centered Space. In *CHI'99 Extended Abstracts on Human Factors in Computing Systems*; Association for Computing Machinery: New York, NY, USA, 1999.
71. Pisharady, P.K.; Vadakkepat, P.; Loh, A.P. Attention Based Detection and Recognition of Hand Postures against Complex Backgrounds. *Int. J. Comput. Vis.* **2013**, *101*, 403–419. [CrossRef]
72. Güler, O.; Yücedağ, İ. Hand Gesture Recognition from 2D Images by Using Convolutional Capsule Neural Networks. *Arab. J. Sci. Eng.* **2022**, *47*, 1211–1225. [CrossRef]
73. Alzubaidi, L.; Zhang, J.; Humaidi, A.J.; Al-Dujaili, A.; Duan, Y.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.A.; Al-Amidie, M.; Farhan, L. Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *J. Big Data* **2021**, *8*, 53. [CrossRef] [PubMed]

74. Agarap, A.F. Deep Learning Using Rectified Linear Units (ReLU). *arXiv* **2018**, arXiv:1803.08375.
75. Subburaj, S.; Murugavalli, S. Survey on Sign Language Recognition in Context of Vision-Based and Deep Learning. *Meas. Sens.* **2022**, *23*, 100385. [CrossRef]
76. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
77. Poojary, R.; Pai, A. Comparative Study of Model Optimization Techniques in Fine-Tuned CNN Models. In Proceedings of the 2019 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2019, Ras Al Khaimah, United Arab Emirates, 19–21 November 2019; pp. 2–5. [CrossRef]
78. Ozdemir, M.A.; Kisa, D.H.; Guren, O.; Onan, A.; Akan, A. EMG Based Hand Gesture Recognition Using Deep Learning. In Proceedings of the TIPTEKNO 2020—Tip Teknolojileri Kongresi—2020 Medical Technologies Congress, TIPTEKNO 2020, Antalya, Turkey, 19–20 November 2020. [CrossRef]
79. Theckedath, D.; Sedamkar, R.R. Detecting Affect States Using VGG16, ResNet50 and SE-ResNet50 Networks. *SN Comput. Sci.* **2020**, *1*, 79. [CrossRef]
80. Esi Nyarko, B.N.; Bin, W.; Zhou, J.; Agordzo, G.K.; Odoom, J.; Koukoyi, E. Comparative Analysis of AlexNet, Resnet-50, and Inception-V3 Models on Masked Face Recognition. In Proceedings of the 2022 IEEE World AI IoT Congress, AIIoT 2022, Seattle, WA, USA, 6–9 June 2022; Institute of Electrical and Electronics Engineers Inc.: Piscataway, NJ, USA, 2022; pp. 337–343.
81. Hossain, B.; Sazzad, S.M.H.; Islam, M.; Akhtar, N.; Aziz, A.; Attique, M.; Tariq, U.; Nam, Y.; Nazir, M.; Jeong, C.W.; et al. An Ensemble of Optimal Deep Learning Features for Brain Tumor Classification. In Proceedings of the 2019 International Conference on Electrical and Computing Technologies and Applications, ICECTA 2019, Ras Al Khaimah, United Arab Emirates, 19–21 November 2019; Volume 211, pp. 2–5. [CrossRef]
82. Muslikhin, M.; Horng, J.R.; Yang, S.Y.; Wang, M.S.; Awaluddin, B.A. An Artificial Intelligence of Things-based Picking Algorithm for Online Shop in the Society 5.0's Context. *Sensors* **2021**, *21*, 2813. [CrossRef] [PubMed]
83. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv* **2017**, arXiv:1412.6980.