

Hand Sign Virtual Reality Data Processing Using Padding Technique

Teja E. E. Tju¹, Julaiha P. Anggraini¹, and Muhammad U. Shalih¹

¹Faculty of Information Technology, Universitas Budi Luhur, Jakarta Selatan, Indonesia

Corresponding author: Teja E. E. Tju (e-mail: teja.endraengtju@budiluhur.ac.id).

ABSTRACT This study focuses on addressing the challenges of processing hand sign data in Virtual Reality environments, particularly the variability in data length during gesture recording. To optimize machine learning models for gesture recognition, various padding techniques were implemented. The data was gathered using the Meta Quest 2 device, consisting of 1,000 samples representing 10 American Sign Language hand sign movements. The research applied different padding techniques, including pre- and post-zero padding as well as replication padding, to standardize sequence lengths. Long Short-Term Memory networks were utilized for modeling, with the data split into 80% for training and 20% for validation. An additional 100 unseen samples were used for testing. Among the techniques, pre-replication padding produced the best results in terms of accuracy, precision, recall, and F1 score on the test dataset. Both pre- and post-zero padding also demonstrated strong performance but were outperformed by replication padding. This study highlights the importance of padding techniques in optimizing the accuracy and generalizability of machine learning models for hand sign recognition in Virtual Reality. The findings offer valuable insights for developing more robust and efficient gesture recognition systems in interactive Virtual Reality environments, enhancing user experiences and system reliability. Future work could explore extending these techniques to other Virtual Reality interactions.

KEYWORDS Recurrent Neural Networks (RNNs), Sequential Data, Signal Processing.

I. INTRODUCTION

The development of Virtual Reality (VR) technology has opened new opportunities in various fields [1], [2], including education [3], [4], [5], rehabilitation [6], [7], [8], and the development of applications for individuals with special needs [9], [10]. One promising application is hand sign interpretation for communication with individuals who have speech disabilities [11], [12], [13]. However, hand-sign data processing in a VR environment faces several challenges. Specifically, the processing of hand sign data from VR devices encounters difficulties, particularly concerning the variation in data length generated during the recording process. Therefore, in-depth research is needed to optimize the padding data processing method to enhance hand sign interpretation accuracy and efficiency.

Previous research has attempted to use VR technology to assist individuals with speech disabilities through hand sign interpretation. Some studies have used image datasets with American Sign Language (ASL) [13], [14], [15] and Malaysian Sign Language (MSL) [11]. Other research has utilized triboelectric gloves that produce voltage graph datasets [16]. Padding techniques are generally applied in studies with sequential or graphical datasets, such as sign

language recognition [17], speech emotion recognition [18], and padding modules [19] with neural network modeling, as well as traffic flow prediction using Long Short-Term Memory (LSTM) models [20].

The novelty of this research lies in its application of padding techniques to VR hand sign data, specifically addressing the variation in data length generated during the data collection process using sequential primary data recorded directly from VR devices. While our previous similar studies [21] have focused solely on post-zero padding with 28 parameters, requiring more complex RNN models, this research employs 22 parameters, allowing for a simpler model architecture without sacrificing effectiveness. This approach highlights the trade-off between model complexity and parameter count, demonstrating that a streamlined model can still achieve efficient performance. The innovative use of padding techniques in the context of VR hand sign data, which is relatively new, offers a targeted solution for enhancing VR applications and supporting individuals with speech disabilities.

II. METHODS

The research process is divided into four main stages, as illustrated in Figure 1, Data Collection, Implementing Padding Technique, Machine Learning Modeling, and Evaluation and Testing. These stages ensure a comprehensive approach to addressing the challenges of hand sign interpretation in a VR environment.

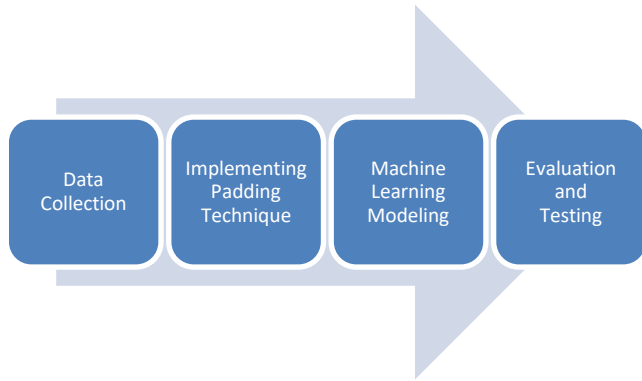


Figure 1. Overview of research stages.

A. DATA COLLECTION

Primary data collection was conducted using the VR device Meta Quest 2 [22], as shown in Figure 2. Ten types of hand sign movements were collected based on ASL [23], [24], [25], [26] and selected for their ease of use and compatibility with VR devices. These signs, illustrated in Figure 3, were chosen because they are common and straightforward, ensuring the VR system can accurately capture them. Each sign was recorded with 100 samples, providing sufficient data for analysis.



Figure 2. Meta Quest 2: Immersive, all-in-one VR device [27].

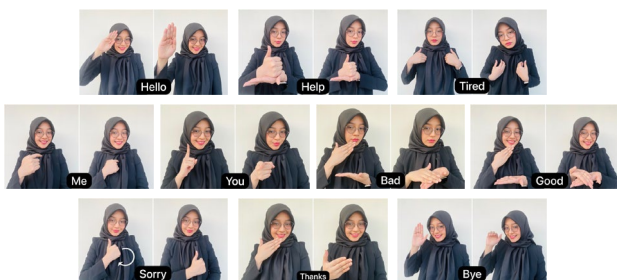


Figure 3. 10 Hand sign movements.

Data collection was performed using an application developed with Unity Editor [28], [29], as shown in Figure 4. Each data recorded consists of 11 parameters each from the left and right hand, including trigger touch, trigger pressed, grip pressed, thumb touch, position (X, Y, Z), and quaternion (W, X, Y, Z).

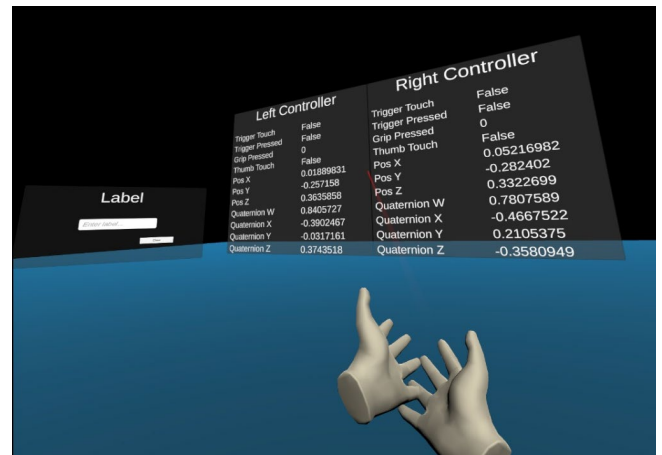


Figure 4. VR data recording application.

B. IMPLEMENTING PADDING TECHNIQUES

Several padding methods were studied to understand how to implement them in the context of VR hand gesture data. Commonly used padding techniques, such as zero and replication, are applied to balance the data length. Specifically, variations like pre-zero, post-zero, pre- and post-zero, pre-replication, post-replication, and pre- and post-replication padding are explored. Figure 5 illustrates the explanation of these padding techniques.

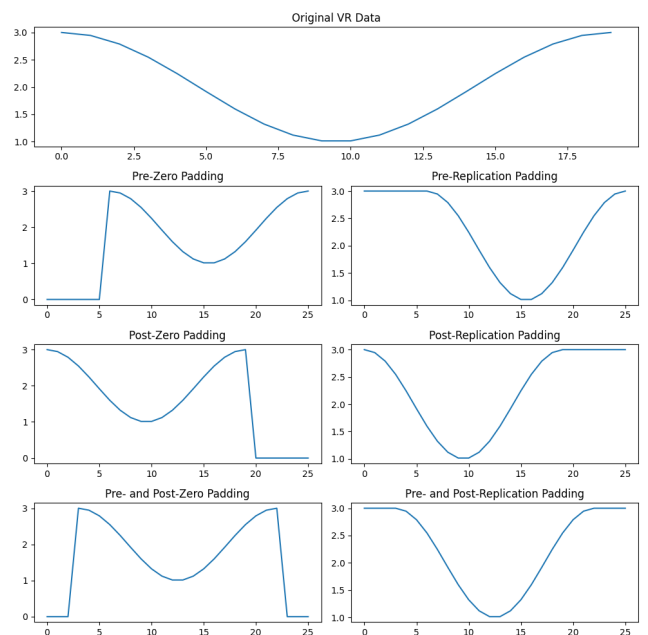


Figure 5. Comparison of padding techniques.

The selected padding methods were applied to the collected data and compared to identify the most effective approach for managing data length variation. This process ensured consistency and integrity in the machine learning workflow. The goal of evaluating different padding techniques was to improve the accuracy and efficiency of machine-learning models for interpreting VR hand gestures.

C. MACHINE LEARNING MODELING

After the data has been processed using padding techniques, machine learning models are developed and trained. The data is divided into training data (80%) and validation data (20%). Recurrent Neural Networks (RNNs) with Long Short-Term Memory (LSTM) cells are utilized for modeling. LSTMs are a specific type of RNN that effectively handles sequential data and captures long-range dependencies while mitigating the vanishing gradient problem common in traditional RNNs [30], [31].

The model is trained on the training dataset and its performance is evaluated using the validation dataset. Further testing is conducted with new, unseen data to assess the model’s effectiveness in real-world scenarios.

D. EVALUATION AND TESTING

Thoroughly evaluating the performance of the trained model, several evaluation metrics were employed, including accuracy, precision, recall, F1 score, and confusion matrix analysis [32]. These metrics provided a comprehensive view of the model’s performance, identifying areas where it excelled and where further improvements were needed.

To further validate the robustness of the model, an additional 100 new data samples were collected. These samples, which the model had not previously encountered, were used to test its performance on unseen data. This step was crucial in determining the model’s real-world applicability and ensuring that it could generalize effectively beyond the initial dataset.

The evaluation process, therefore, not only confirmed the model's effectiveness but also guided subsequent refinement and optimization efforts, ensuring a reliable and efficient solution for interpreting VR hand gestures.

III. RESULT AND DISCUSSION

The data collection phase successfully yielded 1,000 samples, evenly distributed across 10 selected ASL hand signs (https://github.com/umaruta4/SignLanguage_MTC_Data/tree/main/new_american_sign_language). Each sign contributed an equal number of samples, ensuring a balanced dataset for further analysis. Figure 6 presents an example from the collected data, highlighting the 11 parameters that define each hand sign.

The overall graphs shown in Figure 6 illustrate the recordings of a specific hand sign movement, the horizontal axis of these graphs denotes the n-th Unity sampling, while the vertical axis values correspond to various sensor readings [28]. The data is organized into graphs labeled from Sensor 0 to Sensor 21, with a detailed explanation of the 11 parameters provided in Table 1.

The application of various padding strategies played a significant role in addressing the challenge of varying data lengths in VR hand gesture datasets. The variations of zero and replication padding techniques were systematically applied to the dataset. The results of these padding implementations, as depicted in Figure 7, show how the raw data in Figure 6 was transformed into a consistent format across all samples. This uniformity was essential in preserving the data's structural integrity and ensuring that the machine learning algorithms could process the data without being influenced by inconsistencies in sequence length.

TABLE I
SENSOR DATA PARAMETERS

Left Hand	Right Hand	Parameter	Vertical Axis Value
Sensor 0	Sensor 11	Trigger Touch	Boolean: 0 or 1
Sensor 1	Sensor 12	Trigger Pressed	Boolean: 0 or 1
Sensor 2	Sensor 13	Grip Pressed	Boolean: 0 or 1
Sensor 3	Sensor 14	Thumb Touch	Boolean: 0 or 1
Sensor 4	Sensor 15	Position X	Meter (m)
Sensor 5	Sensor 16	Position Y	Meter (m)
Sensor 6	Sensor 17	Position Z	Meter (m)
Sensor 7	Sensor 18	Quaternion W	Scalar
Sensor 8	Sensor 19	Quaternion X	Vector
Sensor 9	Sensor 20	Quaternion Y	Vector
Sensor 10	Sensor 21	Quaternion Z	Vector

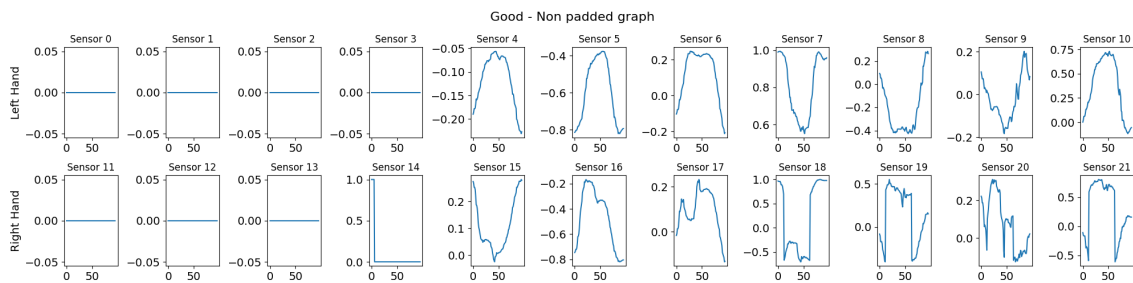


Figure 6. Example of ASL “Good” data with 22 parameters.

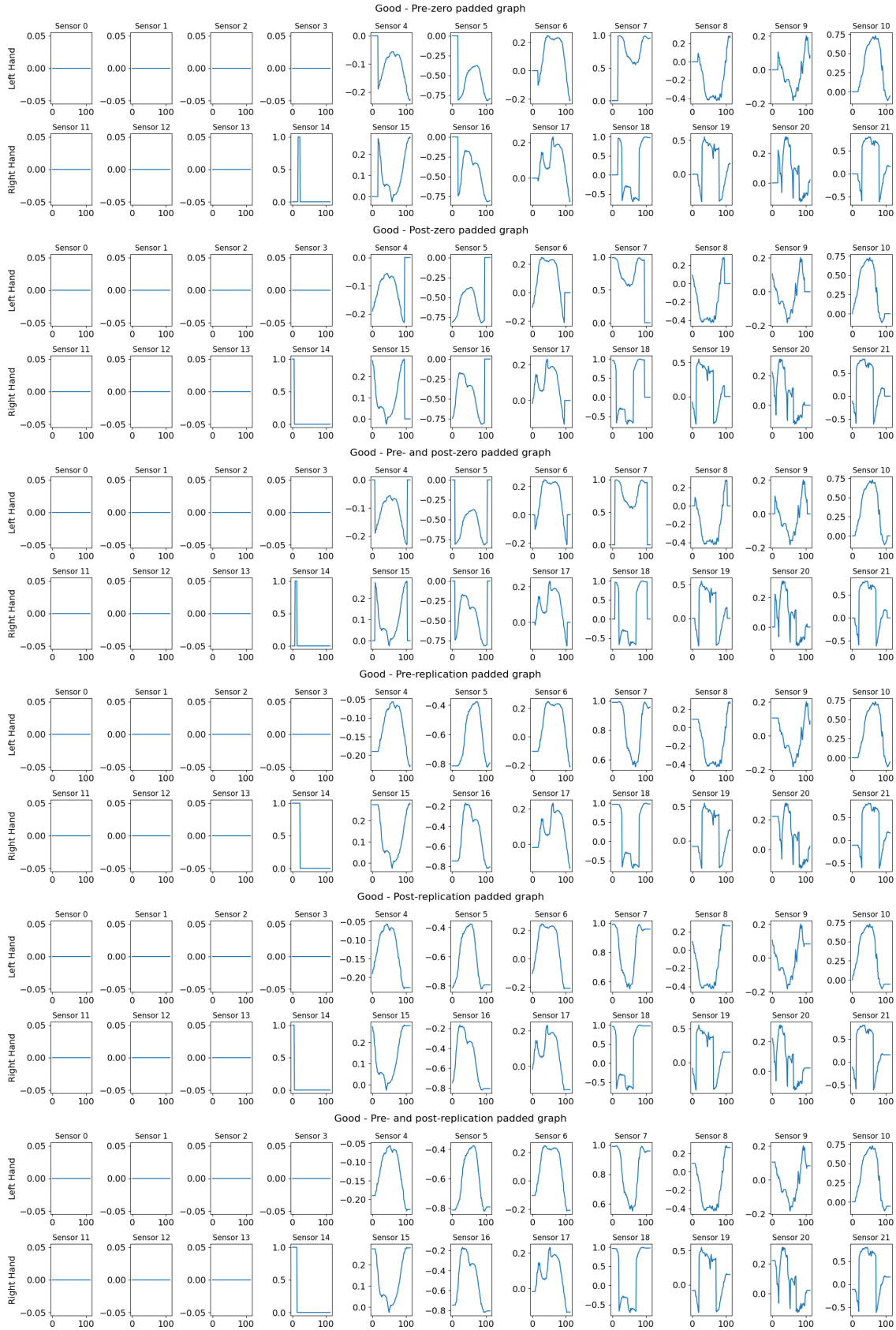


Figure 7. Examples of VR hand gesture data after applying various padding techniques.

The LSTM-based neural network model, shown in Figure 8, was designed to process the sequential hand gesture data captured in the VR environment. After applying padding techniques, the sequence length was standardized to match the maximum Unity sampling length, which in this case was set to 113 time steps. Each time step in the sequence contains 22 feature dimensions, corresponding to the 22 parameters recorded from both hands during gesture performance.

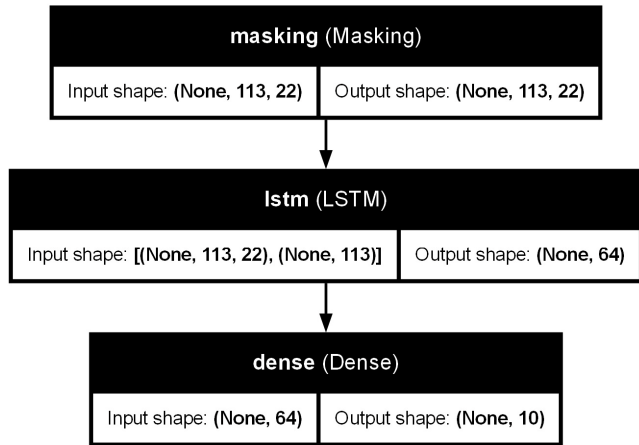


Figure 8. LSTM-based Neural Network model for VR dataset.

The model architecture begins with a **masking layer** that handles padded values by ignoring them while learning, ensuring that the model only processes relevant data. This layer maintains the input shape of (None, 113, 22), where 113 represents the standardized sequence length and 22 denotes the feature dimensions. The None indicates a variable batch size. The model's core is the LSTM layer, specifically designed to capture the temporal dependencies in the sequential gesture data. The input shape for this layer remains (None, 113, 22), and an additional input (None, 113) represents the mask applied to the sequence. It outputs a

reduced representation with 64 features, highlighting the most significant aspects of the data across the 113 time steps. Finally, the **dense layer** aggregates the information extracted by the LSTM layer, outputting a 10-dimensional vector, where each dimension corresponds to the 10 different hand sign movements during training. This structure ensures the model can effectively classify the input sequences into the correct hand gesture categories.

The LSTM-based model was trained on 800 VR hand gesture samples, with 200 samples reserved for validation. Figures 9 and 10 show the model's performance using different padding techniques.

For the **zero padding technique** (Figure 9), the model demonstrated alignment between training and validation accuracy, indicating effective learning with minimal overfitting. The final accuracy confirmed the model's capability to interpret zero-padded data consistently. **Pre-zero padding** achieved 0.58 validation accuracy, indicating baseline effectiveness but challenges in maintaining data integrity. **Post-zero padding** yielded 0.55 accuracy, with minimal impact from the padding position but some inconsistency in data representation. **Pre- and post-zero padding** was the top-performing zero padding method, with 0.99 accuracy, providing balanced data representation.

Replication padding (Figure 10) also produced promising results, with accuracy and loss curves reflecting consistent learning. This method allowed the model to generalize well from the training data, highlighting the importance of selecting appropriate padding techniques. **Pre-replication padding** excelled with 0.97 accuracy, effectively preserving sequence structure for better model learning. **Post-replication padding** reached 0.73 accuracy, performing better than zero padding but less effectively than Pre-replication Padding. **Pre- and post-replication padding** showed strong results with 0.88 accuracy, demonstrating robustness in maintaining temporal structure.

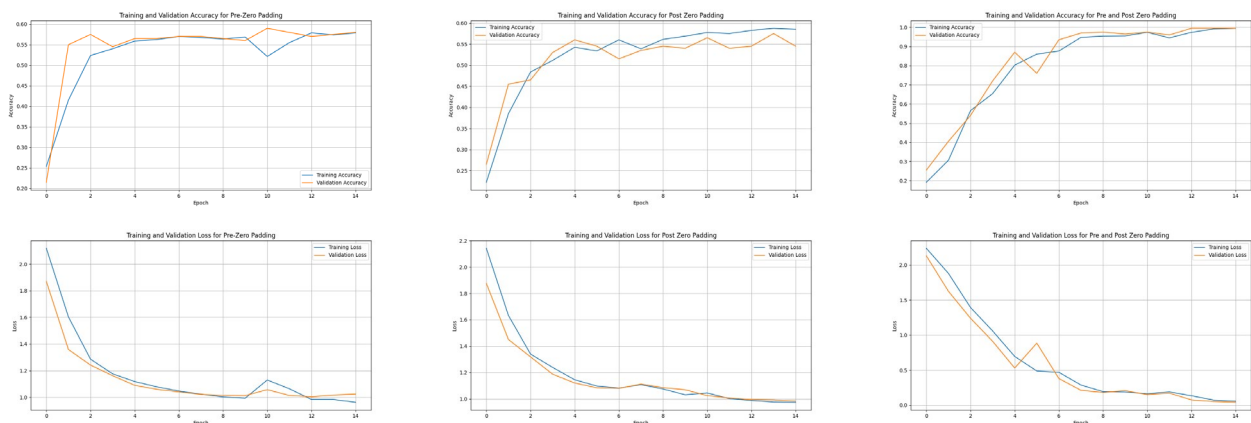


Figure 9. Training and validation accuracy and loss with zero padding.

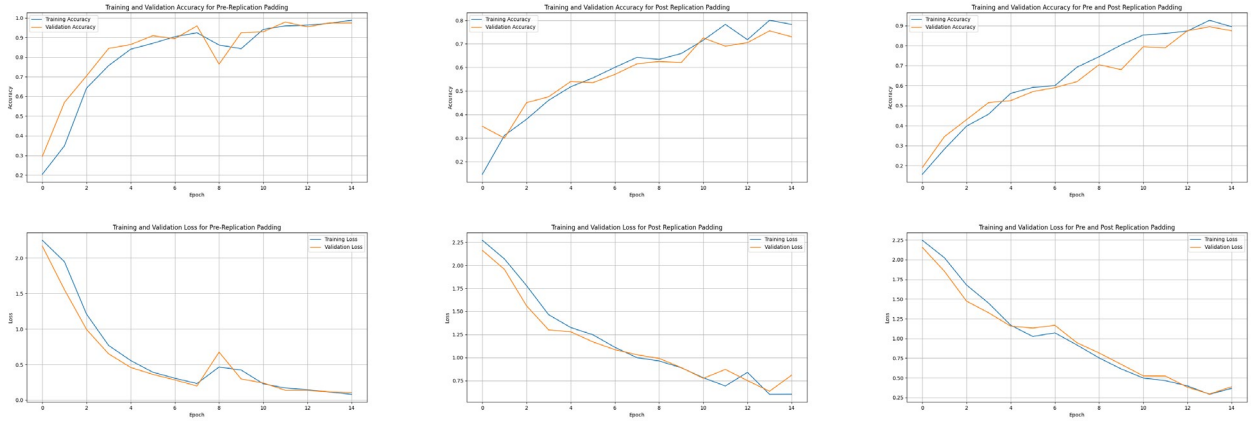


Figure 10. Training and validation accuracy and loss with replication padding.

The trained LSTM-based model was comprehensively evaluated using metrics like accuracy, precision, recall, F1 score, and confusion matrix analysis. These metrics provided an overall assessment of the model's performance, helping to identify its general effectiveness in interpreting VR hand gestures.

Figure 11 shows the confusion matrix for the validation dataset, illustrating how well the model predicted each hand sign after training. The matrix indicates strong performance in some categories but also reveals specific hand signs where the model's predictions were less accurate, suggesting potential areas for further refinement.

An additional 100 unseen data samples were tested to assess the model's robustness. Figure 12 presents the confusion matrix for this test dataset, reflecting the model's ability to generalize to new data. These matrices help to understand the model's performance across various categories.

Table 2 summarizes key performance metrics, including accuracy, precision, recall, and F1 score, for both the validation and test datasets. The table also includes an overall ranking of the padding techniques based on these metrics, offering insights into which methods were most effective in ensuring accurate and consistent model performance.

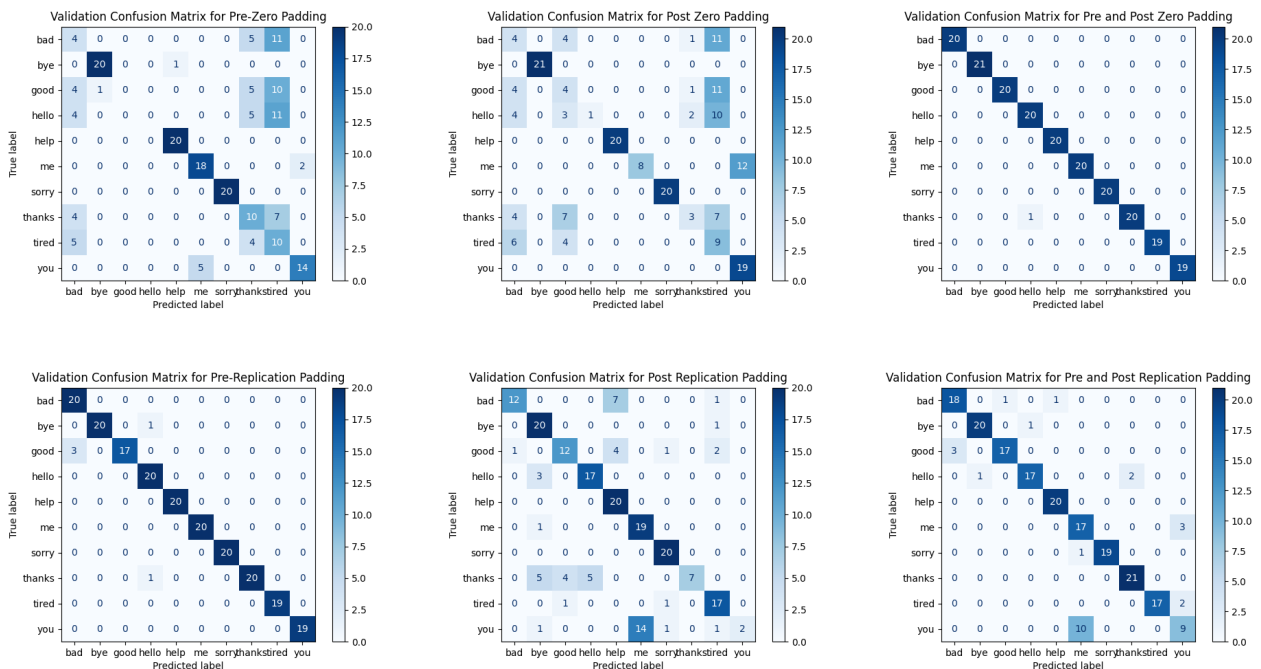


Figure 11. Confusion Matrix for validation dataset.

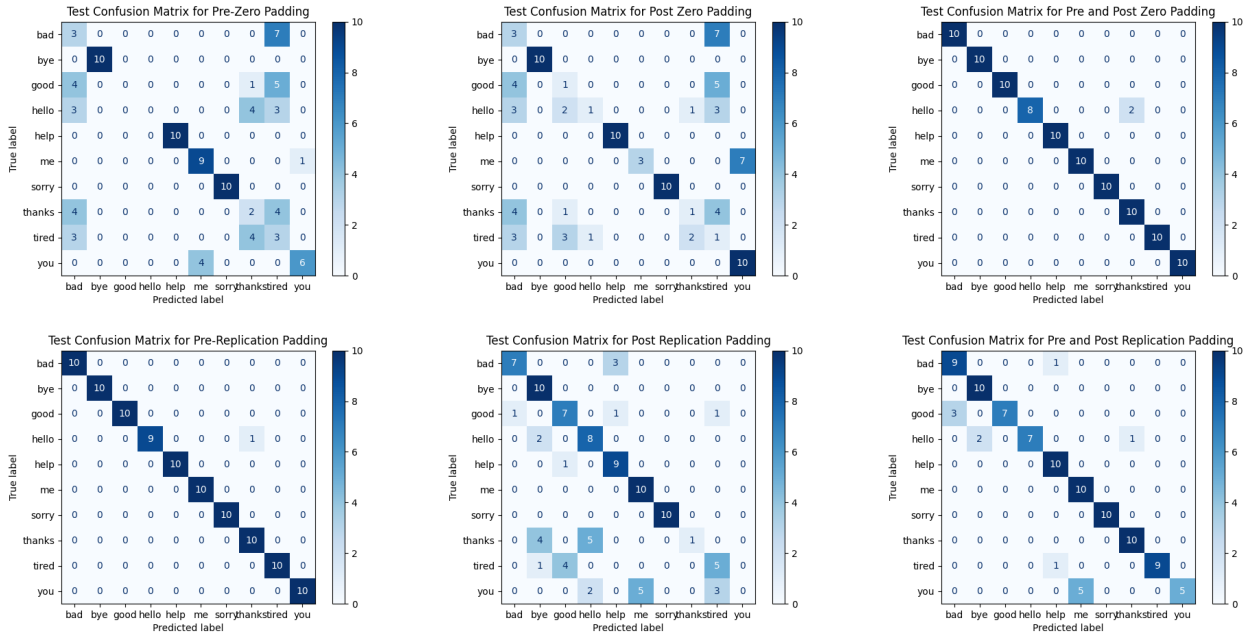


Figure 12. Confusion Matrix for test dataset (100 new data).

TABLE II
AGGREGATE PERFORMANCE METRICS AND RANKING OF PADDING TECHNIQUES

Padding Technique	Accuracy		Precision		Recall		F1 score		Overall Rank	
	Val.	Test	Val.	Test	Val.	Test	Val.	Test	Val.	Test
Pre-Replication Padding	0.97	0.99	0.98	0.99	0.97	0.99	0.98	0.99	2	1
Pre- and Post-Zero Padding	0.99	0.98	1.00	0.98	0.99	0.98	0.99	0.98	1	2
Pre- and Post-Replication Padding	0.88	0.87	0.88	0.90	0.88	0.87	0.87	0.86	3	3
Post-Replication Padding	0.73	0.67	0.79	0.65	0.73	0.67	0.69	0.61	4	4
Pre-Zero Padding	0.58	0.53	0.53	0.50	0.58	0.53	0.55	0.51	6	5
Post-Zero Padding	0.55	0.50	0.66	0.57	0.55	0.50	0.53	0.49	5	6

The analysis of different padding techniques reveals notable variations in performance metrics, including Accuracy, Precision, Recall, and F1 Score, for both validation and test datasets. These metrics provide a comprehensive view of how each padding technique impacts model performance.

Pre-replication padding shows strong performance across all metrics, achieving high scores in both validation and test datasets. It ranks second in validation and first in the test dataset, indicating its robust ability to generalize and maintain a well-balanced model. The consistently high accuracy, precision, recall, and F1 score suggest that this padding technique minimizes misclassifications effectively and performs reliably across different data splits.

In comparison, **pre- and post-zero padding** achieves perfect precision and high recall in the validation dataset, reflecting its strong performance in identifying positive cases within this controlled environment. However, the technique’s performance slightly drops in the test dataset. This decline may be due to the inherent differences between the validation and test data distributions, which can impact how well the model generalizes to new data. Despite this drop, it remains

highly effective and ranks first in validation and second in the test dataset.

Pre- and post-replication padding ranks third in both datasets, showing stable but not exceptional performance. Although it provides balanced metrics, its scores are lower compared to the top two techniques. This suggests that while it performs reliably, it does not reach the high levels of accuracy and balance achieved by **pre-replication** and **pre- and post-zero padding**.

Post-replication padding ranks fourth, with lower accuracy and F1 score compared to the higher-ranked techniques. This indicates a higher rate of misclassifications and less effective performance overall. The lower metrics suggest that this technique is less capable of managing classification tasks with the same efficiency as the top methods.

Pre-zero and **post-zero padding** exhibit the lowest performance, ranking fifth and sixth, respectively. These techniques show poorer accuracy, precision, and recall, leading to higher misclassification rates. Their lower metrics reflect their limited effectiveness in correctly identifying positive cases and achieving balanced classification results.

Choosing the appropriate padding technique is crucial for optimizing model accuracy and generalization. The top methods, **pre-replication** and **pre- and post-zero padding** offer robust performance and balanced metrics, making them suitable for effective model deployment. Conversely, the lower-ranked techniques highlight areas where model performance could be improved, suggesting their lesser suitability for achieving optimal results.

IV. CONCLUSION

This research highlights the significant influence of padding techniques on the performance of RNN models in interpreting VR hand gesture data. Our findings reveal that selecting an appropriate padding method can lead to substantial improvements in model accuracy, precision, recall, and F1 score, even when utilizing simpler RNN architectures. Specifically, techniques like pre-replication padding and pre- and post-zero padding demonstrate superior effectiveness. Pre-replication padding consistently delivers high performance across all evaluation metrics, maintaining robust accuracy and generalization on both validation and test datasets. Meanwhile, pre- and post-zero padding shows excellent results in the validation phase but exhibits a slight reduction in performance during testing, indicating a potential sensitivity to unseen data.

These results highlight the critical role of selecting appropriate padding techniques to optimize model performance in sequence-based data processing. They demonstrate that even with simpler RNN models, the use of strategic padding can substantially enhance learning efficiency and improve the model's ability to generalize from training data to real-world applications. This emphasizes the need for thoughtful preprocessing choices in the design of sequence models to achieve robust and effective outcomes.

Looking ahead, future research could explore advanced padding strategies to optimize model performance. Investigating the interaction between innovative padding methods and different RNN architectures could unlock opportunities for greater accuracy and efficiency, leading to more effective classification systems in VR and other applications. These padding techniques enhance machine learning models' performance and flexibility in complex VR and real-life scenarios by standardizing data input, improving robustness, optimizing computational efficiency, and enabling cross-domain applications. They can significantly improve applications like sign language to speech conversion by ensuring consistent and accurate data processing, enabling real-time translation of hand gestures captured in VR into speech. This is crucial for developing assistive technologies that empower individuals with speech impairments. As VR technology evolves, these padding strategies will be essential for creating more sophisticated, responsive, and adaptable systems for real-world interactions. Continued refinement and innovation in these techniques will drive the next generation of immersive and accessible technologies.

ACKNOWLEDGMENTS

The authors wish to express their profound gratitude to the Directorate of Research, Technology, and Community Service, Ministry of Education, Culture, Research, and Technology of the Republic of Indonesia, for their generous financial support through the 2024 Fiscal Year Research Program. We are equally grateful to Universitas Budi Luhur for providing essential infrastructure, institutional backing, and an enabling environment that greatly contributed to the successful completion of this study. The support, both financial and technical, has been invaluable in ensuring the research's success.

AUTHORS CONTRIBUTION

Teja Endra Eng Tju: Conceptualization, methodology, formal analysis, supervision, and writing—original draft preparation;

Julaiha Probo Anggraini: Writing—review and editing, and project administration;

Muhammad Umar Shalih: Data curation, investigation, software, and validation;

All authors have read and approved the final version of the manuscript.

COPYRIGHT



This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.

REFERENCES

- [1] S. D. Sulistyaningrum, R. S. Putri, A. Herawati, and S. Irianto, "Trends of virtual reality for learning empirical evidence from different fields," *Journal of Education and Learning (EduLearn)*, vol. 16, no. 4, pp. 531–541, Nov. 2022, doi: 10.11591/edulearn.v16i4.20648.
- [2] L. Muñoz-Saavedra, L. Miró-Amarante, and M. Domínguez-Morales, "Augmented and virtual reality evolution and future tendency," *Applied Sciences (Switzerland)*, vol. 10, no. 1, 2020, doi: 10.3390/app10010322.
- [3] M. Alizadeh and N. Cowie, "Self-directed learning using VR," *Pacific Journal of Technology Enhanced Learning*, vol. 4, no. 1, pp. 10–11, Feb. 2022, doi: 10.24135/pjtel.v4i1.130.
- [4] H. R. Spero, I. Vazquez-Lopez, K. Miller, R. Joshaghani, S. Cutchin, and J. Enterkine, "Drones, virtual reality, and modeling: communicating catastrophic dam failure," *Int J Digit Earth*, vol. 15, no. 1, pp. 585–605, Dec. 2022, doi: 10.1080/17538947.2022.2041116.
- [5] N. Albertini, J. Baldini, A. Dal Pino, F. Lazzari, S. Legnaioli, and V. Barone, "PROTEUS: an immersive tool for exploring the world of cultural heritage across space and time scales," *Herit Sci*, vol. 10, no. 1, 2022, doi: 10.1186/s40494-022-00708-3.
- [6] G. E. Lysenko, A. V. Shehegolev, B. N. Bogomolov, and D. P. Meshakov, "Virtual reality therapy in patients after trauma surgery: prospective clinical trial," *Annals of Critical Care*, vol. 2023, no. 3, 2023, doi: 10.21320/1818-474X-2023-3-58-65.
- [7] C. P. Kamm, R. Kueng, and R. Blättler, "Development of a new immersive virtual reality (VR) headset-based dexterity training for patients with multiple sclerosis: Clinical and technical aspects,"

- Technology and Health Care*, pp. 1–12, Aug. 2023, doi: 10.3233/THC-230541.
- [8] C. P. Kamm, R. Blättler, R. Kueng, and T. Vanbellingen, “Feasibility and usability of a new home-based immersive virtual reality headset-based dexterity training in multiple sclerosis,” *Mult Scler Relat Disord*, vol. 71, p. 104525, Mar. 2023, doi: 10.1016/j.msard.2023.104525.
- [9] Z. Jiang, A. Meltzer, and X. Zhang, “Using virtual reality to implement disability studies’ advocacy principles: uncovering the perspectives of people with disability,” *Disabil Soc*, pp. 1–21, Jan. 2023, doi: 10.1080/09687599.2022.2150601.
- [10] A. Tariq, T. Rana, and M. Nawaz, “Virtual Reality for Disabled People: A Survey,” in *2018 12th International Conference on Open Source Systems and Technologies (ICOSST)*, IEEE, Dec. 2018, pp. 17–21. doi: 10.1109/ICOSST.2018.8632182.
- [11] N. R. Rahim and A. A. A. Rahim, “A VIRTUAL REALITY APPROACH TO SUPPORT MALAYSIAN SIGN LANGUAGE INTERACTIVE LEARNING FOR DEAF-MUTE CHILDREN,” *J Pharm Negat Results*, vol. 13, 2022.
- [12] K. Sangeethalakshmi, K. G. Shanthi, A. Mohan Raj, S. Muthuselvan, P. Mohammad Taha, and S. Mohammed Shoaib, “Hand gesture vocalizer for deaf and dumb people,” *Mater Today Proc*, vol. 80, 2023, doi: 10.1016/j.matpr.2021.07.324.
- [13] A. Vaitkevicius, M. Taroza, T. Blažauskas, R. Damaševičius, R. Maskeliunas, and M. Woźniak, “Recognition of American Sign Language gestures in a Virtual Reality using Leap Motion,” *Applied Sciences (Switzerland)*, vol. 9, no. 3, 2019, doi: 10.3390/app9030445.
- [14] A. Shaw *et al.*, “JengASL: A Gamified Approach to Sign Language Learning in VR,” *J WSCG (Plzen)*, vol. 31, no. 1–2, 2023, doi: 10.24132/JWSCG.2023.4.
- [15] G. Park and W. Woo, “Hybrid 3D Hand Articulations Tracking Guided by Classification and Search Space Adaptation,” in *Proceedings of the 2018 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2018*, 2018. doi: 10.1109/ISMAR.2018.00029.
- [16] F. Wen, Z. Zhang, T. He, and C. Lee, “AI enabled sign language recognition and VR space bidirectional communication using triboelectric smart glove,” *Nat Commun*, vol. 12, no. 1, 2021, doi: 10.1038/s41467-021-25637-w.
- [17] N. Takayama, G. Benitez-Garcia, and H. Takahashi, “Masked Batch Normalization to Improve Tracking-Based Sign Language Recognition Using Graph Convolutional Networks,” in *Proceedings - 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2021*, 2021. doi: 10.1109/FG52635.2021.9667007.
- [18] F. Reggiswarashari and S. W. Sihwi, “Speech emotion recognition using 2D-convolutional neural network,” *International Journal of Electrical and Computer Engineering*, vol. 12, no. 6, 2022, doi: 10.11591/ijece.v12i6.pp6594-6601.
- [19] F. Alrasheedi, X. Zhong, and P. C. Huang, “Padding Module: Learning the Padding in Deep Neural Networks,” *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3238315.
- [20] X. Hu, X. Wei, Y. Gao, W. Zhuang, M. Chen, and H. Lv, “An attention-mechanism-based traffic flow prediction scheme for smart city,” in *2019 15th International Wireless Communications and Mobile Computing Conference, IWCMC 2019*, 2019. doi: 10.1109/IWCMC.2019.8766639.
- [21] T. E. E. Tju and M. U. Shalih, “Hand Sign Interpretation through Virtual Reality Data Processing,” *Jurnal Ilmu Komputer dan Informasi*, vol. 17, no. 2, pp. 185–194, Jun. 2024, doi: 10.21609/jiki.v17i2.1280.
- [22] “Meta Quest 2: Our Most Advanced New All-in-One VR Headset | Oculus | Meta Store.” Accessed: Aug. 04, 2024. [Online]. Available: <https://www.meta.com/quest/products/quest-2/tech-specs/#tech-specs>
- [23] “Top 10 & 25 American Sign Language Signs for Beginners – The Most Known Top 10 & 25 ASL Signs to Learn First | Start ASL.” Accessed: Aug. 04, 2024. [Online]. Available: <https://www.startasl.com/top-10-25-american-sign-language-signs-for-beginners-the-most-know-top-10-25-asl-signs-to-learn-first/>
- [24] “‘you’ ASL American Sign Language.” Accessed: Aug. 04, 2024. [Online]. Available: <https://www.lifeprint.com/asl101/pages-signs/y/you.htm>
- [25] “‘help’ American Sign Language (ASL).” Accessed: Aug. 04, 2024. [Online]. Available: <https://www.lifeprint.com/asl101/pages-signs/h/help.htm>
- [26] “‘me’ American Sign Language (ASL).” Accessed: Aug. 04, 2024. [Online]. Available: <https://www.lifeprint.com/asl101/pages-signs/m/me.htm>
- [27] “Meta Quest 2: Immersive All-In-One VR Headset | Meta Store | Meta Store.” Accessed: Aug. 04, 2024. [Online]. Available: <https://www.meta.com/quest/products/quest-2/#overview>
- [28] “Unity - Manual: Unity User Manual 2022.3 (LTS).” Accessed: Aug. 04, 2024. [Online]. Available: <https://docs.unity3d.com/Manual/index.html>
- [29] “Unity - Scripting API: CommonUsages.” Accessed: Aug. 04, 2024. [Online]. Available: <https://docs.unity3d.com/ScriptReference/XR.CommonUsages.html>
- [30] “[1808.03314] Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network.” Accessed: Aug. 11, 2024. [Online]. Available: <https://arxiv.org/abs/1808.03314>
- [31] A. Sherstinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” *Physica D*, vol. 404, p. 132306, Mar. 2020, doi: 10.1016/j.physd.2019.132306.
- [32] M. K. Suryadewiansyah and T. E. E. Tju, “Naïve Bayes dan Confusion Matrix untuk Efisiensi Analisa Intrusion Detection System Alert,” *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 8, no. 2, pp. 81–88, 2022, doi: 10.25077/teknosi.v8i2.2022.81-88.