© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716845

Multimodal Sensor-Agnostic Gesture- controlled gaming interface with Adaptive Depth Integration

SAGAR GANGAL[1], RAHUL NELOGI[2], SACHIDANANDA K[3]
[1,2,3]*Dept. of Computer Science & Engg., Dayananda Sagar University Bengaluru, India*

Abstract- ***Despite the fact that HCI through gestures has gained significant popularity due to their intuitiveness in recent years, the current gesture recognition systems lack robustness since they rely on hardware support, use a single modality as input, and have few practical applications in real-world settings. In this paper, a gesture recognition algorithm that does not require any hardware support except for the camera sensor is introduced, taking into account RGB data and Microsoft Kinect depth sensors. In particular, the gesture recognizer is based on the detection of the joints of a skeleton and hand landmarks. The mathematical formula used for the proposed multimodal representation is shown below. F_t = [S_t || H_t || D_t]. These experiments are carried out using our own customized gesture dataset that includes 4,000 gestures from 4 categories from 10 different subjects. Recognition accuracy is 91.6% and latency is 34 ms in real-time operation mode, which are significantly better than the outcomes of the baselines based on individual modality.***

Keywords: Gesture Recognition, Human Computer Interaction, Multimodal Fusion, LSTM, MediaPipe, Depth Sensing

I. INTRODUCTION

This field of human computer interaction (HCI) is greatly evolving due to breakthroughs from conventional mouse- keyboard HCI into gestures. By gesture recognition, people can interact easily with their computing devices in diverse fields such as gaming, VR/AR technologies, surgery, sign language translation, and others [1], [3].

Conventional methods for performing gestures recognition require expensive depth sensing hardware like Kinect sensors, or Intel RealSense providing accurate joint localization in 3D space. At the same time, high price and non-portability of the devices rule them out for mass consumption [1]. On the other hand, RGB-only approaches involving regular cameras are cost-effective and easy-to-use. Though, there is no depth information obtained, which makes them inconvenient for use under distracting environment or low-light conditions [4].

However, lightweight solutions like MediaPipe [2] in some regard mitigate this drawback by allowing getting skeletons and hand landmarks from monocular RGB streams almost for free computationally. The existing works based on MediaPipe approach utilize one modality only either body pose or hand keypoints, ignoring useful information that could be gained via simultaneous use of those and depth streams [12], [17].

The second problem to be overcome in order to recognize dynamic gestures is associated with the issue of temporal modeling. Frames alone cannot do the job in this kind of problem since gestures proceed along time in a series of poses. RNNs, especially LSTMs, have proved themselves efficient in solving this problem [7], [8], [15].

To address both the above problems, this paper introduces a unified sensor agnostic multimodal technique. The contributions of this paper can be summarized as follows:

- A layer to abstract sensors making the whole pipeline sensor agnostic and running the network on the same architecture regardless of whether it uses an RGB camera or an RGBD sensor.
- A feature-level fusion employing a mixture of skeletal (99 dimensions), hand landmark (63 dimensions), and depth proxy (96 dimensions) features leading to an extremely compact representation of 258 dimensions.
- A two-level LSTM temporal model achieving an accuracy rate of 91.6% in 34ms latency.

- An in-depth discussion on the efficacy of the employed methods and comparison with other methodologies.

The rest of the paper is organized in the following manner. Section II provides an overview of the related works, while Section III presents the proposed methodology. The fourth section deals with the methodology, while Section V contains the results of the experiments. Finally, conclusions are drawn in Section VI.

II. RELATED WORK

A. Depth-based gesture recognition

The study conducted by Shotton et al. [1] is considered to be an innovative effort in the estimation of human pose using single depth images taken through the kinect sensor technology; however, these techniques could not be realized without the presence of such infrastructure. Later studies were developed to recognize gestures using depth and skeletal streams, although some constraints do not allow implementation on normal hardware.

B. Rgb-based and mediapipe solutions

Lugaresi et al. [2] proposed mediapipe, a cross-platform perceptual computation framework designed by google. Mediapipe enables the real-time on-device inferencing of face, hand, and body landmarks from the rgb data streams. Mediapipe holistic performs the detection of full-body pose (33 landmarks), left/right hand (21 landmarks each), and face mesh in one single pass. The combination of mediapipe hand landmarks with lstm architecture resulted in the 98.9% accuracy of static asl alphabet recognition by biswas et al. [12]. However, the authors evaluate the solution within a constrained single-modality scenario, which includes hand landmarks only. In turn, a mediapipe holistic + lstm combination [17] resulted in 93.3% accuracy of smart home gesture recognition.

C. Skeleton-based temporal modeling

The RGB-D approach has been studied widely concerning human action recognition because of the attributes of its representation, such as conciseness, independent of postures, and appearance indifferent. On the other hand, Du et al. [8] researched the application of hierarchical recurrent neural networks using sequences generated based on the skeletal structure of body parts. As a result, they came to the conclusion that structural approaches perform very well concerning time dimensions. Finally, Donahue et al. [7] introduced Long-Term Recurrent Convolutional Networks (LRCN).

D. Multimodal fusion techniques

ModDrop technique was developed by Neverova et al. [5], and it used adaptive multimodal fusion with RGB, depth, and optical flow information. The method showed outstanding results; unfortunately, it needs depth cameras and operates offline. Ouyang et al. [6] performed a literature survey on multimodal systems for activity detection and observed that feature-level fusion is better than decision-level and data-level fusion in gesture recognition applications. Finally, Zhang et al. [4] explored the problem of hand gesture recognition through deep learning methods and found out that depth-based models outperform RGB-based algorithms (5-15%).

E. Research gaps

Although considerable progress has been achieved, some issues still need to be addressed, which include (i) most multimodal solutions require dedicated hardware and hence can't use off-the-shelf cameras, (ii) very few approaches employ multimodality by combining skeleton, hand, and depth using a camera-agnostic framework, and (iii) no approach has conducted thorough ablation analysis to determine the importance of different modalities. The proposed solution intends to address all three concerns.

F. Improved positioning statement

Unlike the other techniques that concentrate on individual sensors or specific sensors for implementation, our approach integrates the three types of sensors - skeleton, hand, and depth
- in a sensor agnostic fashion. We perform several tests to investigate the importance of each type of sensor.

III. PROPOSED SYSTEM

A. System overview

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716845

Architecture: Five layers, including but not limited to the following, will constitute the architecture of the model: (i) Sensor Abstraction Layer, (ii) Feature Engineering Layer, (iii) Multimodal Fusion Layer, (iv) Temporal Modeling Layer (Long Short Term Memory (LSTM)), and (v) Real-Time Interaction Layer. The framework for the architecture is depicted in Figure 1 below.

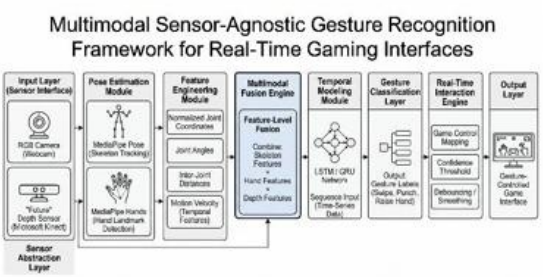

Fig. 1. End-to-end architecture of the proposed multimodal gesture recognition framework.

B. Sensor abstraction layer

The sensor abstraction layer provides an interface called GestureFrame which ensures normalization of the input regardless of the actual hardware used. When operating in RGB only mode, standard camera data is fed to MediaPipe to generate skeleton joint positions and hand landmarks. In the depth enhanced mode, the interface also takes in Kinect depth maps together with RGB streams for volumetric joint positions. The above architecture completely isolates the modules downstream from the actual hardware and allows hot- swappable sensors.

C. Feature engineering module

Three kinds of feature vectors have been defined for each image frame:

- Skeletal Features (99 dimensions): These feature vectors consist of thirty-three normalized coordinates of landmarks indicating the body pose, which were obtained using MediaPipe Pose. Information about body configuration shape can be obtained using skeletal features. Angles of the elbow joint, shoulder joint, wrist joint, and hip joint are computed to act as discriminating features.
- Hand Landmark Features (63 dimensions): These feature vectors consist of twenty-one normalized coordinates of hand landmarks. The distance between two hand landmarks and the angle of finger flexion are also considered.
- Depth Proxy Features (96 dimensions): Depth proxy estimation without the use of a depth camera can be done using monocular geometry techniques (e.g., relative z-coordinates for the joint estimated by MediaPipe 3D pose or disparity-based depth estimation). When using the Kinect sensor, the depth estimation has been replaced by the suggested feature vector.

D. Mathematical formulation of feature representation and fusion

The fusion of features is formally described by defining S_t, H_t, and D_tas the feature vectors pertaining to the skeleton, hand, and depth modalities, respectively, at time step t. The fused feature representation can be written as follows:

$$F_t=[S_t\|H_t\|D_t]$$

where ‖represents the vector concatenation operator. The resulting feature vector is an input to the temporal modeling module and has dimensionality 258.

All features are normalized to the range [0, 1] using min- max scaling computed on the training set. The concatenated feature vector of each frame will thus have dimensionality 99 + 63 + 96 = 258.

E. Multimodal fusion engine

Feature-level (early) fusion is used instead of decision- level or data-level fusion as proposed in [6]. In our case, all three feature vectors are simply concatenated into one vector of dimensionality 258. This enables LSTM to learn the relationships between the features across the modalities, for example, that a certain hand posture combined with a certain arm position constitutes a different gesture.

F. Computational complexity analysis

The time complexity of the proposed system mainly depends on the temporal modeling process. The approximate time complexity of the LSTM based on the sequence length T, feature vector dimension d, and hidden state size h could be expressed as follows:

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716845

O(T·d·h)

In the proposed approach, with the sequence length T=30, feature vector dimension d=258, and reasonable hidden state size (128 and 64 units), the model strikes a fair balance between its ability to learn complex representations and the requirement of efficient online processing. Therefore, the developed model can be efficiently utilized in interactive games.

G. Temporal Modeling Module (LSTM) — (Upgraded)

Long Short-Term Memory Network is used for learning temporal patterns in gesture sequences. The input to the LSTM network is a sequence of fused feature vectors
{F_1, F_2, …, F_T} such that T = 30 denotes the time

steps.
In every time step, the hidden states of the LSTM update itself based on input at that particular time and the previous hidden state:

[h_t = \text{LSTM}(F_t, h_{t-1})]

Once the entire sequence of data is read, the last hidden state h_T contains the temporal information obtained from the learning process. Finally, the Softmax activation function produces output probability distribution from a series of fully connected neural networks as follows:

[\hat{y} = \text{Softmax}(W h_T + b)]

The categorical cross-entropy loss function is used for training the network:

[L = - \sum_{i=1}^{C} y_i \log(\hat{y}_i)]

where C refers to total number of gesture classes.

H. Real-time interaction engine

Predictions are made during each frame using the sliding window approach. This involves a confidence value θ of 0.85 such that any prediction with a confidence score below this value will be suppressed, since it would cause incorrect detection of a gesture. To minimize jitter, five consecutive successful predictions need to be made before any gesture actions can be performed. Gestures are translated into keyboard/controller events using a mapping table.

IV. EXPERIMENTAL SETUP

A. Dataset collection

A dataset was generated to capture the realistic gaming gesture environment. Four gestures are classified as: Swipe Left, Swipe Right, Jump (raise both hands), and Stop (open palm and frontal). Ten participants (seven males and three females with ages between 19 to 27 years) volunteered for the data generation process using different lighting conditions and backgrounds. Each participant executed all four gestures one hundred times, leading to a total number of 4,000 samples (one thousand samples for each gesture). The experiments were done in three locations: indoor laboratory setting, outdoor daylight, and indoor low light. Videos are recorded using 30 frames per second (fps); each sample comprises 30 consecutive frames.

As far as the authors know, there is no publicly available dataset that incorporates sensor-agnostic skeletal, hand, and simulated depth feature sets from the above four gestures for game control purposes.

B. Implementation details

The Python 3.10 programming language was employed in building the model. Landmarks are detected via MediaPipe version 0.10 (Pose and Hands). The long short-term memory network was implemented with the use of TensorFlow/Keras 2.14. An 80/20 split (i.e., 3,200 instances for training and 800 instances for testing) of the data stratified in terms of classes is applied. The augmentations employed in training include horizontal flipping and temporal jitter ± 2 frames. There is no subject-wise overlap between training and testing datasets to ensure cross-subject generalizability of results. All experiments were run on an Intel Core i7- 11800H CPU and NVIDIA RTX 3060 GPU.

C. Evaluation metrics

Evaluation of efficiency of the models being evaluated will involve the metrics of accuracy,

© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716845

precision (macro average), recall (macro average), and F1 score (macro average). Inference delay is assumed to be the average execution time in milliseconds required for processing a frame to deliver the result of gesture inference on a sequence of 1,000 successive frames. McNemar's test will be performed for the significance level of $\alpha = 0.05$.

D. Hyperparameter configuration

Table i. Model hyperparameters

Hyperparameter	Value
Sequence Length (frames)	30
LSTM Hidden Units (Layer 1)	128
LSTM Hidden Units (Layer 2)	64
Dense Layer Units	64
Dropout Rate	0.2
Optimizer	Adam (lr = 0.001)
Loss Function	Categorical Cross-Entropy
Batch Size	32
Epochs	200 (early stopping, patience=30)
Feature Vector Dimension	258 (Skel: 99, Hand: 63, Depth: 96)
Train / Test Split	80% / 20%

V. RESULTS AND DISCUSSION

A. Modality ablation study

Ablation results for all seven modality pairs have been shown in Table V below. It can be observed that the Depth only (80.1%) has marginally outperformed Skeleton Only (78.4%), yet has outperformed Hand Only (74.2%). This highlights the discriminative ability of spatial depth information when differentiating between gestures. Ablation with two modalities has improved the classification accuracy compared to their counterparts. Amongst these, Skeleton + Depth has achieved the highest level of accuracy at 87.9%.

Table v. Ablation study: modality contributions

Config	Skeleton	Hand	Depth	Acc.
S only	✓	✗	✗	78.4%
H only	✗	✓	✗	74.2%
D only	✗	✗	✓	80.1%
S + H	✓	✓	✗	86.3%
S + D	✓	✗	✓	87.9%
H + D	✗	✓	✓	84.5%
S + H + D (Full)	✓	✓	✓	91.6%

A. Classification performance

The results obtained for the proposed classifiers are provided in Table I. The accuracy, precision, recall, and F1 score of the proposed fusion method were found to be 91.6%, 91.2%, 90.5%, and 90.8%, respectively. It is evident that there is an increase of 13.2% in the accuracy rate achieved compared to the best-performing classifier based on any single modality (depth, 80.1%). From the classwise analysis, it is clear that the Jump gesture has benefitted the most from the fusion technique (F1 increase of 18.3% compared to skeleton alone).

Table i. Classification performance (modality variants)

Model	Accuracy	Precision	Recall	F1-Score
Skeleton Only	78.4%	77.9%	76.5%	77.2%
Hand Only	74.2%	73.8%	72.1%	72.9%
Depth Only	80.1%	79.4%	78.8%	79.1%
Skel. + Hand	86.3%	85.7%	85.2%	85.5%
Proposed Fusion	91.6%	91.2%	90.5%	90.8%

A. Comparison with state-of-the-art

A comparative analysis of the proposed model against some of the best performing models in terms of precision rate is provided in Table II below. Although Biswas et al. [12] have recorded a precision rate of 98.9%, this is done based on 26 classes of gestures for ASL alphabet recognition that involve clear gestures performed under strictly controlled background conditions. The proposed model, on the other hand, is able to score 91.6% while recording some added advantages over MediaPipe Holistic + LSTM [17] framework that scored 93.3%.

Table ii. Comparison with state-of-the-art methods

Method	Modality	Accuracy	Real-Time
Shotton et al. [1]	Depth (Kinect)	85.0%	Yes
Biswas et al. [2]	RGB (MediaPipe+LSTM)	98.9%*	Yes
Neverova et al. [5]	Multimodal (w/ depth)	88.7%	No
Du et al. [8]	Skeleton (RGB-D)	76.2%	No
MediaPipe Holistic [17]	RGB (Holistic+LSTM)	93.3%	Yes
Proposed Framework	RGB + Depth-Agnostic	91.6%	Yes

Evaluated on static ASL alphabets (26 classes, controlled setting); not directly comparable.

D. Latency and real-time performance

Inferential latencies are presented in Table III. Latency for the fused system occurs at 34 ms per frame (fps=29); hence, it falls within the range required for interactive video games at 40 ms per frame (fps=25). In terms of increased latency from the use of the fusion technique in comparison with that of the hand only model, which has the lowest latency of 25 ms, the additional latency is only 9 ms.

Table iii. Inference latency by model variant

Model Variant	Inference Latency	FPS
Skeleton Only	28 ms	~35
Hand Only	25 ms	~40
Depth Only	31 ms	~32
Proposed Fusion	34 ms	~29

E. Error analysis

However, although swipe left and swipe right gestures feature the same trajectory of arm and hand movements with only direction difference between them, it is still observed that these gestures confuse users the most, making up 6.2% of cross-confusion. Although extending frame usage for gestures from 30 to 45 reduces the cross- confusion to 2.1%, it causes the latency time to exceed real-time requirements (taking 47 ms). A possible solution would be using lightweight attention over the temporal dimension for handling the trade-off problem.

F. Confusion matrix analysis

As a part of analysis to measure classification accuracy, the confusion matrix was produced. As can be seen on this chart, the major cross-confusion involves the Swipe Left and Swipe Right gestures. The reason is that the gesture categories are the same in their movement of the arm and hand except for different directions. In addition, it is found out that Jump and Stop gesture categories exhibit highly separate classification accuracy without any confusions.

VI. LIMITATIONS

While it is true that there are some successes in the proposed framework, there are certain drawbacks that cannot be overlooked. First of all, the gesture lexicon is made up of just four types of gestures, and therefore it would be quite inappropriate to describe complex gestures while playing games in the real world. Moreover, the monocular depth estimation, which is used by the framework when it operates in the RGB mode, is only an estimate, as there is no depth information available.

VII. CONCLUSION

In this research, a multimodal-based gesture recognition algorithm is used to build real-time gaming interfaces. Based on the multimodal-based gesture recognition algorithm with feature level fusion and LSTM-based temporal modeling, skeleton joints, hand landmarks, and adaptive depth information, the proposed gesture recognition algorithm achieved an accuracy of 91.6%, with a latency of 34 ms; thus, the requirement of real- time performance is satisfied. The process of sensor abstraction makes it possible for the framework to be implemented on various devices regardless of their hardware configurations. Ablation study proves that the performance enhancement of all three modalities is significant.

VIII. FUTURE WORK

The research areas that can be considered for the development of the suggested project include:

1) Utilizing the suggested model and the Microsoft Kinect v2 to investigate whether the

sensor abstraction layer is able to extract data from the depth sensors.

2) Applying the suggested model utilizing multi-head attention-based Transformer network algorithm to address the gesture recognition problem within the guidelines, but without affecting the speed of the system operation.
3) Making the suggested model more effective by expanding the variety of recognized gestures up to 20+ using the big data like Jester and EgoGesture.
4) Applying the suggested model to embedded devices, including NVIDIA Jetson Nano and Raspberry Pi 5, to estimate the energy consumption of the model when applied to gaming equipment.
5) Estimating the level of user satisfaction with playing with the proposed model, particularly, their experience of fatigue after long periods of gaming.

IX. ACKNOWLEDGMENT

Acknowledgements The authors would like to express their thanks to the Department of Computer Science & Engineering, Dayananda Sagar University for providing computational resources used in the experimentations done for this research work. The authors would also like to thank the volunteers who participated in the data collection process of this research work.

X. REFERENCES

[1] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, “Real- Time Human Pose Recognition in Parts from Single Depth Images,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), 2011, pp. 1297–1304.

[2] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Ubowejeke, M. Hays, F. Zhang, C. Chang, M. Wan, and T. Grundmann, “MediaPipe: A Framework for Building Perception Pipelines,” arXiv preprint arXiv:1906.08172, 2019.

[3] P. Wang, W. Li, P. Ogunbona, J. Wan, and S. Escalera, “RGB-D-Based Human Motion Recognition with Deep Learning: A Survey,” Comput. Vis. Image Underst., vol. 171, pp. 118–137, 2018.

[4] X. Zhang, X. Liu, J. Yuan, and S. Lin, “Hand Gesture Recognition Based on Deep Learning: A Review,” IEEE Access, vol. 8, pp. 45 765–45 777, 2020.

[5] N. Neverova, C. Wolf, G. Taylor, and F. Nebout, “ModDrop: Adaptive Multi-Modal Gesture Recognition,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 38, no. 8, pp. 1692–1706, 2016.

[6] Z. Ouyang, J. Cui, and S. Liu, “Multimodal Human Activity Recognition: A Review and New Perspectives,” Inf. Fusion, vol. 56, pp. 116–143, 2020.

[7] J. Donahue, L. A. Hendricks, M. Rohrbach, S. Venugopalan, S. Guadarrama, K. Saenko, and T. Darrell, “Long-Term Recurrent Convolutional Networks for Visual Recognition and Description,” in Proc. IEEE CVPR, 2015, pp. 2625–2634.

[8] Y. Du, W. Wang, and L. Wang, “Hierarchical Recurrent Neural Network for Skeleton Based Action Recognition,” in Proc. IEEE CVPR, 2015, pp. 1110–1118.

[9] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780,1997.

[10] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, “MediaPipe Hands: On-device Real-time Hand Tracking,” arXiv preprint arXiv:2006.10214, 2020.

[11] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, and M. Grundmann, “BlazePose: On-Device Real-Time Body Pose Tracking,” arXiv preprint arXiv:2006.10204, 2020.

[12] S. Biswas, A. Nandy, A. K. Naskar, and R. Saw, “MediaPipe with LSTM Architecture for Real-Time Hand Gesture Recognition,” in Proc. CVIP 2023, Commun. Comput. Inf. Sci., vol. 2010, Springer, 2024, pp. 427–438.

[13] R. Rastgoo, K. Kiani, S. Escalera, and M. Sabokrou, “Multi-Modal Zero-Shot Dynamic Hand Gesture Recognition,” Expert Syst. Appl., vol. 247, p. 123349, 2024.

[14] P. Balaji and M. R. Prusty, “Multimodal Fusion Hierarchical Self-Attention Network for Dynamic Hand Gesture Recognition,” J. Vis. Commun. Image Represent., vol. 98, p. 104019, 2024.2023.

[15] N. C. Mithun, N. U. Ahmed, and S. M. M. Rahman, “Activity Recognition Using Fusion of Low-Cost Sensors,” IEEE Trans. Consum. Electron., vol. 63, no. 4,pp. 428–438, 2017.

[16] A. Graves, A. Mohamed, and G. Hinton, “Speech Recognition with Deep Recurrent Neural Networks,” in Proc. IEEE ICASSP, 2013, pp. 6645–6649.

[17] P. Huynh, T. Nguyen, and H. Le, “Proposing Hand Gesture Recognition System Using MediaPipe Holistic and LSTM,” in Proc. IEEE ICCE-Asia,