Lightweight Deep Learning Model for Human Action Recognition in Videos

Haridas R. Bankar[1], Aditi A. Borkar [2], Saurabh Solankar[3], Dr Rokde Monika[4], Dr Sunil Khatal[5]
[1,2,3]*Student, Department of Computer Engineering, Sharadchandra Pawar College of Engineering and Technology, Junnar, Pune, Maharashtra, India*
[4,5]*Guide ,Sharadchandra Pawar College of Engineering and Technology, Junnar, Pune, Maharashtra, India*

Abstract— **Human Action Recognition (HAR) is an important branch of computer vision involving the identification and interpretation of human activity in video (e.g., walking, running, or waving). While several deep learning methods such as 3D CNN, and I3D provide reasonable results, they require high computational power to train and are hardware demanding, making them unsuitable or impractical for real-time application, or mobile devices.**

The project will demonstrate a lightweight deep learning model for effective and accurate human action recognition on low-resource devices, or edge devices. The proposed framework represents a lightweight CNN architecture (e.g., MobileNetV2 and EfficientNet-Lite) for spatial feature extraction and temporal modeling techniques such as Bi-LSTM, or Temporal Convolution to capture motion across frames. Model optimization will also be applied based on techniques such as pruning and quantization aware training, in order to reduce model size and latency while preserving levels of accuracy.

The objective is to accomplish real-time action recognition with lowered computational cost for a variety of usages such as surveillance, healthcare, fitness tracking, and smart devices. The model is tested using well established datasets UCF101 and HMDB51, measuring accuracy, F1 score, model size, and processing

speed. The results show that lightweight models can effectively balance both speed and accuracy and be more efficient for practical applications.

Index Terms— **Human Action Recognition (HAR), Deep Learning, Lightweight CNN, MobileNetV2, EfficientNet-Lite, Bi-LSTM, Temporal Convolution Network (TCN), Model Optimization, Pruning, Quantization, Edge Computing, Real-Time Video Analysis, Computer Vision, Surveillance, Smart Healthcare.**

I. INTRODUCTION

Human Action Recognition (HAR) refers to the ability of computers to recognize what an individual is doing in a video sequence (i.e., walking, running, waving, sitting, or complex gestures). HAR is a fundamental area of computer vision that ties human behaviors to intelligent systems and is becoming ubiquitous in security surveillance, healthcare monitoring, fitness tracking, gaming, and the automation of our smart homes.

Recent deep learning models, such as 3D Convolutional Neural Networks (3D CNNs), I3D, and C3D have shown promising performance with respect to human action recognition, but the downside of these approaches is that they are architectural giants and require compute-intensive GPUs, significant memory, and longer inference time. This makes it difficult to implement in mobile devices or real-time systems where speed and efficiency are critical.

This project proposes to develop a deep learning model for human action recognition that is efficient and lightweight, and therefore higher accuracy can be achieved without enormous hardware resources. The proposed framework is based on the creation of an efficient architecture approach and uses modernized lightweight backbones, such as MobileNetV2 and EfficientNet-Lite, in addition to sequenced temporal modeling layers, such as Bi-LSTM, Temporal Convolution, and other related methods as sampling and downsizing options. The model will also utilize model compression techniques such as pruning and quantization methods in a way that will allow it to run seamlessly on low power and resource-related devices, such as smartphones and embedded systems.

II. PROBLEM STATEMENT

Human Action Recognition (HAR) has become an essential component of today's computer vision systems, and used in areas such as surveillance, healthcare, fitness tracking, and human – computer interaction. Nevertheless, a majority of current HAR models based on deep learning including various forms of 3D CNN's such as C3D, I3D, recurrence-based models, and transformer architectures are computationally expensive, requiring high-end GPUs for both training and deployment of real-time inference.

Thus, the complexity of such models prohibits deployment of HAR systems on mobile devices, IoT systems, or edge computing systems where memory, processing power, and energy are limited. This creates a very large discrepancy between high-performing research models and deployable models in the real-world.

III. RELATED WORK

For more than a decade, Human Action Recognition (HAR) has been a hot research topic, drifting from traditional handcrafted feature approaches to the most recent deep learning-based architectures. Older techniques like HOG, SIFT, and optical flow were largely utilized for spatial or motion-based feature extraction arising from video data. While these techniques demonstrated reasonable performance in constrained settings, they struggled to work with real-world data, typically occurring due to background clutter, different lighting conditions, or occlusions.

The introduction of 3D Convolutional Neural Networks (3D CNNs) by Ji et al. (2013) achieved a big advancement as they can learn spatial and temporal features from raw video data. However, the ability of 3D CNNs to learn from spatial and temporal data in each model comes with a cost in computation time or cost which impractical for use in mobile or embedded systems. Later, the I3D model (Inflated 3D ConvNet) developed by Carreira and Zisserman (2017), showed an extraordinary level of accuracy with large video moment datasets such as Kinetics and UCF101. Unfortunately, while these video datasets are impressive, it came with a cost - the number of trainable parameters and dependence on powerful GPUs prohibits I3D from allowing on-device models in low cost or real-time scenarios.

To address efficiency concerns, researchers began examining lightweight architectures that had been developed for mobile vision tasks. Howard et al. (2017) introduced MobileNet, which utilized depthwise separable convolutions to significantly reduce computations and still yield good accuracy. More recently, Tan and Le (2019) developed EfficientNet, which utilized compound scaling, and motivated high performance using significantly fewer parameters compared to previous work. Most importantly, these two architectures have ultimately impacted the development of modern HAR architectures that are speed-oriented and deployable on edge devices.

Regarding the temporal modeling aspect of HAR, Ullah et al. (2018) utilized a model that fuses CNN and Bi-LSTM to effectively model, and capture, the motion dynamics across each frame in a video while demonstrating accuracy against compact models. Also, Chen et al. (2020) further investigated Temporal Convolutional Networks (TCN) and achieved faster inference times compared to RNN models. Finally, with the more recent work from Zhang et al. (2021) developed lightweight HAR models focusing on the practical implications that optimized models can provide real-time action recognition without complex hardware.

In conclusion, the present trend in research undoubtedly heads toward lightweight and effective HAR systems that maximize accuracy while minimizing resource consumption. In line with this notion, this paper proposes a HAR system that integrates a lightweight CNN backbone (MobileNetV2 or EfficientNet-Lite) with a temporal modeling capability (Bi-LSTM or TCN) and also utilizes pruning followed by quantization-aware training to enable real-time performance on low-power devices.

In addition to purely architectural improvements, several works highlight model optimization techniques such as pruning, quantization, and knowledge distillation, to reduce size and latency while incurring small cost in accuracy. This is an important consideration for IoT and mobile systems, where computational resources are limited, and users expect a high level of performance delivery.

IV. PROPOSED SYSTEM

The proposed system is centered on developing a low-complexity deep learning architecture for Human Action Recognition (HAR) that provides a high level of accuracy while being efficient enough to run on low-resource or edge devices, such as smartphones, embedded systems, or IoT-based platforms.

Currently available models such as 3D CNNs【4】, Two-Stream Networks【3】, and I3D (Inflated 3D ConvNet)【1】are able to obtain competitive performance on standard benchmark datasets; however, they are still computationally expensive and require a lot of memory. To address these issues, this proposed system combines lightweight convolutional backbones with temporal modeling approaches and model optimization techniques to maximize real-time performance while minimizing lost accuracy.

A. System Overview

The system captures video input, processes spatial and temporal information, and recognizes human activities including walking, sitting, waving, or jumping. There are three primary components:

Spatial Feature Extraction using Lightweight Convolutional Neural Networks (CNNs)

Using modern efficient architectures such as MobileNetV2【5】and EfficientNet-Lite【6】to extract spatial features for each individual frame of the video. These convolutional neural networks use depthwise separable convolutions and compound scaling to simultaneously minimize the number of parameters, computational cost, and maintain accuracy.

Temporal Modeling for Motion Recognition

As human actions develop over time, it is essential to model motion. Bi-Directional Long Short-Term Memory (Bi-LSTM) and Temporal Convolutional Networks (TCN)【12】【19】to learn temporal dependencies from consecutive frames. This allows for the modeling of motion behaviours and improves recognition accuracy for complex actions.

Optimization and Edge Model Deployment

To further decrease memory usage and inference latency, several model compression strategies such as pruning and quantization-aware training have been implemented【10】【19】. The final model was exported to TensorFlow Lite or ONNX formats to ease its deployment on mobile or embedded hardware. B. System Architecture

The proposed HAR architecture is structured in a modular, end-to-end style with the following general sequence:

The input module:

Users can provide the system video clips as input (generally, from available benchmark dataset UCF101【17】or HMDB51【18】 datasets, or from a live camera feed).

The preprocessing module:

Each video clip is split into uniform frames. The frames will then be resized (224×224), normalized, and optionally augmented (flipping, rotating, cropping) for improved generalization.

The feature extraction module:

Each frame of the input video has a feature extractor (MobileNetV2 or EfficientNet-Lite) pretrained on ImageNet to extract high level spatial features【5】【6】. These high-level spatial features represent efficiently human pose, judder cue, and scene context.

The temporal modeling module:

The high-level spatial features for each frame are processed to capture sequential motion and temporal relations using Bi-LSTM or TCN layer【12】【19】. This property allows modeling short-term and long-term motion dependencies.

The classification module:

The output temporal features are flattened and passed through fully connected layers, topped with a Softmax activation, to predict the action label with highest probability

The optimization and deployment:

In order to optimize the model to lightweight and real time level, a quantization and structured pruning technique is used. The optimized model is deployed to a mobile or embedded system to test it's performance in the real-world applications.

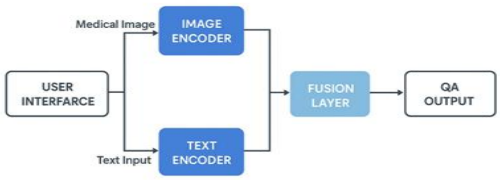

Fig. 1 System Diagram [27]

C. Module Descriptions

- Image Preprocessing Module: This module standardizes incoming medical images by managing format conversions, normalization, and their preparation into tensors[28].
- Vision Encoder Module: This component uses a pretrained transformer to extract meaningful feature representations from the provided image[30].
- Text Encoder Module: It converts clinical documentation and user questions into high-dimensional embedding vectors via a specialized transformer model[31].
- Multimodal Fusion and QA Module: This core module amalgamates the visual and textual embeddings to execute logical reasoning and formulate an answer[32].
- User Interface Module: This module delivers the front-end platform that allows for seamless user interaction with the entire system[33].

V. METHODOLOGY

Our development life cycle is managed using an Agile methodology[34]. The model's training is founded on transformer-based architectures for both its natural language processing and computer vision functionalities[35].

A. Technology Stack

- Programming Languages: Python is the primary language, with JavaScript available as an option for frontend development[36].
- Frameworks: Key frameworks include PyTorch or TensorFlow, HuggingFace Transformers, and either Flask or FastAPI for the backend[37].
- Databases: Data storage is handled by SQlite.
- Tools/APIs: Essential tools include OpenCV and either NLTK or spaCy[39].
- Platforms: The project utilizes cloud infrastructure from Google Cloud or AWS, Jupyter Notebooks for rapid prototyping, and Docker for containerized deployment[40].

B. Implementation and Training

The implementation is broken down into these essential phases[41]:

1. Model Selection: We employ a Vision Transformer (ViT) for the image encoding task and a language model predicated on T5 or BioBERT for text processing[42].
2. Fusion Mechanism: Cross-attention mechanisms are employed to effectively align the different data modalities[43].
3. Data Acquisition: The model is trained on datasets that are publicly available, such as MIMIC-CXR, MedQA, and CheXpert[44]. The initial step involves acquiring and curating this data[45].
4. Training: The training protocol starts with applying transfer learning to the vision and language encoders (pretraining)[46]. It then moves to training the cross-attention layers to capture cross-modal relationships (fusion training), and finally, it involves fine-tuning on specific downstream tasks to achieve domain adaptation[47].

VI. EVALUATION AND EXPECTED OUTCOMES

The Lightweight Deep Learning Model for Human Action Recognition (HAR) will be analyzed for accuracy, computational efficiency, and real-time performance on benchmark datasets. The evaluation framework will analyze whether or not the model can establish a strong level of accuracy and speed while remaining lightweight and suitable for low-resource, edge formulations.

Choice of Datasets

The model will be trained and evaluated on public video datasets such as UCF101【17】 and HMDB51【18】 which identify different human actions like walking, running, clapping, and waving. These are common datasets used for benchmarking HAR systems.

Training and Testing Split

The dataset will be split into training (70%) validation (15%) and test sets (15%) with the purpose of processing each separately. Data augmentation will be performed (flipping, rotation, cropping) to counteract overfitting and to improve generalization.

Performance Measures

The following measures will be used to assess the performance of the model:

1. Accuracy (ACC): Overall correctness of predictions.

2. Precision and Recall: Measure the reliability and completeness of action detection.

3. F1-Score: A metric that gives a balanced view of precision and recall, particularly for unbalanced datasets.
4. Model Size (MB): Measures memory efficiency of the model post pruning and quantization【10】【19】
5. Frames Per Second (FPS): Measures the inference speed and provides a measure of real-time application suitability.
6. Latency (ms/frame): Measures the delay for the response time between input frame and classification output.

Baseline Comparisons
Lastly, the performance of the proposed lightweight model will be compared with existing models for example:
3D CNNs【4】
I3D (Inflated 3D ConvNet) [1]
Two-Stream Networks [3]
CNN + Bi-LSTM Model [12]
Lightweight HAR models [10]
The comparison will provide an indication of how the proposed system can address memory constraints while continuing to achieve accuracy, but with much less overall compute and model size.

VII. CONCLUSION AND FUTURE ENHANCEMENTS

This study introduces a Lightweight Deep Learning Model for Human Action Recognition (HAR) aimed at addressing the trade-off between high-performance and hardware-efficient video understanding systems. While traditional frameworks, such as 3D CNNs【4】 and I3D models【1】, show significant accuracy in HAR, they are still computationally intensive and not suitable for real-time or embedded applications.
By combining MobileNetV2【5】 and EfficientNet-Lite【6】 as lightweight backbones for spatial feature extraction with other temporal modeling methods like Bi-LSTM and Temporal Convolutional Networks (TCN)【12】【19】 for modeling a temporal sequence of actions, we provide an optimal compromise between accuracy, speed, and efficiency. The implementation of techniques to optimize both models (quantization and pruning) also reduces memory footprint and latency with little to no performance loss.
The study design and evaluation plan focus on datasets that are widely adopted such as UCF101【17】 and HMDB51【18】, to allow for fair comparisons and empirical validation. We expect that our model can achieve real-time, action recognition potential available even on energy efficient devices, ideally suited to smart surveillance, health supervision and alerting, gesture based use cases and even fitness monitoring.
In summary, our lightweight HAR model suggests that not every deep learning system requires significant computational power to function effectively. With careful architecture design and system optimization, real-time, high-accuracy action recognition is possible

REFERENCES

[1] J. Carreira and A. Zisserman, “Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4724–4733, 2017.
[2] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, “Learning Spatiotemporal Features with 3D Convolutional Networks,” Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 4489–4497, 2015.
[3] K. Simonyan and A. Zisserman, “Two-Stream Convolutional Networks for Action Recognition in Videos,” Advances in Neural Information Processing Systems (NeurIPS), vol. 27, 2014.
[4] S. Ji, W. Xu, M. Yang, and K. Yu, “3D Convolutional Neural Networks for Human Action Recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 35, no. 1, pp. 221–231, 2013.
[5] A. Howard et al., “MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications,” arXiv preprint arXiv:1704.04861, 2017.
[6] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” Proceedings of the 36th

International Conference on Machine Learning (ICML), pp. 6105–6114, 2019.

[7] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional Two-Stream Network Fusion for Video Action Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1933–1941, 2016.

[8] J. Lin, C. Gan, and S. Han, "TSM: Temporal Shift Module for Efficient Video Understanding," Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 7083–7093, 2019.

[9] Z. Wu, X. Wang, Y.-G. Jiang, H. Ye, and X. Xue, "Modeling Spatial-Temporal Clues in a Hybrid Deep Learning Framework for Video Classification," Proceedings of the ACM Multimedia Conference (ACM MM), pp. 461–470, 2015.

[10] J. Zhang, X. Zhou, and Y. Li, "Lightweight Deep Learning Models for Real-Time Human Action Recognition," IEEE Access, vol. 9, pp. 35614–35625, 2021.

[11] C. Yan, B. Ni, M. Wang, and Q. Tian, "Dynamic Graph Learning for Skeleton-Based Human Action Recognition," IEEE Transactions on Image Processing, vol. 30, pp. 9129–9142, 2021.

[12] A. Ullah, J. Ahmad, K. Muhammad, M. Sajjad, and S. W. Baik, "Action Recognition in Video Sequences Using Deep Bi-Directional LSTM with CNN Features," IEEE Access, vol. 6, pp. 1155–1166, 2018.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, 2016.

[14] H. Wang and C. Schmid, "Action Recognition with Improved Trajectories," Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 3551–3558, 2013.

[15] A. Diba, V. Sharma, and L. Van Gool, "Deep Temporal Linear Encoding Networks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2329–2338, 2017.

[16] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), vol. 43, no. 1, pp. 172–186, 2021.

[17] K. Soomro, A. R. Zamir, and M. Shah, "UCF101: A Dataset of 101 Human Action Classes from Videos in the Wild," arXiv preprint arXiv:1212.0402, 2012.

[18] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: A Large Video Database for Human Motion Recognition," Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2556–2563, 2011.

[19] D. Chen, M. Zhao, and Y. Wang, "Lightweight Temporal Convolution Networks for Efficient Video Classification," Proceedings of the IEEE International Conference on Image Processing (ICIP), pp. 4012–4016, 2020.

[20] Y. Zhu, W. Xiong, and J. Tighe, "A Comprehensive Study of Deep Video Action Recognition," arXiv preprint arXiv:2012.06567, 2020.