Hybrid Human Activity Recognition: Integrating Traditional Feature Engineering with Deep Learning Approach

Vijaya J.[1*], Nenavathu Pranay[2], G.S. Abhinav[2], Alla Abhiram[2], Bypuneni Chaitanya Krishna[2]

[1]Assistant Professor/ Dept DSAI, IIITNR, Raipur, Chhattisgarh, India, 493661

[2]UG Student/ IIITNR, Raipur, Chhattisgarh, India, 493661

[*]Corresponding Author

DOI: https://dx.doi.org/10.51244/IJRSI.2025.12110095

Received: 25 November 2025; Accepted: 01 December 2025; Published: 10 December 2025

ABSTRACT

Human Activity Recognition (HAR) is a vital research area with applications in healthcare, security, and intelligent environments. This paper presents a hybrid framework that combines traditional feature engineering with deep learning to enhance HAR performance. It leverages the Histogram of Oriented Gradients (HoG) for spatial feature extraction and Support Vector Machines (SVM) for structured classification. Additionally, Vision Transformers (ViT) and ResNet architectures are integrated to improve accuracy: ViT captures global dependencies through attention mechanisms, while ResNet enhances deep feature learning through skip connections. Experimental results demonstrate that this approach balances computational efficiency, interpretability, and high accuracy on large datasets.

Keywords: Human Activity Recognition (HAR), Histogram of Oriented Gradients (HoG), Support Vector Machines (SVM), Vision Transformers (ViT), ResNet, Deep Learning.

INTRODUCTION

Human Activity Recognition (HAR) has become a pivotal research area owing to its wide-ranging applications across multiple fields, including healthcare, surveillance, and smart environments. This area of study is essential for understanding human behavior, driving automation, and the development of intelligent systems capable of adapting dynamically to changing conditions and responding appropriately [1-2]. In recent years, the integration of machine learning (ML) and deep learning (DL) techniques has revolutionized HAR by allowing systems to learn intricate patterns and complex features directly from raw, unprocessed data. This shift has enhanced the accuracy and robustness of activity recognition models, making them more versatile and scalable. Among these advancements, Support Vector Machines (SVM) combined with Histogram of Oriented Gradients (HoG), as well as deep learning architectures like ResNet and Vision Transformers (ViT), have gained prominence. HoG is widely used for extracting spatial features from video frames, while deep learning models such as ResNet and ViT offer superior feature learning capabilities. ResNet's residual blocks mitigate the vanishing gradient problem, allowing deeper networks to capture intricate spatial and temporal dependencies in human activities [3-7].

ViTs, on the other hand, treat images as sequences of patches, leveraging self-attention to capture local and global dependencies, making them particularly effective for complex datasets [8-10]. This paper proposes a hybrid framework that integrates HoG+SVM with ResNet and ViT to achieve robust and efficient HAR. By combining the interpretability and computational efficiency of traditional methods with the powerful learning capabilities of deep learning, our approach addresses real-world challenges such as variations in pose, scale, illumination, and occlusions. Although HoG+SVM is lightweight and interpretable, it struggles with large and complex data sets. In contrast, deep learning models excel in accuracy but require significant computational resources and labeled data. Our hybrid approach leverages the strengths of both techniques to provide a balanced and scalable solution for HAR applications.

LITERATURE REVIEW

Human Activity Recognition (HAR) has evolved significantly with advances in preprocessing, feature extraction, and classification techniques. Traditional approaches relied on hand-crafted features and classical machine learning models, but recent deep learning architectures have improved accuracy and adaptability. This section reviews key methodologies and situates our hybrid framework within the existing literature [11-25].

Preprocessing techniques are used to improve the quality of input data by removing noise, reducing dimensionality, and optimizing feature extraction. Various preprocessing techniques have been explored in HAR research to enhance the robustness of models. To prevent overfitting, Mekruksavanich et al. (2021) implemented 10-fold cross-validation, ensuring that the model generalizes well to unseen data. This method systematically partitions the dataset, training and validating on different subsets, improving overall reliability [11]. Ankita et al. (2021) applied a Gaussian blur in HAR tasks to smooth images, reduce high-frequency noise, and enhance feature clarity. This step is particularly useful for feature-based methods like Histogram of Oriented Gradients (HoG)[12]. Zhang et al. (2022) highlighted the benefits of augmenting training data by introducing transformations such as rotation, flipping, and scaling. This approach mitigates data scarcity issues and improves the generalization of deep learning models [13]. Xu et al. (2023) showed that grayscale conversion reduces computational complexity while preserving critical spatial information, making it a widely used preprocessing step in image-based HAR [14]. While these studies focus on individual preprocessing techniques, our approach integrates cross-validation, Gaussian blur, data augmentation, and grayscale conversion to maximize efficiency. This ensures that our hybrid model processes input data effectively, improving feature extraction and classification performance.

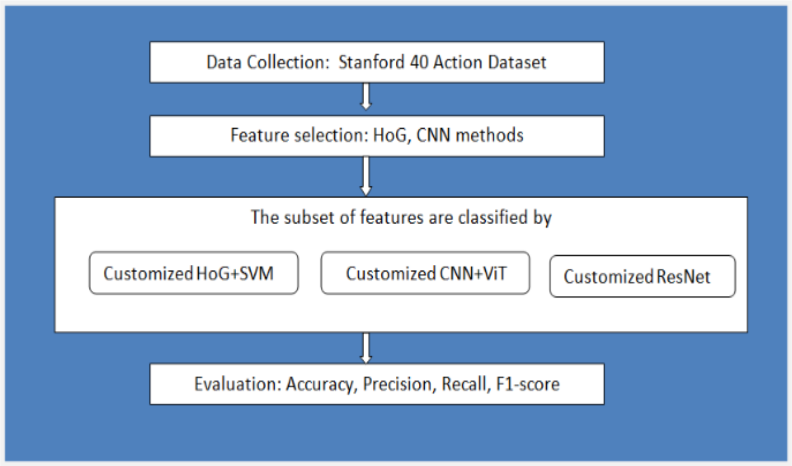
Feature extraction impacts the system’s ability to capture meaningful data. Khan et al. (2024) method leverages multiple viewpoints in dynamic environments. The study explores the effectiveness of this fusion in addressing challenges like occlusion and varying perspectives [15]. Dua et al (2021) effectively capture both spatial and temporal dependencies. Their approach demonstrated improved recognition accuracy by integrating CNN Gated Recurrent Units (GRU) for sequential pattern learning [16]. Muhammad et al. (2021) presented attention mechanisms with dilated convolutions that improve the accuracy [17]. Xiao et al. (2021) approach improves model accuracy by efficiently extracting relevant features while maintaining data privacy across decentralized devices. The study demonstrates the potential of federated learning in enhancing activity recognition performance in distributed sensor networks [18]. While these studies focus on deep learning-based methods like CNN-GRU, attention-based LSTM, and federated learning, which require high computational power, large datasets, and extensive training, our proposed HoG is computationally efficient and interpretable, and it effectively captures information related to edges and gradients.

Classification algorithms play a pivotal role in distinguishing between various human activities. Al-Qaness et al. (2022) enhance feature extraction by leveraging residual connections and attention layers, improving recognition accuracy. The approach effectively captures intricate spatial-temporal dependencies, making it suitable for complex activity patterns [19]. Wang et al (2019) used deep learning to extract hierarchical spatial features, making them effective for HAR tasks that require fine-grained recognition. Their convolutional layers are well-suited for processing visual data and capturing local patterns like edges and textures [20]. Hussain et al (2022) used Vision Transformers to employ global dependencies in input data. Unlike CNNs, which focus on local patterns, ViTs analyze the entire image context. This makes them particularly effective for HAR tasks involving relationships between distant regions in images. [21]. Trujillo et al (2023) integrate CNNs and ViTs to create a complementary system where CNNs handle local feature extraction, and ViTs provide global contextual understanding. This synergy ensures comprehensive activity recognition, improving performance compared to using either architecture alone [22]. Ronald et al (2021) used ResNet, and its ability to extract highly abstract features is particularly beneficial for recognizing subtle variations in activity patterns. They proposed iSPLInception, a deep learning architecture combining Inception and ResNet models for human activity recognition [23]. Tang et al. (2022) introduced a triple cross-domain attention mechanism for human activity recognition using wearable sensors. Their method focuses on improving model performance by effectively capturing and integrating information from multiple domains, such as spatial, temporal, and sensor data [24]. Tang et al. (2022) proposed a multiscale deep feature learning approach for human activity recognition using wearable sensors. Their method leverages multiple scales to capture diverse features from sensor data, improving recognition accuracy [25]. Muksimova et al. (2025) proposed a Cross-Modal Transformer-based approach for streaming dense video captioning; leveraging Neural ODE for a precise temporal method enhances video understanding by effectively capturing spatial-temporal dependencies.

Proposed HAR Model

In this study, an integrated approach of traditional and Deep Learning methods with enhanced Feature Extraction is proposed to handle the Human Activity Recognition system. Figure 1 summarizes our proposed model, starting with data collection. In this work, the Stanford 40 Action Dataset is collected, and its description is shown in Table A(appendix). Next phase, the important features are identified using the HOG approach and CNN. Further, the selected features from the HOG method are given to the traditional SVM classifier, and the selected features from the CNN method are given to the vision transformer to classify the human activity recognition. This hybrid method has three variants, such as SVM + HoG Approach, CNN+Vision Transformer, and Resnet Models.

Figure 1: Proposed HAR Framework


Data Collection: Stanford 40 Action Dataset
Feature selection: HoG, CNN methods
The subset of features are classified by
Customized HoG+SVM
Customized CNN+ViT
Customized ResNet
Evaluation: Accuracy, Precision, Recall, F1-score

Data Collection and Partitioning

We selected the Stanford 40 Actions Dataset [28] for this study due to its lower noise levels, as it contains fewer third-party elements in the background compared to other datasets, ensuring a clearer focus on the primary subject. Additionally, it offers a diverse set of 40 human actions with a well-balanced number of samples per category, making it suitable for robust and generalizable Human Activity Recognition (HAR) models. This dataset strikes a balance between real-world complexity and controlled variability, allowing for effective feature extraction and classification. The Dataset contains images of varying sizes. However, the average image size in this dataset is approximately 300 × 200 pixels, and it also has bounding boxes that specify the location of the person acting in each image. These annotations help in localizing the subject, enabling precise action recognition in HAR, and Figure 2 represents a few sample images from the dataset, which consists of 9532 images with 5,532 samples used for training and the remaining 4,000 samples reserved for testing.

Figure 2: A few sample images from the dataset

 Playing violin	 (b) Applauding

HoG-based Feature selection

Histogram of Oriented Gradients (HoG) is one of the widely used feature extraction techniques for human activity recognition. It effectively captures gradient and edge information, making it suitable for object detection and recognition tasks. The HoG descriptor works by computing the gradient orientations of an image and constructing histograms based on these orientations in localized regions of the image. The process of computing HoG features involves the following key steps:

Gradient Computation: To detect edges and changes in intensity, the image gradient is computed using derivative operators. Typically, the Sobel filter is used to approximate gradients in the horizontal and vertical directions.

Orientation Binning: The image is divided into small spatial regions known as cells (e.g., 8 × 8 pixels). Each pixel within a cell contributes to a histogram based on the orientation of its gradient. The orientation space is quantized into bins (typically 9 bins spanning 0◦ to 180◦). The contribution of each pixel to the histogram is weighted by its gradient magnitude. For a given cell with pixel indices i, j, the histogram bin corresponding to the gradient orientation is updated as follows:

$$H_b = H_b + G(i, j)$$

Block Normalization: To enhance robustness against illumination changes, normalization is applied over blocks (e.g., 16 × 16 pixels consisting of multiple 8 × 8 cells). This ensures that the feature representation is invariant to lighting and contrast variations. Let v be the feature vector of concatenated histograms from all cells in a block. The normalization is performed using one of the following methods, and this normalization step ensures that features remain stable under different illumination conditions.

L2-norm normalization:

$$v' = \frac{v}{\sqrt{\|v\|^2 + \epsilon^2}}$$

Where ϵ is a small constant which is used to prevent division by zero.

L1-norm normalization:

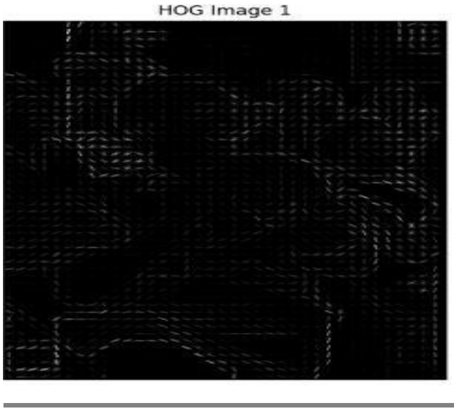
$$v' = \frac{v}{\sum |v| + \epsilon}$$

Final Feature Vector Representation: The final HoG feature vector for an image is obtained by concatenating the normalized histograms from all blocks:

$$F_{HoG} = [H_1, H_2 ... H_N]$$

Where H_i represents the normalized histogram from the ith block. Figure 3 represents the HoG techniques extracted image of Applauding action, and Algorithm 1 shows the pseudo-code for feature extraction steps in this study. These steps enhance the accuracy and efficiency of the recognition model. At the same time, this technique has another important drawback: it suffers from high sensitivity to changing pixel intensity.

Figure 3: HoG Image of Applauding action



Such factors, caused by environmental variation, illumination, or noise, significantly affect the accuracy of features extracted and later influence the classification models being developed. The inherent dependence of HOG on intensity gradients indicates the necessity for more robust techniques for feature extraction that can mitigate intensity variations.

CNN-based Feature selection:

CNNs constitute the essence of the feature extraction step. Their hierarchical structure, along with their ability to learn abstract representations, makes them perfect for extracting necessary features from preprocessed images. The process starts at the convolutional layers, where low-level features, such as edges, lines, and textures, are detected. Going deeper, the CNN continues to capture more complex, abstract patterns in the shape and structural relationship that depict human activity. It extracts hierarchical features by applying filters at different levels, and we used ReLU activation, Max pooling, and Soft Max for final classification.

Classification Techniques

Classification algorithms play a pivotal role in distinguishing between various human activities. Traditional methods and modern approaches like deep learning architectures (ResNet, CNN, and ViT) have been extensively explored.

HoG features given to SVM (HoG+SVM)

The application of HoG along with the SVM approach is very strong for HAR, ensuring the benefits of both. The workflow of HoG + SVM involves the following steps:

- Grayscale Conversion: Converts the image to grayscale, enhancing efficiency by lowering computational cost and minimizing noise.
- Feature Extraction (HoG): Calculates gradient magnitude and direction, normalizes the histogram, and concatenates the data to create a robust feature vector.
- Classification (SVM): Classifies activities by finding the optimal hyperplane that separates different activity classes.

Due to its ability to capture local spatial gradients, the HoG method, combined with SVM's capability to classify these features in a high-dimensional hyperplane, is effective. However, issues arise, particularly regarding sensitivity to intensity variations and the computational complexity when handling large datasets.

Algorithm 1 Feature Extraction Steps

```
procedure PREPROCESS_IMAGE (image)
    Convert to grayscale
    Apply Gaussian blur
    Resize to target size
    return preprocessed image
end procedure
procedure EXTRACT HOG FEATURES (image)
    Preprocessed_ Image ← Preprocess_ Image (image)
    Compute gradients of Preprocessed_Image
    Divide Preprocessed_Image into cells
    Compute HoG histograms
    return HoG features
end procedure
procedure EXTRACT_CNN FEATURES(image, cnn_model)
    Preprocessed_ Image ← Preprocess_image(image)
    Features ← CNN model. Extract features(Preprocessed Image )
    return features
end procedure
```

CNN features given to VIT

The application of CNN, along with the Visual Transformer approach, is very strong for HAR, ensuring the benefits of both. The proposed pipeline for CNN-ViT is structured as below,

- Grayscale Conversion: This simplifies the input, where only intensity variations will now be present, reducing any color noise.
- Gaussian Blur: A low-pass filter that filters off high-frequency noise, smoothing the image and promoting stabilization of the features, together with better extraction of features.
- CNN Feature Extraction: Extracts hierarchical features from preprocessed images, reduces dimensionality to preserve critical activity information.
- ViT Classification: Processes CNN features using attention mechanisms to learn global and local dependencies, classifying human activity with high accuracy

This attention mechanism is to focus on relevant regions of the image while ignoring irrelevant or noisy background details, as shown in Figure 4 for the image Applauding action.

Rationale for a Custom ResNet

The traditional ResNet models are deep networks that have been successful in a variety of image recognition tasks. These models typically consist of hundreds of layers and use pre-trained weights obtained from large datasets. However, such architectures can be too large for human activity recognition tasks, especially when the dataset is smaller or more specific. The complexity of using a full ResNet model is due to its size, which leads to high computational cost, longer training times, and potential overfitting when applied to smaller datasets. This will result in a lightweight ResNet model, as the network will be shallow and will not inherit the weights from the pre-trained ResNet, as shown in Algorithm 2. This custom ResNet reduces computational complexity by using fewer residual blocks while preserving the core identity mapping mechanism to enhance gradient flow during training. This makes it computationally lighter and focuses more on learning specific patterns in the HAR dataset rather than the general features present in pre-trained ResNets, as shown in Figure 5. The mathematical formulation of a residual block is as follows:

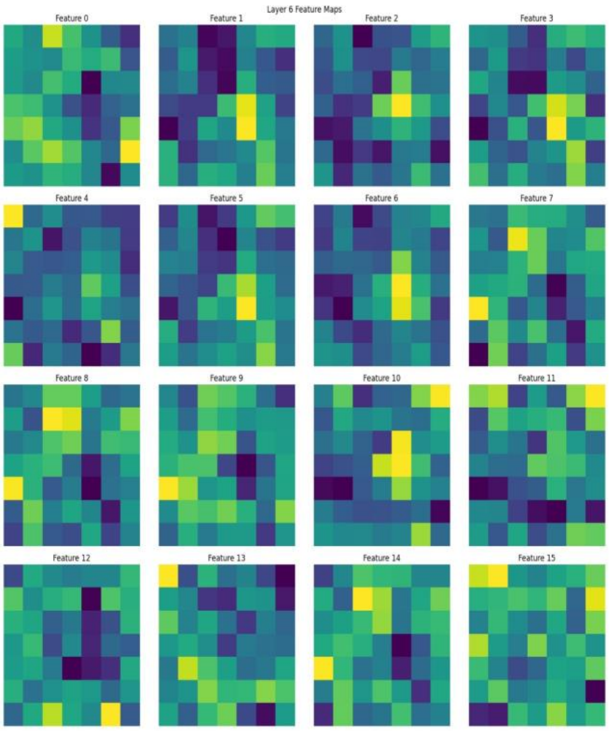


Figure 4: One of the layer weights of Applauding action using Attention mapping

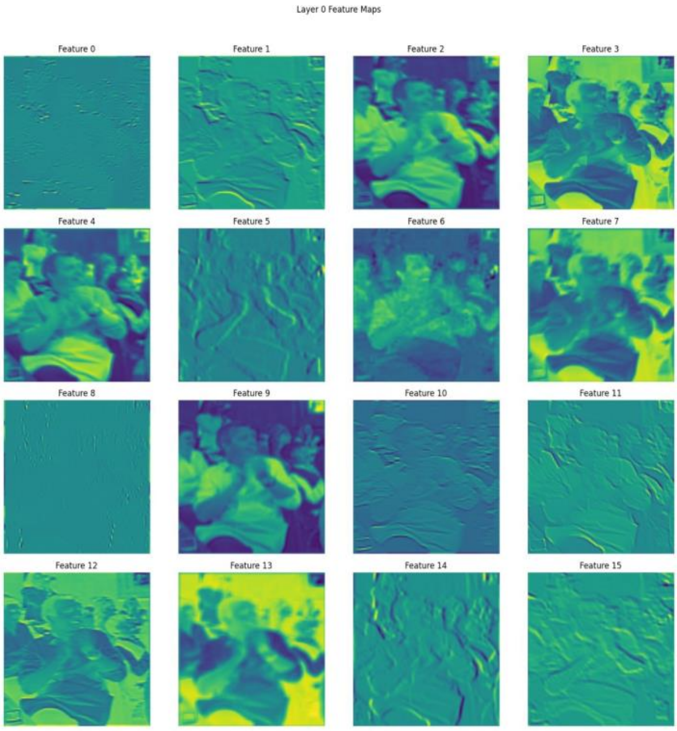


Figure 5: First Layer Feature map of applauding action using ResNet

Algorithm 2 Custom ResNet for HAR

```
Initialize System Configuration
Set GPU = NVIDIA Tesla V100
Set RAM $\geq$ 16 GB
Load Python Libraries: NumPy, Matplotlib, OpenCV, Torchvision
Select Framework: PyTorch
Select Dataset: Stanford 40 Action Dataset
Procedure BUILD RESNET(input size, num classes)
    Initialize Input Layer: (224,224,3) for RGB or (224,224,1) for grayscale
    For each residual block in range(3 to 4) do
        Apply convolutional layers with Batch Normalization and ReLU
        Add Skip Connection
    end for
    Flatten output and apply Fully Connected Layer (size = num classes)
    Apply Softmax activation
    return Model
end procedure
```

Performance Analysis and Result

Performance Measure

The proposed customized human activity recognition models are evaluated by following the performance measures. The confusion matrix can be used to visualize how well the model distinguishes between different activities, such as fixing a bike, looking through a telescope, phoning, throwing frisby, washing dishes, applauding, etc. Also, it can derive the overall classification Accuracy, Error rate, Precision, Recall, and F1-score.

Evaluation setup

In this study, four sets of experiments are conducted. Initially, a series of tests is carried out to assess the performance of HoG combined with individual classification models such as DT, SVM, KNN, NB, and RF. In the second phase, experiments focus on analyzing the hybridization behavior and effectiveness of these methods, incorporating CNN-based attribute selection with Vision Transformer. The third phase evaluates the performance of a customized ResNet model and its variants. Finally, the proposed techniques are compared against existing human activity recognition systems to assess their efficiency.

Result for setup I

In the first setup, we used HoG techniques for feature selection, which used a 3780-feature map. Table 1 depicts the performance of HoG along with the single classification models like DT, SVM, KNN, NB, and RF, on experimenting it using parameters like accuracy, Precision, Recall, and F1-score, and the corresponding confusion matrix table for the proposed HoG+SVM includes 40 action class labels represented in Tables B & C (appendix). **Result Summary:** Among all the classifiers considered, the SVM model could achieve an accuracy of 89.90% depending on the dataset's complexity, and outperforms all other classifiers in terms of accuracy and robustness to background noise. Figure 6 depicts the performance of HoG along with the single classification model in terms of accuracy.

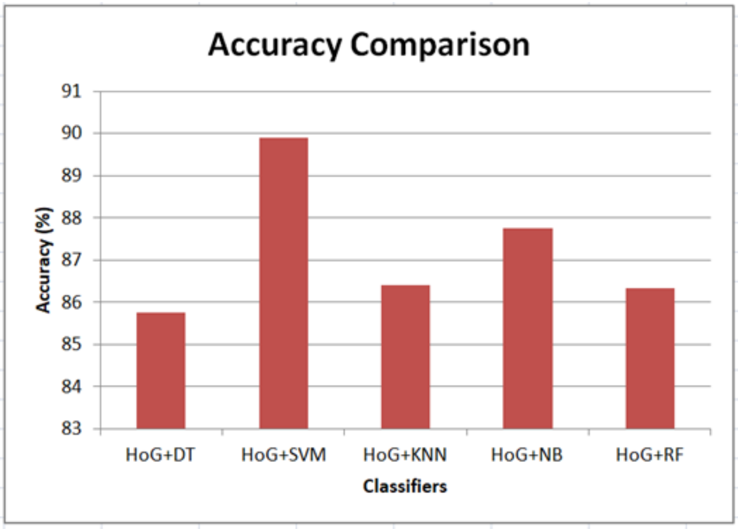
Table 1: Performance comparison between HoG along with Base classification

Classifier	HoG+DT	**HoG+SVM**	HoG+KNN	HoG+NB	HoG+RF
Accuracy	85.75	**89.80**	86.40	87.75	86.33
Error rate	14.25	**10.20**	13.60	12.25	13.67
Precision	88.42	**90.89**	87.64	88.77	88.52
Recall	88.84	**93.00**	91.08	92.04	89.76
F1-score	88.63	**91.93**	89.33	90.38	89.14

Table 2: Performance comparison between CNN

Classifier	**CNN+ViT**	CNN (ResNet 18)	ViT (Base)
Accuracy	**93.98**	91.55	88.58
Error rate	**06.03**	08.45	11.43
Precision	**94.31**	92.93	92.58
Recall	**96.16**	93.60	88.84
F1-score	**95.23**	93.26	90.67

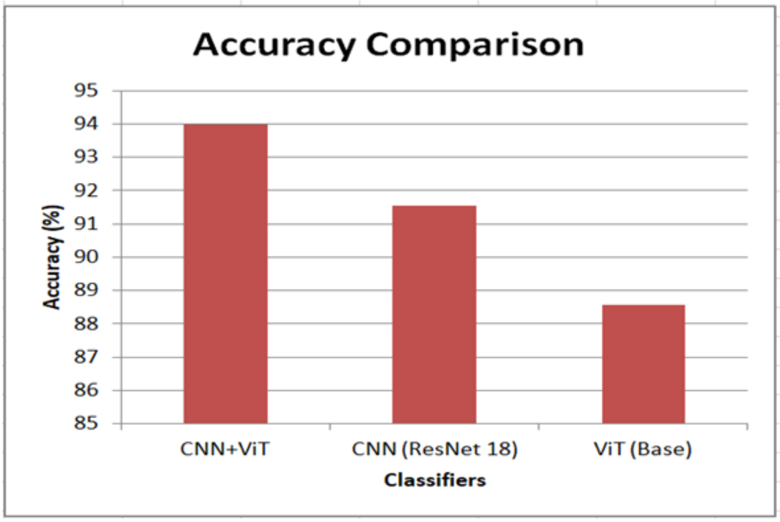
Figure 6: Accuracy comparison between HoG along with Base classificatio



Result for setup II

We used CNN techniques for feature selection, which used a 64 x 64-feature map followed by the Vision Transformer for classification Model building. Table 2 depicts the performance of CNN along with the Vision transformer classification model on experimenting it using parameters like accuracy, Precision, Recall, and F1-score. To compare the efficiency of the proposed model, we used two more classification models, such as CNN (ResNet 18) and ViT (Base). Figure 7 depicts the performance of CNN+ViT along with the CNN (ResNet 18) and ViT (Base) in terms of accuracy.

Figure 7: Accuracy comparison between CNN along ViT



Result Summary: Typically, the CNN + ViT model could achieve an accuracy of 93.98% and outperform both approaches individually in terms of accuracy, particularly when considering the CNN's ability to capture low-level features and ViT's ability to model global dependencies. However, ViTs generally provide robust performance on long-range dependencies, leading to higher precision and recall for complex activities. Validation loss over epochs to check for overfitting and convergence to expect a steady increase in accuracy, plateauing after some epochs, depending on the complexity of the dataset and model. Also, we checked inference time, and in real-time applications, inference time should be measured to assess the model's suitability for deployment. For instance, with a trained model, each image might take 50-200 ms depending on the complexity of the model and hardware.

along ViT

Result for setup III

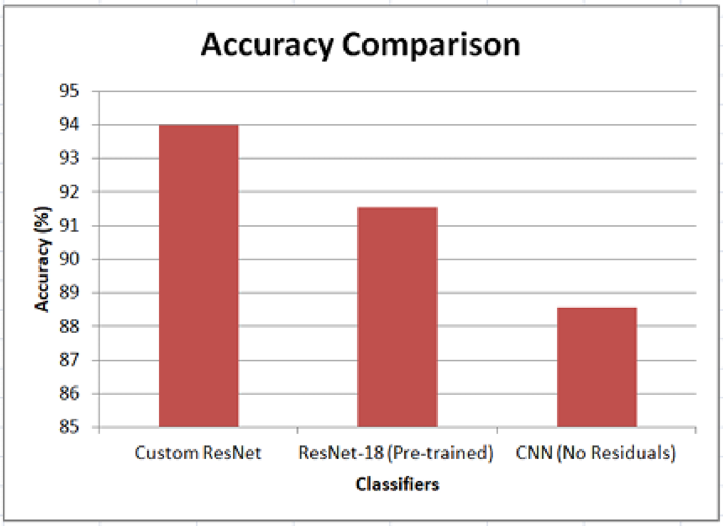
In this setup, we used a Custom ResNet for Human Activity Recognition classification Model. Table 3 depicts the performance of custom ResNet classification model on experimenting it using parameters like accuracy, Precision, Recall and F1-score. To compare the efficiency of the proposed model we used two more classification model such as ResNet-18 (Pre-trained) and CNN (No Residuals).

Table 3: Performance comparison between Custom ResNet and baseline model

Classifier	Custom ResNet	ResNet-18 (Pre-trained)	CNN (No Residuals)
Accuracy	**95.63**	91.55	87.02
Error rate	**04.38**	08.45	12.98
Precision	**96.89**	92.93	88.20
Recall	**96.08**	93.60	85.47
F1-score	**96.49**	93.26	86.81

Result Summary: Typically, the Custom ResNet model could achieve an accuracy of 95.63% and outperform both approaches individually in terms of accuracy. The custom ResNet achieves higher accuracy than a CNN model without residuals, showing the benefit of residual connections in deeper networks. Also, the custom ResNet has a lower inference time compared to ResNet-18, making it more efficient for real-time HAR applications, and despite fewer parameters, the custom ResNet generalizes well to unseen data due to the use of residual blocks and GAP. However, the custom ResNet model performs well in distinguishing activities with similar patterns, thanks to its ability to learn hierarchical features. The model converges within 50-60 epochs, indicating that the reduced depth is sufficient for the task. The validation accuracy plateaus after a few epochs, indicating that the model is generalizing well and not overfitting. Figure 8 depicts the performance of Custom ResNet along with the ResNet-18 (Pre-trained) and CNN (No Residuals) in terms of accuracy. However, Custom ResNet generally provides robust performance on long-range dependencies, leading to higher precision and recall for complex activities.

Figure 8: Accuracy comparison between the Custom ResNet and baseline model



Result for setup IV

In this setup, the efficiency of the proposed techniques is compared with existing techniques proposed by Lin et al (2021)[26] and Yao et al (2023) [27]. Table 4 depicts the performance of the proposed three classification models, such as HoG++SVM, CNN+ViT, and custom ResNet, compared with three existing classification models, such as EAPT (Efficient Attention Pyramid Transformer) [26], Pose Generation Network (PGN), and Pose Refinement Network (PRN) [27].

Table 4: Performance Comparison of three customized proposed models vs Existing models

Classifier	HoG+SVM	CNN+ViT	Custom ResNet	EAPT [31]	PGN [32]	PRN [32]
Accuracy	89.80	93.98	**95.63**	93.30	86.10	91.90
Error rate	10.20	06.03	**04.38**	06.70	13.90	08.10
Precision	90.89	94.31	**96.89**	95.85	89.24	97.76
Recall	93.00	96.16	**96.08**	93.32	88.88	90.12
F1-score	91.93	95.23	**96.49**	94.57	89.06	93.78

Lin et al. introduce EAPT, an efficient attention pyramid transformer designed to enhance image processing tasks. Their method optimizes attention mechanisms across hierarchical levels to capture both local and global image features effectively. The approach improves processing efficiency and accuracy, demonstrating significant advancements in transformer-based image analysis. This work provides valuable insights into scalable and high-performance image processing techniques [26]. Yao et al. present a transformer-based method for scene-aware human pose generation. Their approach integrates contextual scene information to generate realistic and contextually appropriate human poses. This method advances the field by achieving higher accuracy and naturalness in pose generation tasks. The work showcases the potential of transformers in enhancing scene-aware synthesis applications [27].

Result Summary: Typically, the Custom ResNet model could achieve an accuracy of 95.63% and outperform the existing approaches individually in terms of accuracy. Figure 9 shows the Accuracy Comparison of three customized proposed models vs existing models. The ROC curve compares the performance of six classifiers as shown in Figure 10, with Custom ResNet achieving the highest AUC (0.95), followed by CNN+ViT (0.91) and EAPT (0.90). HoG+SVM and PRN show moderate performance with an AUC of 0.86, while PGN performs the lowest at 0.81. A higher AUC indicates better classification ability, with curves closer to the top-left corner representing superior models.

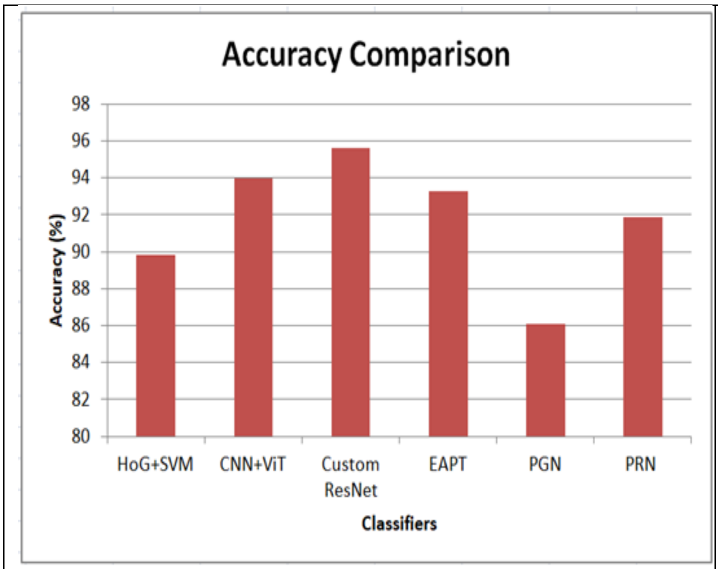


Figure 9: Accuracy Comparison of three customized proposed models vs Existing models

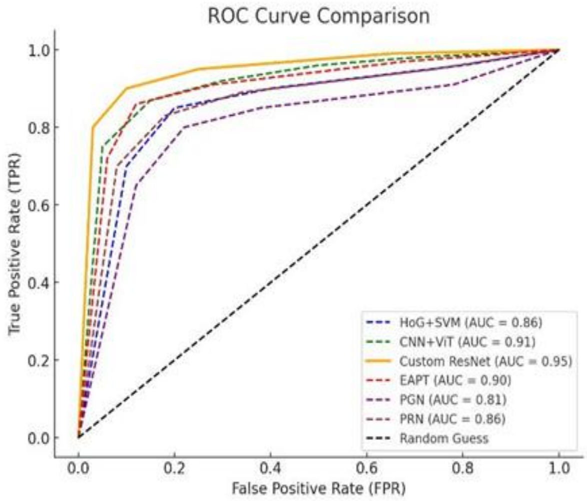


Figure 10: RoC Comparison

Practical Use Cases in Real-World Settings

The proposed hybrid HAR model can be effectively applied in several real-world domains:

- **Healthcare and Elderly Assistance:**Continuous monitoring of daily living activities, early detection of abnormal movements, and fall detection for elderly or post-operative patients.
- **Smart Home Automation:**Recognition of user activities to automatically adjust lighting, HVAC systems, or appliance control for energy optimization and enhanced comfort.
- **Workplace Safety:**Monitoring workers' movements to detect unsafe postures, fatigue patterns, or hazardous actions in construction, manufacturing, and mining industries.
- **Fitness and Sports Analytics:**Real-time activity tracking, exercise form correction, and personalized training feedback using wearable sensors.
- **IoT and Wearable Devices:**The hybrid design enables deployment on low-power edge devices, improving inference speed and reducing the need for continuous cloud communication.

CONCLUSION

In this project, we investigated advanced methodologies for Human Activity Recognition (HAR), combining Convolutional Neural Networks (CNNs), Vision Transformers (ViTs), and a custom ResNet model to address the challenges of computational efficiency, dataset limitations, and model complexity. Our preprocessing approach emphasized grayscale image conversion to reduce complexity and Gaussian Blur application to enhance feature stability. CNNs enabled hierarchical feature extraction, while ViTs contributed by capturing global image relationships through their attention mechanisms, improving the model’s robustness in scenarios with clutter or occlusion. A custom ResNet was developed to optimize performance on smaller datasets, ensuring computational efficiency by simplifying the architecture and excluding pre-trained weights while leveraging techniques such as batch normalization and global average pooling for enhanced generalization. Experiments revealed that the custom ResNet achieved superior accuracy, precision, recall, and inference speed compared to traditional CNNs and pre-trained ResNet models, demonstrating its suitability for real-time applications. Data augmentation strategies effectively mitigated overfitting, further supporting the model’s applicability to smaller datasets. The integration of these methodologies offers a versatile and adaptive solution for HAR, capable of handling diverse environments and activities, making it suitable for deployment in resource- limited settings such as mobile devices and embedded systems for applications like activity monitoring and health tracking. This work highlights the potential of combining CNNs, ViTs, and ResNet to create a robust system for HAR, providing a foundation for future research and practical applications.

Declarations

Author contribution Information

All authors are equally contributed.

Competing interests

There is no conflict of interest between the authors

Funding

The authors does not receive any funding.

Data Availability statement

Throughout the research used the public available data set which is cited in 28.

REFERENCES

1. Khan, Irfanullah, Antonio Guerrieri, Edoardo Serra, and Giandomenico Spezzano. ”A hybrid deep learning model for UWB radar-based human activity recognition.” Internet of Things 29 (2025): 101458.
2. Yadav, Santosh Kumar, Kamlesh Tiwari, Hari Mohan Pandey, and Shaik Ali Akbar. A review of multimodal human activity recognition with special emphasis on classification, applications, challenges and future directions. Knowledge- Based Systems 223 (2021): 106970.
3. Nawal, Yala, Mourad Oussalah, Belkacem Fer- gani, and Anthony Fleury. New incremental SVM algorithms for human activity recognition in smart homes. Journal of Ambient Intelli- gence and Humanized Computing 14, no. 10 (2023): 13433-13450.
4. Athavale, Vijay Anant, Suresh Chand Gupta, Deepak Kumar, and Savita Savita. Human action recognition using CNN-SVM model. Advances in Science and Technology 105 (2021): 282-290.
5. Patel, Chirag I., Dileep Labana, Sharnil Pandya, Kirit Modi, Hemant Ghayvat, and Muhammad Awais. Histogram of oriented gradient-based fusion of features for human action recognition in action video sequences. Sensors 20, no. 24 (2020): 7299.
6. Maheswari, B. Uma, R. Sonia, M. P. Rajaku- mar, and J. Ramya. Novel machine learning for human actions classification using histogram of oriented gradients and sparse representation. Information Technology and Control 50, no. 4 (2021): 686-705.
7. Khan, Zanobya N., and Jamil Ahmad. Atten- tion induced multi-head convolutional neu- ral network for human activity recognition. Applied soft computing 110 (2021): 107671.
8. Khan, Saad Irfan, Hussain Dawood, M. A. Khan, Ghassan F. Issa, Amir Hussain, Mrim M. Alnfiai, and Khan Muhammad Adnan. ”Transition-aware human activity recognition using an ensemble deep learning framework.” Computers in Human Behavior 162 (2025):108435.
9. Wensel, James, Hayat Ullah, and Arslan Munir. Vit-ret: Vision and recurrent transformer neu- ral networks for human activity recognition in videos. IEEE Access 11 (2023): 72227-72249.
10. Qu, Lele, Xiayang Li, Tianhong Yang, and Shuang Wang. ”Radar-Based Continuous Human Activity Recognition Using Multido main Fusion Vision Transformer.” IEEE Sen sors Journal (2025).
11. Mekruksavanich, Sakorn, and Anuchit Jitpat tanakul. Lstm networks using smartphone data for sensor-based human activity recognition in smart homes. Sensors 21, no. 5 (2021): 1636.
12. Ankita, Shalli Rani, Himanshi Babbar, Sonya Coleman, Aman Singh, and Hani Moaiteq Aljahdali. An efficient and lightweight deep learning model for human activity recognition using smartphones. Sensors 21, no. 11 (2021): 3845.
13. Zhang, Shibo, Yaxuan Li, Shen Zhang, Farzad Shahabi, Stephen Xia, Yu Deng, and Nabil Alshurafa. Deep learning in human activity recognition with wearable sensors: A review on advances. Sensors 22, no. 4 (2022): 1476.
14. Xu, Yang, and Ting Ting Qiu. Human activity recognition and embedded application based on convolutional neural network. Journal of Artifi cial Intelligence and Technology 1, no. 1 (2021): 51-60.
15. Khan, Muhammad Attique, Kashif Javed, Sajid Ali Khan, Tanzila Saba, Usman Habib, Junaid Ali Khan, and Aaqif Afzaal Abbasi. Human action recognition using fusion of mul tiview and deep features: an application to video surveillance. Multimedia tools and appli cations 83, no. 5 (2024): 14885-14911.
16. Dua, Nidhi, Shiva Nand Singh, and Vijay Bhaskar Semwal. Multi-input CNN-GRU based human activity recognition using wearable sen sors. Computing 103, no. 7 (2021): 1461-1478.
17. Muhammad, Khan, Amin Ullah, Ali Shariq Imran, Muhammad Sajjad, Mustafa Servet Kiran, Giovanna Sannino, and Victor Hugo C. de Albuquerque. Human action recogni tion using attention based LSTM network with dilated CNN features. Future Generation Com puter Systems 125 (2021): 820-830..
18. Xiao, Zhiwen, Xin Xu, Huanlai Xing, Fuhong Song, Xinhan Wang, and Bowen Zhao. A federated learning system with enhanced fea ture extraction for human activity recognition. Knowledge-Based Systems 229 (2021): 107338.
19. Al-Qaness, Mohammed AA, Abdelghani Dahou, Mohamed Abd Elaziz, and A. M. Helmi. ”Multi-ResAtt: Multilevel residual network with attention for human activity recognition using wearable sensors.” IEEE Transactions on Industrial Informatics 19, no. 1 (2022): 144-152.
20. Wang, Jindong, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. ”Deep learning for sensor-based activity recognition: A survey.” Pattern recognition letters 119 (2019): 3-11.

21. Hussain, Altaf, Tanveer Hussain, Waseem Ullah, and Sung Wook Baik. ”Vision trans former and deep sequence learning for human activity recognition in surveillance videos.” Computational Intelligence and Neuroscience 2022, no. 1 (2022): 3454167.
22. Trujillo-Guerrero, Mar´ıa Fernanda, Stadyn Rom´an-Niemes, Milagros Ja´en-Vargas, Alfonso Cadiz, Ricardo Fonseca, and Jos´e Javier Serrano-Olmedo. ”Accuracy comparison of CNN, LSTM, and transformer for activity recognition using IMU and visual markers.” IEEE Access 11 (2023): 106650-106669.
23. Ronald, Mutegeki, Alwin Poulose, and Dong Seog Han. ”iSPLInception: an inception ResNet deep learning architecture for human activity recognition.” IEEE Access 9 (2021): 68985-69001.
24. Tang, and Aiguo Song. ”Triple cross-domain attention on human activity recognition using wearable sensors.” IEEE Transactions on Emerging Topics in Computational Intelligence 6, no. 5 (2022): 1167-1176.
25. Tang, and Jun He. ”Multiscale deep fea ture learning for human activity recognition using wearable sensors.” IEEE Transactions on Industrial Electronics 70, no. 2 (2022): 2106 2116.
26. Lin, Xiao, et al. ”EAPT: efficient attention pyramid transformer for image processing.” IEEE Trans on Multimedia 25 (2021): 50-61.
27. Yao, Jieteng, et al. ”Scene-aware human pose generation using transformer.” Proceedings of the 31st ACM International Conference on Multimedia. 2023. http://vision.stanford.edu/Datasets/40actions.htm

APPENDIX

Table A -Data set description

Action	No of Images (Training)	No of Images (Testing)	Total Images
Fixing a bike	184 (64.8%)	100 (35.2%)	284
Looking through a telescope	159 (61.4%)	100 (38.6%)	259
Phoning	100 (50.0%)	100 (50.0%)	200
Throwing frisby	112 (52.8%)	100 (47.2%)	212
Washing dishes	195 (66.1%)	100 (33.9%)	295
Applauding	188 (65.3%)	100 (34.7%)	288
Playing guitar	103 (50.7%)	100 (49.3%)	203
Pushing a cart	89 (47.1%)	100 (52.9%)	189
Holding an umbrella	156 (60.9%)	100 (39.1%)	256
Blowing bubbles	187 (65.2%)	100 (34.8%)	287
Riding a horse	173 (63.4%)	100 (36.6%)	273
Climbing	128 (56.1%)	100 (43.9%)	228
Cooking	451 (60.2%)	100 (39.8%)	751
Writing on a book	99 (49.7%)	100 (50.3%)	199
Cleaning the floor	192 (65.8%)	100 (34.2%)	292
Reading	195 (66.1%)	100 (33.9%)	295
Smoking	91 (47.6%)	100 (52.4%)	191
Walking the dog	103 (50.7%)	100 (49.3%)	203
Rowing a boat	159 (61.4%)	100 (38.6%)	259
Fixing a car	189 (65.4%)	100 (34.6%)	289
Watching TV	160 (61.5%)	100 (38.5%)	260
Taking photos	100 (50.0%)	100 (50.0%)	200
Running	135 (57.4%)	100 (42.6%)	235
Cutting trees	145 (59.2%)	100 (40.8%)	245
Texting message	193 (65.9%)	100 (34.1%)	293
Drinking	196 (66.2%)	100 (33.8%)	296
Waving hands	85 (46.0%)	100 (54.0%)	185
Writing on a board	151 (60.2%)	100 (39.8%)	251
Jumping	114 (53.3%)	100 (46.7%)	214
Pouring liquid	141 (58.5%)	100 (41.5%)	241
Riding a bike	97 (49.2%)	100 (50.8%)	197
Shooting an arrow	93 (48.2%)	100 (51.8%)	193
Using a computer	102 (50.5%)	100 (49.5%)	202
Cutting vegetables	130 (56.5%)	100 (43.5%)	230
Fishing	193 (65.9%)	100 (34.1%)	293
Gardening	82 (45.1%)	100 (54.9%)	182
Feeding a horse	123 (55.2%)	100 (44.8%)	223
Playing violin	110 (52.4%)	100 (47.6%)	210
Brushing teeth	83 (45.4%)	100 (54.6%)	183
Looking through a telescope	146(59.3%)	100 (40.7%)	246

Table B: 40 × 20 [1-20 column] Confusion Matrix with proposed HoG+SVM model

Actual / Predicted	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
C1	94	0	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
C2	0	92	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	1
C3	0	0	91	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
C4	1	0	0	91	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
C5	0	0	0	0	90	0	0	1	0	1	0	0	0	0	0	1	0	1	0	0
C6	0	0	0	0	0	89	0	1	0	1	0	0	0	0	1	0	0	0	0	0
C7	1	1	0	1	0	1	88	0	1	0	0	0	1	0	0	1	0	0	0	0
C8	0	1	0	0	0	1	0	87	1	1	0	1	1	0	0	0	0	0	1	0
C9	0	1	0	0	0	0	1	1	87	0	0	0	0	0	0	0	0	0	0	1
C10	0	2	0	0	0	0	0	1	0	90	0	1	0	0	1	0	0	1	0	0
C11	0	1	1	0	2	0	0	2	0	1	88	0	0	0	0	0	0	0	0	0
C12	0	0	0	0	0	0	0	0	0	0	0	90	1	1	0	1	0	0	0	0
C13	0	0	1	0	0	0	1	0	1	1	0	2	87	1	0	0	0	0	0	0
C14	1	1	1	1	0	0	0	0	0	0	1	1	1	88	0	0	0	0	0	0
C15	0	0	0	0	0	0	0	1	0	1	0	1	0	0	93	1	0	1	0	0
C16	0	0	0	0	0	0	0	1	0	1	1	0	1	0	0	89	0	0	0	0
C17	1	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	91	0	0	0
C18	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	94	1	0
C19	0	0	0	0	0	1	0	0	1	0	1	0	0	1	1	1	0	1	86	0
C20	0	0	0	0	1	0	1	1	0	0	0	0	2	0	0	0	0	0	1	89
C21	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
C22	0	1	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0	0
C23	0	0	0	0	0	0	1	0	2	0	1	1	0	0	0	0	0	0	0	0
C24	1	0	1	0	0	1	0	0	0	0	0	0	0	1	0	1	1	0	0	0
C25	1	0	0	1	0	0	0	0	0	0	1	1	0	0	1	0	1	1	0	0
C26	0	0	1	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0	0
C27	0	0	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0
C28	0	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	1	0	1	0
C29	0	0	0	0	0	0	1	0	0	1	1	0	0	0	1	0	0	0	1	0
C30	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
C31	0	1	0	0	0	0	0	0	1	0	0	1	0	2	0	0	0	0	0	0
C32	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
C33	1	0	0	0	0	0	0	2	0	0	0	0	1	0	0	1	0	3	0	0
C34	0	0	1	0	1	1	0	1	2	0	0	1	0	0	0	0	0	0	0	0
C35	0	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	1	0	0	0
C36	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
C37	2	0	0	0	0	1	0	0	0	0	0	0	2	0	0	1	0	1	2	0
C38	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1
C39	0	1	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1
C40	0	0	0	0	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Table C: 40 × 20 [21-40 column] Confusion Matrix with proposed HoG+SVM model

Actual / Predicted	C21	C22	C23	C24	C25	C26	C27	C28	C29	C30	C31	C32	C33	C34	C35	C36	C37	C38	C39	C40
C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	1	0	0
C2	0	0	0	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
C3	0	2	1	0	0	1	0	0	2	0	1	0	0	0	0	0	0	0	0	0
C4	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	1	0	1	0
C5	0	1	0	0	0	1	0	1	0	1	1	0	0	0	1	0	0	0	0	0
C6	0	0	1	0	0	0	1	0	0	0	3	0	1	0	0	1	0	1	0	0
C7	0	0	0	0	0	0	0	1	0	0	1	1	0	0	1	0	1	0	0	0
C8	1	0	0	1	1	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0
C9	0	2	0	1	1	0	0	1	0	0	0	0	0	2	1	0	1	0	0	0
C10	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
C11	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
C12	0	0	0	1	1	1	1	0	0	0	0	0	2	0	0	0	0	1	0	0
C13	0	1	1	0	1	0	0	0	0	0	0	0	0	1	1	0	1	0	0	0
C14	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	1	1	0	0
C15	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
C16	0	0	1	0	0	1	0	0	2	0	0	1	1	0	0	0	0	0	0	1
C17	2	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	0	0	0	0
C18	0	0	0	0	0	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0
C19	1	0	2	1	0	0	0	1	0	1	0	0	1	0	0	0	0	0	0	0
C20	0	0	0	0	0	0	0	0	0	0	2	1	0	1	0	0	0	1	0	0
C21	93	1	1	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
C22	0	87	0	0	0	1	1	1	0	0	1	0	0	0	2	0	0	1	0	1
C23	1	0	93	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
C24	0	1	0	90	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	1
C25	0	0	1	0	88	1	0	0	0	0	0	0	2	0	0	0	1	0	0	0
C26	0	0	0	0	1	91	0	0	0	0	0	1	1	0	0	0	1	0	0	1
C27	0	0	0	0	0	1	92	0	0	1	0	0	0	0	0	0	0	0	0	2
C28	0	0	0	0	1	1	0	90	0	0	0	0	1	2	0	0	0	0	0	0
C29	0	0	0	0	1	1	0	0	89	0	0	0	2	0	0	0	1	1	0	0
C30	0	0	0	0	0	1	0	0	0	90	1	2	1	1	0	1	0	0	1	0
C31	1	1	0	0	0	0	0	0	0	0	89	2	0	0	0	0	0	0	0	2
C32	0	0	1	1	0	0	0	0	0	2	0	92	0	1	1	0	0	0	0	0
C33	0	0	0	0	1	1	0	1	0	1	0	0	85	1	0	1	0	0	1	0
C34	1	0	0	0	0	0	0	0	0	0	0	1	0	91	0	0	0	0	0	0
C35	0	1	0	0	0	0	0	0	1	0	0	0	0	0	94	0	0	0	0	0
C36	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	94	1	0	0	1
C37	1	0	1	0	0	1	1	0	0	1	0	0	0	0	0	0	86	0	0	0
C38	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	93	1	0
C39	0	1	0	0	0	0	0	0	1	2	0	0	0	1	0	4	0	1	85	0
C40	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	94