Turkish Journal of Engineering
https://dergipark.org.tr/en/pub/tuje
e-ISSN 2587-1366

A Comparative Study of Deep Learning Approaches for Human Action Recognition

Gulsum Yigit[*1]

[1]*Integration Solution Development, Huawei Turkey R&D Center, Istanbul, Turkey, gulsum.yigit@huawei.com*

Cite this study: Yigit, G. (2025). A Comparative Study of Deep Learning Approaches for Human Action Recognition. Turkish Journal of Engineering, 9 (2), 281-289.

https://doi.org/10.31127/tuje.1579795

Keywords
Deep Learning
Squeeze-and-Excitation Block
Residual Block
Zero-shot Learning
Human Action Recognition

Research Article

Received:05.11.2024
Revised:20.12.2024
Accepted:21.12.2024
Published:01.04.2025

Abstract

Human Action Recognition (HAR) plays a crucial role in understanding and categorizing human activities from visual data, with applications ranging from surveillance, healthcare to human-computer interaction. However, accurately recognizing a diverse range of actions remains challenging due to variations in appearance, occlusions, and complex motion patterns. This study investigates the effectiveness of various deep learning model architectures on HAR performance across a dataset encompassing 15 distinct action classes. Our evaluation examines three primary architectural approaches: baseline EfficientNet models, EfficientNet models augmented with Squeeze-and-Excitation (SE) blocks, and models combining SE blocks with Residual Networks. Our findings demonstrate that incorporating SE blocks consistently enhances classification accuracy across all tested models, underscoring the utility of channel attention mechanisms in refining feature representation for HAR tasks. Notably, the model architecture combining SE blocks with Residual Networks achieved the highest accuracy, increasing performance from 69.68% in baseline EfficientNet to 76.75%, marking a significant improvement. Additionally, alternative models, such as EfficientNet integrated with Support Vector Machines (EfficientNet-SVM) and ZeroShot Learning models, exhibit promising results, highlighting the adaptability and potential of diverse methodological approaches for addressing the complexities of HAR. These findings provide a foundation for future research in optimizing HAR systems, with implications for enhancing robustness and accuracy in action recognition applications.

1. Introduction

Recognizing and interpreting human actions, known as Human Action Recognition (HAR), is crucial for a range of practical applications. HAR can enhance autonomous navigation systems by identifying human behaviors to ensure safe operation [1] and can be applied in surveillance to detect potentially dangerous activities [2]. Additionally, HAR plays a significant role in human-robot interaction [3], health monitoring, sports analytics, home automation, fitness tracking, traffic management, augmented reality, and more [4].

Accurately identifying human actions from visual data is challenging due to factors like changing lighting conditions, background clutter, occlusions, and the wide range of human movements. Traditional methods largely relied on handcrafted features and shallow learning algorithms, which often failed to capture the complex patterns and subtleties of human actions.

Deep learning techniques, specifically Convolutional Neural Networks (CNNs) and Deep Belief Networks (DBNs), have revolutionized action recognition in computer vision by effectively reducing data dimensionality and improving classification accuracy [5-7]. Nevertheless, the traditional strategy of training deep learning models from scratch poses challenges due to its high need for labeled data, computational resources, and time. Researchers have increasingly shifted to transfer learning — transferring knowledge from pre-trained models to new tasks or domains. This approach capitalizes on learned representations to enhance model performance without requiring vast new data or resources.

Traditional handcrafted methods such as extended SURF [8], HOG-3D [9], etc., have shown notable effectiveness in HAR but are constrained by their dependence on manually created feature detectors and descriptors. In contrast, deep learning methods have gained prominence across diverse fields, including recognizing human activities. Recent emphasis has also moved towards deep learning approaches for action recognition, acquiring high accuracies on datasets like KTH and UCF sports [10, 11]. Despite the benefits of deep learning, such as improved discriminative ability and efficiency in capturing motion, these models necessitate

substantial amounts of domain-specific data for training, which can be expensive and time-consuming. To mitigate this challenge, transfer learning has arisen as a practical solution, leveraging pre-trained networks from datasets like ImageNet (e.g., AlexNet [12], GoogleNet [13], ResNet [14]). This strategy involves adjusting pre-trained models as feature extractors and incorporating deep representations with traditional classifiers to improve action recognition performance.

In HAR, CNNs have shown superiority over traditional methods like MLP, Naive Bayes, and SVM in differentiating between locomotion activities [15]. Studies have studied CNN architectures to analyze sensor locations' impact on activity recognition [16], and combinations of handcrafted and CNN-generated features have improved classification performance [17]. Significant advancements include 3D ConvNets [18], Convolutional RBMs [19], spatiotemporal learning using 3D ConvNets [20], Deep ConvNets and Two-stream ConvNets [21], which have showcased impressive performance.

Researchers have also designed several RNN-based (Recurrent Neural Networks) models to enhance HAR performance. Unlike earlier models that solely considered single-dimensional time-series inputs, a CNN + RNN model in [22] utilizes stacked multisensor data from each channel for fusion. Ketyk´o et al. [23] address domain adaptation problems using RNNs, while residual networks (ResNets) are favored for more accessible training due to efficient gradient flow through residual connections. RNNs are widely used in HAR as discriminative models, trained to minimize a cost function related to network outputs and labels. DRNNs, such as those in [24], have effectively recognized actions from various datasets.

Recent advancements in HAR have explored the use of Generative Adversarial Networks (GANs) and their variations to address the challenges of obtaining labeled data, which is both difficult and costly [25]. In HAR, GANs have been utilized to create synthetic sensor data that closely mimics real-world data, helping to alleviate issues like imbalanced training sets [26]. Wang et al. [26] demonstrated the effectiveness of GANs by generating artificial data from existing HAR datasets, thereby enhancing model robustness and performance. Their approach involved oversampling and incorporating synthetic sensor data into training sets, which helped balance the dataset and improve model accuracy [27, 28]. Moreover, GANs have been instrumental in transfer learning for HAR, enabling models to generalize better to unseen data from new users without extensive data collection efforts [29]. This process enables cross-subject transfer learning, where knowledge gained from one user can be applied to others.

This study comprehensively compares incorporating advanced methods in the HAR task. Leveraging the EfficientNetV2B3 model [30] pre-trained on the ImageNet dataset as our foundational backbone, we part from classic transfer learning and fine-tuning approaches by incorporating innovative techniques. We incorporate Squeeze-andExcitation (SE) and residual blocks employing transfer learning and fine-tuning strategies on EfficientNetV2B3. Moreover, this study introduces a hybrid model that extracts features from three pre-trained models and incorporates SE and residual blocks. This study also examines the feature extraction approach with EfficientNetV2B3 and SVM (Support Vector Machines) and Random Forest (RF) as classifiers and zero-shot learning techniques. We extensively assess HAR using these techniques.

This paper is structured as follows: In Section 2, we explain the methods utilized in this study. Our experimental findings are discussed in Section 3 and Section 4 includes discussion of the experimental results. Finally, Section 5 is the conclusion.

2. Method

2.1. Squeeze and Excitation (SE) Block

Squeeze and Excitation (SE) blocks enhance the power of a network by adaptively recalibrating channel-wise feature responses [31] (Figure 1a). The SE block contains three primary operations:

- Squeeze Operation: The squeeze operation aggregates global spatial information into a channel descriptor using global average pooling. For a feature map, this operation computes the average value of each channel across all spatial dimensions.
- Excitation Operation: The excitation operation finds a set of weights that emphasize the essential channels via a two-layer fully connected network with ReLU and sigmoid activations. This process generates a vector of weights that show each channel's significance.
- Scaling Operation: The final operation in the SE block is scaling the original feature map with the weights generated by the excitation module. This step effectively underscores the most important features while reducing the less important ones, thereby improving the network's capability.

2.2. The Residual and SE Blocks

Residual blocks, introduced in ResNet (Residual Networks) [14], help train deep networks by handling the vanishing gradient problem. A residual block consists of:

- A series of convolutional layers followed by batch normalization and ReLU activation.
- A shortcut (skip connection) that adds the input of the block to the output of the series of convolutional layers.

The addition of the shortcut helps the network to learn identity mappings, which makes it easier to optimize.

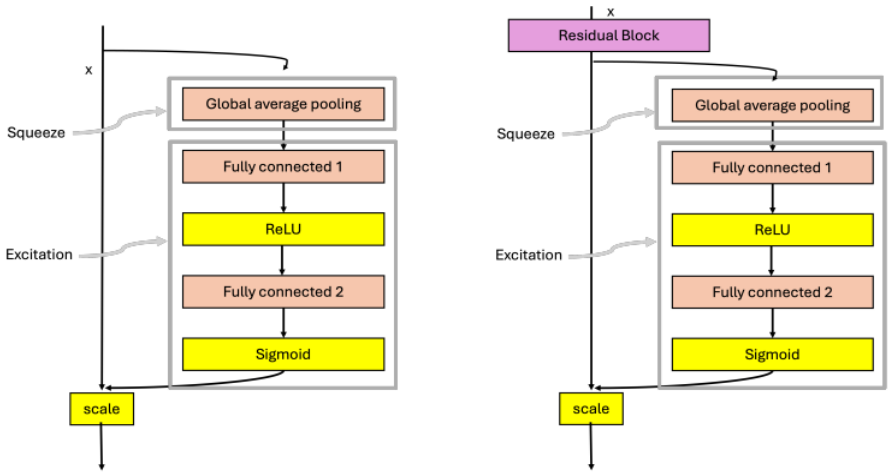

Figure 1. Proposed models with SE Block and Residual Block with SE Block

Figure 1b shows the SE block used to enhance the representational power of a neural network by recalibrating channel-wise feature responses. Beginning with the output from a Residual Block, the features passed to Global Average Pooling, which summarizes each feature map into a single value. This pooled output is then passed through two fully connected layers. The first layer, followed by a ReLU activation, decreases the dimensionality, while the second layer, followed by a Sigmoid activation, restores it, producing scaling factors in the range [0, 1]. These factors are used to scale the original feature maps, highlighting essential features and hiding the less important ones.

Residual Block
Input: base 1. Conv2D (256 filters, 1x1, stride 1, padding='same') 2. BatchNormalization 3. ReLU Activation 4. Conv2D (256 filters, 3x3, stride 1, padding='same') 5. BatchNormalization 6. ReLU Activation 7. Conv2D (1024 filters, 1x1, stride 1, padding='same') 8. Shortcut: Conv2D (1024 filters, 1x1, stride 1, padding='same') 9. BatchNormalization 10. BatchNormalization 11. Element-wise Addition (Residual Connection) 12. ReLU Activation

The block architecture above is designed to improve gradient propagation. Beginning with a 1x1 convolutional layer featuring 256 filters, stride 1, and 'same' padding, followed by batch normalization and ReLU activation, the block then includes a 3x3 convolutional layer with 256 filters and 'same' padding. Subsequently, another 1x1 convolutional layer with 1024 filters and 'same' padding is employed. Simultaneously, a shortcut is created with its own 1x1 convolutional layer to match the dimensions of the main path. Both paths undergo batch normalization, with the shortcut's output being element-wise added to the main path's output. This residual connection allows the network to learn residuals, thereby mitigating the vanishing gradient issue and promoting effective feature learning. The block concludes with a ReLU activation to introduce non-linearity. This structured approach effectively enhances the capability of CNNs to learn intricate features, thereby boosting performance across various tasks such as image classification and object detection.

2.3. Transfer Learning with EfficientNetV2B3

EfficientNetV2B3 is a CNN architecture known for its efficiency and robust performance across various computer vision tasks. It has been pre-trained on large-scale image datasets like ImageNet, where it learned to extract general features from images. EfficientNetV2B3 stands out in deep learning architectures for its efficient scaling and developed block designs. Created upon the principles of compound scaling, it uniformly adapts network depth, width, and resolution, optimizing computational resources while improving accuracy. The model incorporates inverted bottleneck blocks, which decompose convolutions into depthwise and pointwise operations, effectively lowering computational load.

2.3.1. Transfer Learning

Transfer learning is a technique where a model trained on one task is reused or transferred as a starting point for another task. It's advantageous when there is a limited amount of data. Therefore, rather than training an image classification model from scratch for HAR, we can leverage the pre-trained EfficientNetV2B3 model. This allows us to benefit from the learned representations of lower-level features (edges and textures) and higher-level features (shapes and patterns) that distinguish between different classes of human actions.

2.3.2. Fine Tuning

Fine-tuning a pre-trained model like EfficientNetV2B3 concerns adjusting its parameters to fit a specific dataset better, improving its capability to classify human actions in this scenario. Commonly, this procedure starts by unfreezing layers—here, 20 layers are unfrozen—to allow their weights to be updated during training. The model can adjust to the subtle features of the 15 labeled human action classes by concentrating on the top layers, where abstract features are encoded. This process leverages the general features learned from the original ImageNet pre-training in the lower layers, maintaining their ability to identify essential visual patterns while distilling the model's higher-level representations to determine more effectively among the diverse actions in the dataset.

We incorporate SE and residual blocks operating transfer learning and fine-tuning techniques on EfficientNetV2B3 to improve the model's performance. The SE blocks, which adaptively recalibrate channel-wise feature responses by explicitly modeling interdependencies between channels, are integrated to enhance the network's sensitivity to features. Meanwhile, the residual blocks, which enable gradient flow and mitigate the vanishing gradient problem through shortcut connections, enhance convergence and network depth.

This combination of SE and residual blocks, along with the sophisticated transfer learning and fine-tuning techniques, aims to create a robust and highly accurate model for HAR, balancing computational efficiency and predictive undertaking.

2.4. Hybrid Model Structure

The hybrid model designed for HAR incorporates the powers of three well-known models: VGG16, EfficientNetV2B3, and ResNet50. Each of these models has been pretrained on the ImageNet dataset, allowing them to extract significant features from images. Integrating multiple models lets the hybrid model leverage various feature representations, improving its capacity to distinguish subtle nuances in human activities given in the dataset.

In the structure of the hybrid model, each base model—VGG16, EfficientNetV2B3, and ResNet50—is initialized with weights from ImageNet and configured to exclude their top classification layers. This configuration lets the models concentrate only on feature extraction rather than classification, aligning them for HAR. The input images, resized to a standardized 224x224 pixels, undergo feature extraction independently within each base model.

After extracting features from the final convolutional layers of VGG16, EfficientNetV2B3, and ResNet50, the hybrid model combines these features via concatenation. This fusion procedure integrates each model's spatial hierarchies and learned filters, enhancing the overall feature representation. Combining these diverse features gives the hybrid model a more comprehensive understanding of the visual cues associated with different human actions, thereby enhancing its classification accuracy.

Following feature concatenation, the combined feature vector is passed through dense layers with rectified linear unit (ReLU) activation functions. These dense layers facilitate learning complicated relationships among the concatenated features, allowing the model to grasp intricate patterns relevant to HAR. Then, dropout regularization is involved after the dense layers to prevent overfitting by randomly deactivating a fraction of neurons during training, enabling better model generalization.

The output layer of the hybrid model utilizes a softmax activation function to yield a probability distribution across the 15 human action classes present in the dataset. This final layer ensures the model's predictions are normalized, providing a classification prediction for each input image.

2.4.1. SE Block Integration in Hybrid Model Architecture

In the hybrid model architecture, the SE block is applied to the output feature maps of each base pre-trained model. This integration recalibrates channel-wise responses, enhancing feature discriminability. After using the SE block, the improved feature maps are flattened and concatenated to form a unified feature vector. This concatenated feature vector combines improved representations from multiple architectures, enhancing the model's ability.

2.4.2. Residual and SE Blocks Integration in Hybrid Model Architecture

SE Blocks are incorporated into the model to improve the channel-wise feature responses from each base CNN model (VGG16, MobileNetV2, and ResNet50). SE Blocks aims to recalibrate channel-wise feature responses adaptively, focusing on more important features and concealing the less useful ones. This mechanism helps improve the model's discriminative power by emphasizing essential features.

Residual Blocks are introduced to catch and propagate deeper feature representations through the model. Residual connections alleviate the vanishing gradient problem. By propagating gradients more efficiently, Residual Blocks enable the learning of complex features across multiple layers. After passing through SE Blocks, the features from all three base models are concatenated.

SE Blocks improves feature representation by concentrating on informative channels. Residual Blocks enable deeper feature learning and gradient flow, enhancing the model's capability to grasp complex patterns. Concurrently, these components contribute to the usefulness and robustness of the hybrid model architecture, leveraging the power of both SE Blocks for channel recalibration and Residual Blocks for deep feature propagation.

Figure 2 demonstrates an architecture for a hybrid neural network model focusing on HAR. The procedure starts with an input image, shown by a person riding a

bicycle with a child. This image is then processed through three pre-trained CNNs: VGG16, EfficientNetV2B3, and ResNet50. Each network is well-regarded for its unique architecture and feature extraction capabilities. VGG16 is known for its deep but straightforward structure, EfficientNetV2B3 for its computational efficiency and performance, and ResNet50 for its ability to manage deep layers through residual learning.

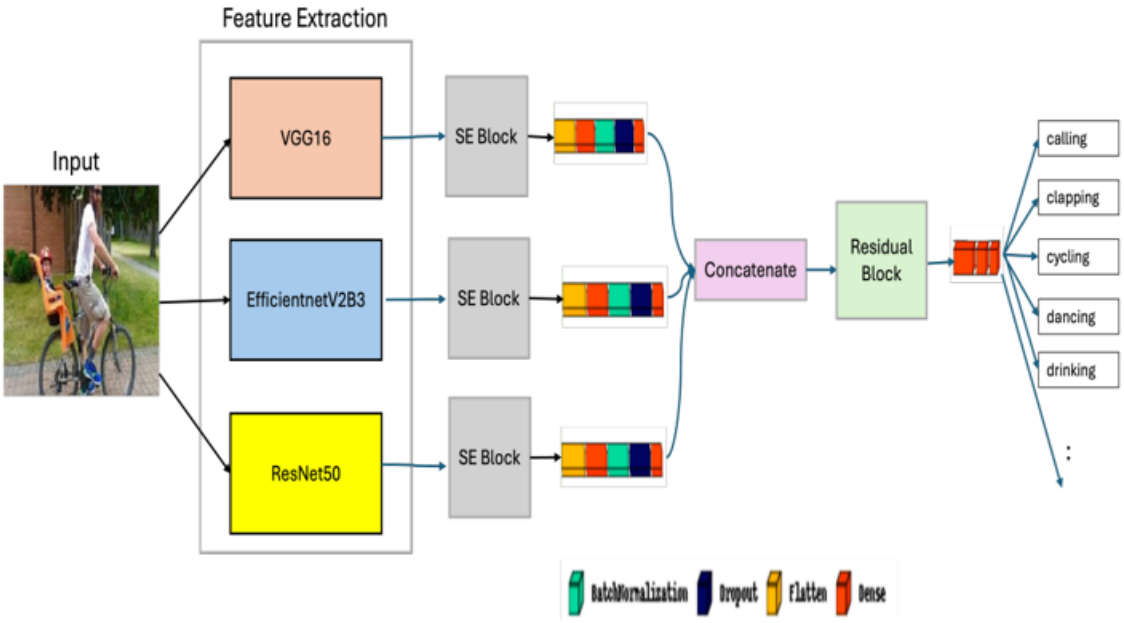

Figure 2. Hybrid Model

After these pre-trained models' initial feature extraction, the outputs are fed into SE Blocks. SE Blocks improve the quality of the extracted features by recalibrating channel-wise feature responses, effectively modeling the interdependencies between channels. This phase is crucial as it amplifies valuable features while reducing the less important ones, thus distilling the overall feature representation obtained from the CNNs.

Once the SE Blocks have processed the features, they are concatenated. This concatenation combines the diverse feature representations from the three networks into a wide feature vector. By incorporating features from multiple models, the architecture leverages the powers of each, showing a more thorough understanding of the input image.

The concatenated feature vector is then passed through a Residual Block. Residual Blocks are beneficial in deep neural networks as they help to mitigate the vanishing gradient problem. This block also distills the combined features, enabling the model to learn more complex and nuanced representations of the input data. Finally, the processed features pass through a series of fully connected layers, including BatchNormalization, Dropout, Flatten, and Dense layers. The output layer results in classification, indicating the recognized human action. In summary, this hybrid model architecture combines the powers of different architectures with developed feature processing techniques like SE Blocks and Residual Blocks.

2.4.3. Feature Extraction using EfficientNetV2B3, SVM and Random Forest as classifiers

The model combines feature extraction using a pre-trained EfficientNetV2B3 model and classification using SVM and RF. EfficientNetV2B3 is employed to extract features from input images. Then, a feature representation of the input images is acquired by removing the top classification layers of EfficientNetV2B3. These features grab hierarchical patterns learned by EfficientNetV2B3 during its training on ImageNet. The extracted features are then fed into an SVM classifier or Random Forest.

2.5. Zero-shot Learning

Zero-shot learning refers to the capability of a model to generalize to unseen classes without explicit training on those classes. Zero-shot learning with OpenAI's CLIP-ViT-Base-Patch32 model harnesses its capabilities. The model CLIP-ViT-BasePatch32 identifies the class that most matches the query without needing explicit training on that specific class. This approach enables effective classification across various categories, making it particularly useful in applications requiring adaptation to new concepts or rapid deployment without thorough training data.

Figure 3. Examples of different human activities from the dataset. Each figure represents a unique activity

3. Results

3.1. Dataset

Our experimentations used a dataset of over 12,000 labeled images sourced from Kaggle (Human Action Recognition dataset) [32]. These images were resized to a standardized 224x224 pixels format for consistency across the dataset. The dataset has 15 classes, each depicting different human activities such as calling, clapping, cycling, dancing, drinking, eating, fighting, hugging, laughing, listening to music, running, sitting, sleeping, texting, and using a laptop. Each activity is annotated with a score of 0 to 14, enabling accurate labeling for classification.

Figure 3 illustrates samples of human activities included in the dataset for training the HAR model. Each subfigure, tagged from (a) to (o), represents a specific activity, such as calling, clapping, and cycling. These images provide various scenarios to enhance the robustness and accuracy of the model in recognizing multiple human actions. We partitioned the dataset into a training set of 10,080 images and a testing set of 2,520 images. Notably, each class within the dataset includes an equal number of examples, assuring balanced representation across all activities as seen in Figure 4. This balanced distribution prevents biases during model training and evaluation.

3.2. Experimental Results

The model training is configured with the Adam optimizer and a learning rate 1e-3 to ensure efficient and adaptive learning. Categorical cross-entropy is the loss function appropriate for multi-class classification tasks, with accuracy as the evaluation metric. The training process spans 30 epochs to adequately learn from the data while mitigating overfitting risks.

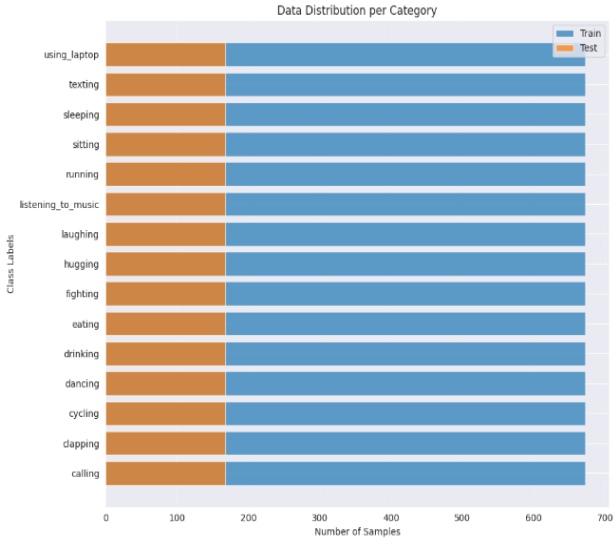

Figure 4. Data distribution per category

Table 1 presents the experimental results of various model configurations and methods applied to the HAR task. The "Base" column represents the performance of the base EfficientNetV2B3 model.

4. Discussion

Based on the experiments conducted, we can deduce the following outcomes:

- Base Model (EfficientNetV2B3): The baseline accuracy for transfer learning EfficientNet is 69.68%. When SE Blocks are added, the accuracy enhances greatly to 72.18%, and further increases to 75.79% with the addition of Residual Blocks.

• Fine-Tuning: Fine-tuning the EfficientNetV2B3 model results in a higher base accuracy of 73.06%. The accuracy improves to 75.08% with the addition of SE Blocks and further increases to 76.87% with the addition of Residual Blocks.

• Hybrid Model: The hybrid model, which incorporates multiple pre-trained models, also demonstrates significant progress with SE and Residual Blocks, reaching accuracies of 75% and 76.75%.

• EfficientNetV2B3-SVM and EfficientNetV2B3-RF: These methods use the EfficientNetV2B3 features with SVM and RF classifiers. The SVM classifier performs an accuracy of 74%, while the RF classifier has a lower accuracy of 58%.

• Zero-Shot Learning: This technique achieves a high accuracy of 75.19%, exhibiting its effectiveness in generalizing to new tasks without additional training.

Table 1 Experimental results

Method	Base	SE Block	Residual and SE Blocks
EfficientNet	69.68%	72.18%	75.79%
Fine Tuning	73.06%	75.08%	**76.87%**
Hybrid Model	73.02%	75%	**76.75%**
EfficientNet-SVM	74%	-	-
EfficientNet-RF	58%	-	-
Zero Shot Learning	75.19%	-	-

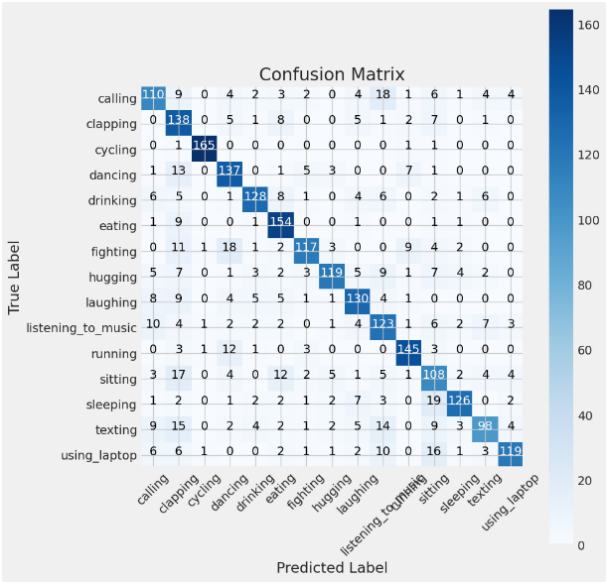

Figure 5. Confusion Matrix for hybrid model with SE and Residual Blocks

These results demonstrate the effectiveness of using SE Blocks and Residual Blocks in improving the performance of the EfficientNetV2B3 model and highlight the potential benefits of SE and residual blocks and hybrid model approaches. SE blocks dynamically emphasize important feature channels, while residual blocks enable deeper networks by ensuring effective feature propagation. Together, they allow HAR models to focus on salient features and capture complex action patterns, significantly improving classification performance. This integration during transfer learning or fine-tuning enhances the robustness and effectiveness of Human Action Recognition systems.

Figure 5 displays the confusion matrix for the hybrid model that includes both SE and residual blocks, offering a visual representation of the classification performance. It highlights the model's ability to accurately classify actions while identifying areas where misclassifications occur, providing insights into specific classes that may need further refinement in the model.

5. Conclusion

This study explored several advanced techniques to enhance HAR using deep learning models. Our focus centered on incorporating and examining the effectiveness of the SE and residual blocks within these models. A thorough analysis showed that the SE and residual blocks significantly improve feature representations' discriminative ability. This is accomplished by recalibrating channel-wise feature responses. This process applies by adjusting the importance of different channels in the feature maps based on their relevance to the task.

Furthermore, our research extended beyond the SE and residual blocks to include methodologies EfficientNetV2B3 for feature extraction and classifiers like SVM and RF. Moreover, our investigation into zero-shot learning highlighted its potential for extending the scope of HAR beyond trained classes from Open AI's model CLIP-ViTBase-Patch32.

In conclusion, our study highlights the importance of incorporating advanced techniques like the SE and residual blocks and leveraging state-of-the-art models to improve HAR tasks.

Funding

This research received no external funding

Author contributions

Gulsum Yigit: Research, Methodology, Visualization, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

1. Lu, M., Hu, Y., & Lu, X. (2020). Driver action recognition using deformable and dilated faster r-cnn with optimized region proposals. Applied Intelligence, 50(4), 1100–1111. https://doi.org/10.1007/s10489-019-01603-4.
2. Lin, W., Sun, M.-T., Poovandran, R., & Zhang, Z. (2008). Human activity recognition for video surveillance. In Proceedings of the 2008 IEEE International Symposium on Circuits and Systems (ISCAS), 2737–2740. IEEE. https://doi.org/10.1109/ISCAS.2008.4542023.

3. Rodomagoulakis, I., Kardaris, N., Pitsikalis, V., Mavroudi, E., Katsamanis, A., Tsiami, A., & Maragos, P. (2016). Multimodal human action recognition in assistive human-robot interaction. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2702–2706. IEEE. https://doi.org/10.1109/ICASSP.2016.7472168
4. Dentamaro, V., Gattulli, V., Impedovo, D., & Manca, F. (2024). Human activity recognition with smartphone-integrated sensors: A survey. Expert Systems with Applications, 123, 143. https://doi.org/10.1016/j.eswa.2024.123143
5. Aydın, V. A. (2024). Comparison of CNN-based methods for yoga pose classification. Turkish Journal of Engineering, 8(1), 65–75. https://doi.org/10.31127/tuje.1275826
6. Gülgün, O. D., & Erol, H. (2020). Classification performance comparisons of deep learning models in pneumonia diagnosis using chest x-ray images. Turkish Journal of Engineering, 4(3), 129–141. https://doi.org/10.31127/tuje.652358
7. Polater, S. N., & Sevli, O. (2024). Deep learning based classification for Alzheimer's disease detection using MRI images. Turkish Journal of Engineering, 8(4), 729–740. https://doi.org/10.31127/tuje.1434866.
8. Willems, G., Tuytelaars, T., & Van Gool, L. (2008). An efficient dense and scale-invariant spatio-temporal interest point detector. In Computer Vision–ECCV 2008: 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part II (Vol. 10, pp. 650–663). Springer. https://doi.org/10.1007/978-3-540-88688-4_48.
9. Klaser, A., Marszalek, M., & Schmid, C. (2008). A spatio-temporal descriptor based on 3d-gradients. In BMVC 2008-19th British Machine Vision Conference (pp. 275–1). British Machine Vision Association. https://doi.org/10.5244/C.22.99
10. Bux, A., Angelov, P., & Habib, Z. (2017). Vision based human activity recognition: A review. In Advances in Computational Intelligence Systems: Contributions Presented at the 16th UK Workshop on Computational Intelligence, September 7–9, 2016, Lancaster, UK (pp. 341–371). Springer. https://doi.org/10.1007/978-3-319-46562-3_23.
11. Charalampous, K., & Gasteratos, A. (2016). Online deep learning method for action recognition. Pattern Analysis and Applications, 19, 337–354. https://doi.org/10.1007/s10044-014-0404-8.
12. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. Advances in Neural Information Processing Systems, 25, 1097–1105. https://doi.org/10.1145/3065386.
13. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1–9). https://doi.org/ 10.1109/CVPR.2015.7298594.
14. He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 770–778). https://doi.org/10.1109/CVPR.2016.90.
15. Ronao, C. A., & Cho, S.-B. (2016). Human activity recognition with smartphone sensors using deep learning neural networks. Expert Systems with Applications, 59, 235–244. https://doi.org/10.1016/j.eswa.2016.04.032.
16. Hughes, D., & Correll, N. (2018). Distributed convolutional neural networks for human activity recognition in wearable robotics. In Distributed Autonomous Robotic Systems: The 13th International Symposium (pp. 619–631). Springer. https://doi.org/10.1007/978-3-319-73008-0_43
17. Dong, M., Han, J., He, Y., & Jing, X. (2019). Har-net: Fusing deep representation and hand-crafted features for human activity recognition. In Signal and Information Processing, Networking and Computers: Proceedings of the 5th International Conference on Signal and Information Processing, Networking and Computers (ICSINC) (pp. 32–40). Springer. https://doi.org/10.1007/978-981-13-7123-3_4
18. Ji, S., Xu, W., Yang, M., & Yu, K. (2012). 3d convolutional neural networks for human action recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 35(1), 221–231. https://doi.org/ 10.1109/TPAMI.2012.59
19. Taylor, G. W., Fergus, R., LeCun, Y., & Bregler, C. (2010). Convolutional learning of spatio-temporal features. In Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part VI (Vol. 11, pp. 140–153). Springer. https://doi.org/10.1007/978-3-642-15567-3_11.
20. Tran, D., Bourdev, L., Fergus, R., Torresani, L., & Paluri, M. (2015). Learning spatiotemporal features with 3d convolutional networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4489–4497. https://doi.org/10.1109/ICCV.2015.510
21. Simonyan, K., & Zisserman, A. (2014). Two-stream convolutional networks for action recognition in videos. Advances in Neural Information Processing Systems, 27. https://doi.org/10.48550/arXiv.1406.2199.
22. Lv, M., Xu, W., & Chen, T. (2019). A hybrid deep convolutional and recurrent neural network for complex activity recognition using multimodal sensors. Neurocomputing, 362, 33–40. https://doi.org/10.1016/j.neucom.2019.06.051
23. Ketykó, I., Kovács, F., & Varga, K. Z. (2019). Domain adaptation for sEMG-based gesture recognition with recurrent neural networks. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), 1–7. IEEE. https://doi.org/10.1109/IJCNN.2019.8852018
24. Inoue, M., Inoue, S., & Nishida, T. (2018). Deep recurrent neural network for mobile human activity

recognition with high throughput. Artificial Life and Robotics, 23, 173–185. https://doi.org/10.1007/s10015-017-0422-x
25. Shi, J., Zuo, D., & Zhang, Z. (2021). A GAN-based data augmentation method for human activity recognition via the caching ability. Internet Technology Letters, 4(5), 257. https://doi.org/10.1002/itl2.257.
26. Wang, J., Chen, Y., Gu, Y., Xiao, Y., & Pan, H. (2018). SensoryGANS: An effective generative adversarial framework for sensor-based human activity recognition. In Proceedings of the 2018 International Joint Conference on Neural Networks (IJCNN) (pp. 1–8). IEEE. https://doi.org/10.1109/IJCNN.2018.8489106.
27. Chan, M. H., & Noor, M. H. M. (2021). A unified generative model using generative adversarial network for activity recognition. Journal of Ambient Intelligence and Humanized Computing, 12(7), 8119–8128. https://doi.org/10.1007/s12652-020-02548-0.
28. Li, X., Luo, J., & Younes, R. (2020). ActivityGAN: Generative adversarial networks for data augmentation in sensor-based human activity recognition. In Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers (pp. 249–254). https://doi.org/10.1145/3410530.3414367.
29. Soleimani, E., & Nazerfard, E. (2021). Cross-subject transfer learning in human activity recognition systems using generative adversarial networks. Neurocomputing, 426, 26–34. https://doi.org/10.1016/j.neucom.2020.10.056.
30. Tan, M., & Le, Q. (2021). Efficientnetv2: Smaller models and faster training. In Proceedings of the International Conference on Machine Learning, 10096–10106. PMLR. https://doi.org/10.48550/arXiv.2104.00298
31. Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 7132–7141). https://doi.org/10.1109/CVPR.2018.00745
32. Kaggle. (2024). Human action recognition dataset. Kaggle. https://www.kaggle.com/datasets/meetnagadia/human-action-recognition-har-dataset