Deep Learning-Based Touch-Event Recognition from Individual RGB Frames: A Comparative Study of CNN Architectures and Model Training Strategies

Arash Ghazvini[1*], Irwandi Hipiny[2], Mazdan Ali Amaran[3],
Hasnizam Abdul Wahid[4], Johari Abdullah[5], Hamimah Ujir[6]
Faculty of Computer Science and Information Technology,
Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia[1, 5]
Faculty of Computing and Software Engineering, i-CATS University College,
Jalan Stampin Timur, 93350 Kuching, Sarawak, Malaysia[2, 6]
Faculty of Economics and Business, Universiti Malaysia Sarawak,94300 Kota Samarahan, Sarawak, Malaysia[3]
Institute of Creative Arts and Technology (iCreaTe),
Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia[4, 5]

***Abstract*—Touch-event recognition from visual data remains a challenging computer vision task, particularly when distinguishing the body regions involved in physical interactions. Unlike video-based action recognition, frame-level touch-event recognition requires models to identify interaction classes from individual RGB images without relying on temporal motion information. However, the effectiveness of different convolutional neural network (CNN) architectures and training strategies for this specific recognition task requires further investigation. This study presents a comparative evaluation of six CNN architectures: ResNet50, ResNet18, EfficientNet-B0, MobileNetV3-Small, a ResNet18-based spatial-attention model, and a Simple CNN. The experiments use 815 unique RGB frames belonging to four classes: No Touch, Touch Head, Touch Hips, and Touch Shoulders. Five training strategies are investigated, including baseline training, targeted data augmentation, class-weighted cross-entropy loss, a combination of targeted augmentation and class-weighted loss, and weighted random sampling. To provide a consistent evaluation, five seeded participant-video-group partitions are employed, with model selection based on validation macro F1. Performance is assessed using accuracy, balanced accuracy, and macro F1. The results indicate that ResNet18 with class-weighted loss achieved the highest mean test macro F1 of 0.344, while the performance of training strategies varied across CNN architectures. These findings provide a comparative baseline for evaluating the influence of CNN architecture selection and training strategies on frame-level body-region touch-event recognition within the investigated dataset.**

***Keywords*—Touch-event recognition; human interaction recognition; deep learning; convolutional neural networks; RGB frames; class-imbalanced learning**

I. Introduction

Human activity and interaction recognition have received considerable attention in computer vision, with research addressing body-pose analysis, physical contact, and the recognition of human interactions in images and videos [1], [2], [3]. These research areas focus on interpreting visual information to identify human activities and interactions [1], [3], [4]. While video-based recognition can exploit information from consecutive frames to capture movement and temporal relationships, image-based recognition relies on the visual information available in an individual frame [1], [4]. Consequently, recognizing physical interactions from a single RGB image can be challenging, particularly when the task requires identifying the specific body region involved in an interaction [1], [2], [5]. This makes the recognition of visible touch events from individual images a distinct image-classification problem [1], [5].

Convolutional neural networks (CNNs) have been widely applied to image-classification tasks because they can learn hierarchical visual features directly from image data [6], [7], [8]. Different architectures, such as ResNet [7] and EfficientNet [9], employ different approaches to feature extraction and representation [10]. However, their classification performance may vary depending on the characteristics of the task and the distribution of the training data [6], [11], [12]. One particular challenge is class imbalance, in which some classes contain fewer training samples than others [6], [11], [13], [14]. Under these conditions, a model may become biased towards more frequently represented classes, making it difficult to achieve consistent recognition performance across all categories [6], [11], [13], [14]. Consequently, accuracy alone may not provide a sufficient assessment of classification performance, making metrics such as macro F1 and balanced accuracy important for evaluating models on imbalanced datasets [11], [12], [15], [16].

Several training strategies can be employed to address class imbalance in image classification. Data augmentation increases the diversity of training samples, while class-weighted cross-entropy loss assigns different weights to classes to account for differences in their representation [6], [11], [13], [14]. Weighted random sampling provides another approach by adjusting the probability of selecting samples from different classes during training [11], [12], [14]. Although these strategies address class imbalance in different ways, their effectiveness may depend on the architecture and characteristics of the dataset [6], [12], [14]. In addition, spatial-attention mechanisms can be incorporated into CNN architectures to emphasize relevant spatial regions

*Corresponding author.

during feature learning [2], [17]. However, the relative effectiveness of different CNN architectures, training strategies, and spatial attention for frame-level body-region touch-event recognition requires further investigation [1], [5]. A systematic comparison under consistent evaluation conditions can help examine how these approaches perform in this specific classification setting [1], [11], [14].

The images used in this study were selected from the TikTok dance video dataset introduced by Hipiny et al. [18], which was originally developed for ranked dance video analysis rather than touch-event recognition. Selected frames from this dataset were used to formulate a four-class image-classification task comprising 815 unique RGB frames. The task involves identifying visible touch events associated with the head, hips, and shoulders, using four class labels: No Touch, Touch Head, Touch Hips, and Touch Shoulders. Each image is classified independently from a single RGB frame, without using temporal information or motion between consecutive frames. The resulting dataset exhibits class imbalance, providing a suitable setting for examining different training strategies and their effectiveness across the four classes.

Therefore, this study aims to conduct a comparative evaluation of six CNN architectures: ResNet50 [7], ResNet18 [7], EfficientNet-B0 [9], MobileNetV3-Small [19], a ResNet18-based spatial-attention model [17], and a Simple CNN [8]. Five training strategies are investigated: baseline training, targeted data augmentation, class-weighted cross-entropy loss, a combination of targeted augmentation and class-weighted loss, and weighted random sampling. The evaluation is conducted using five seeded participant-video-group partitions, with model selection based on validation macro F1. In addition, the ResNet18-based spatial-attention model is compared with the standard ResNet18 using matched split and initialization seeds. By evaluating these architectures and training strategies under a consistent experimental protocol, this study provides a comparative assessment of their effectiveness for frame-level body-region touch-event recognition within the investigated dataset.

II. Related Work

This makes the recognition of visible touch events from individual images a distinct image-classification problem [1], [5]. This section reviews previous research related to touch-event recognition, human-interaction analysis, and deep learning-based image classification. It first discusses existing approaches to physical-contact and human-interaction recognition, followed by CNN architectures, attention mechanisms, and training strategies for addressing class imbalance. The relevant studies are also compared in terms of their research focus, methods, and contributions. Finally, the limitations of existing approaches are discussed to establish the motivation for the present study.

A. Touch-Event and Human-Interaction Recognition

Research on touch-event recognition and physical body contact has investigated different visual cues for identifying interactions between people. Chen et al. [5] studied touch-event recognition in caregiver-infant videos by combining color and motion information with hand tracking and contour analysis to identify periods of physical contact. Their approach demonstrates how multiple visual cues can be combined to recognize contact events in video data. Similarly, Müller et al. [20] examined human pose in relation to body contact, while Narasimhaswamy et al. [2] investigated hand detection and physical-contact recognition in unconstrained images. These studies address different aspects of physical-contact recognition, particularly the use of body configuration and hand-related information.

Another approach to touch recognition was introduced by Ibrahim and Lyons [21] through FaceTouch, which applies supervised contrastive learning to hand-to-face touch detection. Unlike approaches that consider general physical contact, this work focuses on a specific interaction involving the face. Human interactions have also been investigated using still-image classification. Jiménez-Velasco et al. [1] proposed Proxemics-Net++ to classify human interactions in still images using visual and pose-related information. This study is particularly relevant to the present work because it considers interaction recognition using individual images rather than relying exclusively on video sequences. However, general interaction classification and the identification of specific body regions involved in touch events represent different recognition objectives.

Broader human activity recognition research has also explored different input modalities and feature representations. Zhang and Wang [22] surveyed RGB-D-based human action recognition, while Kumar et al. [23] and Kaur et al. [24] reviewed deep-learning and broader approaches to human activity recognition, respectively. Shafizadegan et al. [3] examined multimodal vision-based action recognition, whereas Azhar et al. [4] investigated multimodal feature extraction using a bidirectional LSTM for human-interaction recognition. These studies demonstrate the use of different visual modalities and temporal representations in activity and interaction analysis. Nevertheless, approaches that use multiple modalities or temporal information differ from the present task, in which each RGB frame is classified independently without information from consecutive frames. The main characteristics of these recognition studies, including their research focus, methods, and relevance to the present work, are summarized in Table I.

B. CNN Architectures and Training Strategies for Image Classification

Convolutional neural networks (CNNs) have become an important approach to image classification because they learn hierarchical visual features directly from image data. LeCun et al. [8] presented an early CNN-based approach to image recognition, providing a foundation for subsequent developments in deep convolutional architectures. Among these developments, ResNet [7] introduced residual connections to facilitate the training of deeper networks. EfficientNet [9] proposed a compound scaling strategy to balance network depth, width, and resolution, while MobileNetV3 [19] introduced efficient architectural designs for resource-constrained applications. These architectures provide different approaches to feature extraction and computational efficiency, making their comparative evaluation relevant to image-classification tasks.

TABLE I. COMPARISON OF TOUCH-EVENT AND HUMAN-INTERACTION RECOGNITION STUDIES

Research focus	Method/approach	Main contribution	Reference
Touch-event recognition	Color and motion cues, hand tracking, and contour analysis	Identifies periods of physical contact in caregiver-infant videos	[5]
Body contact and human pose	Human-pose analysis	Examines the relationship between body pose and contact	[20]
Physical-contact recognition	Hand detection and visual analysis in unconstrained images	Investigates contact recognition in unconstrained visual settings	[2]
Hand-to-face touch detection	FaceTouch and supervised contrastive learning	Addresses recognition of hand-to-face touch events	[21]
Human-interaction recognition	Proxemics-Net++ using visual and pose-related information	Classifies interaction categories in still images	[1]
Human activity recognition	RGB-D and deep-learning approaches; literature surveys	Provide an overview of activity-recognition methods and applications	[22], [23], [24]
Multimodal activity and interaction recognition	Multimodal feature extraction and vision-based recognition	Explore the use of multiple modalities for activity and interaction analysis	[3], [4]

In addition to architectural developments, attention mechanisms have been investigated to improve the representation of informative visual features. The Convolutional Block Attention Module (CBAM) [17] combines channel and spatial attention to refine feature representations. Such mechanisms provide a way to emphasize relevant information within feature maps and have motivated the development of attention-based CNN models. Furthermore, Plested et al. [25] reviewed deep transfer learning approaches for image classification, providing background on the use of pretrained models for visual recognition. Together, these studies establish the architectural and methodological background for evaluating different CNN models and examining the role of spatial attention in the present study.

Apart from model architecture, training strategies can influence classification performance, particularly when the available training data contain unequal class distributions. Data augmentation is commonly used to increase the diversity of training samples. Cheung and Yeung [26] reviewed automated data-augmentation methods, discussing approaches for selecting transformations to improve model training. However, increasing sample diversity is not the only way to address unequal class representation. Chen et al. [13] and Zhang et al. [12] surveyed methods for class-imbalanced and long-tailed recognition, respectively. Cui et al. [6] proposed class-balanced loss based on the effective number of samples, providing an approach for incorporating class distribution into loss calculation. Fu et al. [11] further reviewed methodological approaches to long-tailed visual recognition.

These studies provide the background for investigating different strategies to address class imbalance, including data augmentation, loss-based approaches, and sampling methods. In the present study, these approaches are examined through targeted data augmentation, class-weighted cross-entropy loss, their combination, and weighted random sampling. The relevant CNN architectures and training approaches are summarized in Table II.

TABLE II. CNN ARCHITECTURES AND TRAINING APPROACHES

Research area	Method/approach	Main contribution	Reference
CNN-based image recognition	Convolutional neural networks	Established an early foundation for CNN-based visual recognition	[8]
CNN architecture	Residual connections	Facilitates the training of deeper networks	[7]
CNN architecture	Compound scaling	Balances network depth, width, and resolution	[9]
Efficient CNN architecture	Lightweight network design	Focuses on computationally efficient visual recognition	[19]
Attention mechanism	Channel and spatial attention	Refines feature representations using attention	[17]
Transfer learning	Review of deep transfer learning	Reviews pretrained-model approaches for image classification	[25]
Data augmentation	Review of automated augmentation methods	Examines approaches for selecting image transformations	[26]
Imbalanced learning	Reviews of class-imbalanced and long-tailed recognition	Discuss approaches to unequal class distributions	[12], [13]
Class-imbalanced learning	Class-balanced loss based on effective sample numbers	Incorporates class distribution into loss calculation	[6]
Long-tailed visual recognition	Review of methodological approaches	Summarizes methods for long-tailed recognition	[11]

Previous studies have investigated physical-contact recognition, hand-to-face touch detection, human-interaction classification, and broader human activity recognition using different visual representations and input modalities. Other research has introduced CNN architectures, attention mechanisms, transfer-learning approaches, and strategies for addressing class imbalance. However, these studies address different recognition objectives and experimental settings. In particular, a comparative investigation of different CNN architectures and training strategies for identifying body-region touch events from individual RGB frames remains relevant.

To address this research problem, the present study evaluates six CNN architectures, including a ResNet18-based spatial-

attention model, under five training strategies. The experiments use four touch-event classes and five seeded participant-video-group partitions, with model selection based on validation macro F1. By examining architecture selection and training strategies under a consistent evaluation protocol, this study provides a comparative assessment of frame-level body-region touch-event recognition within the investigated dataset.

III. METHODOLOGY

This section describes the methodology adopted for frame-level touch-event recognition. It first defines the classification task and presents the dataset, followed by the group-aware data partitioning procedure used to construct five experimental splits. The six CNN architectures and five training strategies are then introduced, including the image preprocessing, data augmentation, and class-imbalance handling procedures. Finally, the optimization and model selection settings, together with the evaluation metrics, are presented to explain how the performance of the investigated configurations is assessed.

A. Task Definition and Dataset

This study addresses touch-event recognition as a four-class image classification task. Given a single RGB frame x, the classifier predicts one of four classes: No Touch, Touch Head, Touch Hips, or Touch Shoulders. The three touch classes indicate the visible body region associated with a labelled touch event, whereas No Touch represents the absence of a labelled touch event in the frame. Each frame is classified independently without relying on temporal information from neighboring frames. The initial dataset contained 826 PNG images. Exact content duplicates were identified using MD5 hashes, and 11 duplicate files were removed by retaining the first filename in sorted order. This resulted in 815 unique frames for the experiments. The retained dataset consisted of 667 No Touch, 29 Touch Head, 67 Touch Hips, and 52 Touch Shoulders images. Fig. 1 illustrates the distribution of the retained images across the four classification categories, highlighting the variation in sample representation among the classes.

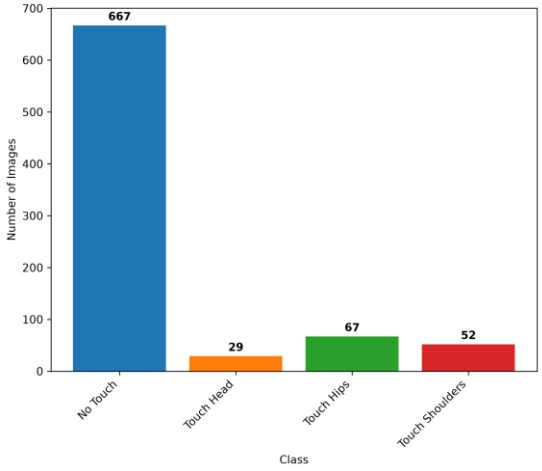

Fig. 1. Distribution of the 815 retained images across the four touch-event classes after exact duplicate removal.

As illustrated in Fig. 1, the class distribution is highly imbalanced. No Touch is the dominant class, accounting for 81.84% of the retained images. In contrast, Touch Head, Touch Hips, and Touch Shoulders represent 3.56%, 8.22%, and 6.38% of the dataset, respectively. The difference is particularly evident between No Touch and Touch Head, the least represented class. This imbalance means that the classifier has considerably more training examples for No Touch than for the three touch-related classes, which may make the recognition of less frequent categories more challenging. It also highlights the importance of evaluating performance across all classes rather than relying on overall accuracy alone. Accordingly, different training strategies for addressing class imbalance are investigated in this study.

The duplicate assessment also revealed metadata inconsistencies among the removed files. Eight of the 11 duplicate files were associated with different participant-video metadata groups, while the remaining three were within-group duplicates. For duplicates associated with different groups, the retained filename determined the group assignment. As a result, identical visual content could originally have been associated with different participant-video groups. This limitation was retained as part of the dataset curation procedure. In addition, seven filenames did not contain participant-video identifiers. These frames were assigned to a common group based on timestamp proximity and hash checks to reduce the possibility of frames from the same recording being distributed across different data partitions.

B. Group-Aware Data Partitioning

To evaluate model performance under different data partitioning conditions, five train-validation-test partitions were generated using random seeds 42, 123, 456, 789, and 1011. Frames were grouped according to participant-video identifiers parsed from their filenames to maintain group-level separation during partitioning. The split procedure targeted an approximate distribution of 70% for training, 15% for validation, and 15% for testing, while enforcing minimum class representation in each partition. The main configuration parameters used for generating these partitions are summarized in Table III.

TABLE III. SUMMARY OF THE GROUP-AWARE DATA PARTITIONING CONFIGURATION

Parameter	Description
Number of partitions	5
Random seeds	42, 123, 456, 789, 1011
Training proportion	Approximately 70%
Validation proportion	Approximately 15%
Testing proportion	Approximately 15%
Grouping unit	Participant-video pair
Group exclusivity	Enforced across train, validation, and test sets
Dataset size	815 unique frames

The grouping strategy uses the participant-video pair as the partitioning unit rather than the participant alone. Consequently, an individual appearing in multiple videos may occur in different partitions through separate videos. Therefore, the evaluation measures generalization across previously unseen participant-video groups rather than generalization to individuals entirely absent from the training data. Each random

seed generates a different partition from the same set of 815 unique images. Since the resulting test sets may overlap across seeds, the five-seed evaluation characterizes performance variation across different partitions rather than performance on five independent external test sets. Throughout the partitioning process, group exclusivity was maintained across the training, validation, and test sets, while record coverage, minimum class representation, and dataset fingerprint consistency were also checked.

C. CNN Architectures

Six convolutional neural network (CNN) architectures were evaluated to investigate their suitability for frame-level touch-event classification. The selected architectures represent different network designs and computational characteristics, enabling a comparative assessment of feature extraction and classification performance on the target dataset. ResNet50 and ResNet18 employ residual connections to facilitate the training of deep networks [7]. EfficientNet-B0 adopts a compound scaling strategy that jointly considers network depth, width, and input resolution [9], whereas MobileNetV3-Small is designed to provide computationally efficient inference [19]. CNNs have demonstrated long-established capabilities in image recognition [8], while transfer learning offers a practical approach to adapting pretrained feature representations to relatively small target datasets [25]. In addition to these pretrained architectures, a Simple CNN trained from scratch was included as a compact reference model.

Fig. 2 provides an overview of the six CNN architectures evaluated in this study and their respective roles in the four-class touch-event recognition task. The framework illustrates the use of different feature extraction architectures, including conventional residual networks, an efficient network design, a lightweight architecture, a CNN trained from scratch, and a spatial-attention-enhanced ResNet18 variant. All six configurations follow the same classification objective, allowing their performance to be compared under the experimental settings described in the following subsections.

A spatial-attention variant of ResNet18 was also evaluated to investigate whether emphasizing spatially informative regions of the feature maps could improve the recognition of visible touch-related body regions. The model incorporates a spatial-attention module based on the spatial component of the Convolutional Block Attention Module (CBAM) [17]. Specifically, channel-wise average pooling and maximum pooling are applied to the ResNet18 feature map, and their outputs are concatenated along the channel dimension. The resulting feature representation is passed through a 7×7 convolutional layer and a sigmoid activation function to generate a single-channel spatial attention mask. This mask is multiplied elementwise with the original feature map to emphasize spatially relevant features before global average pooling and four-class classification. Unlike the complete CBAM architecture, which sequentially applies channel and spatial attention, the implemented variant uses spatial attention only. To support a controlled comparison, the ResNet18 baseline and its spatial-attention variant use matched classifier initialization in the paired experiments.

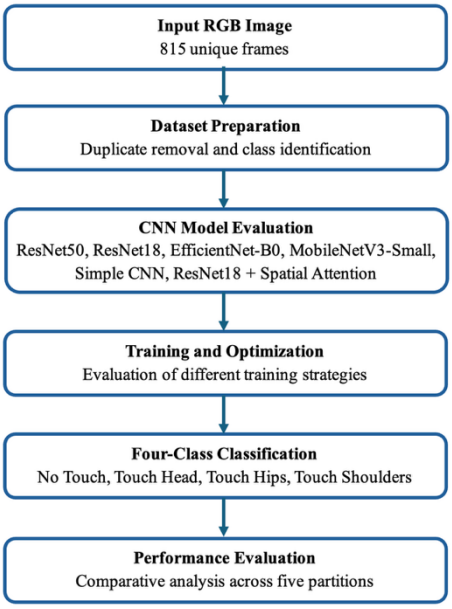

Fig. 2. Overview of the experimental workflow for CNN-based frame-level touch-event recognition.

The pretrained ResNet50, ResNet18, EfficientNet-B0, MobileNetV3-Small, and ResNet18 backbone of the spatial-attention model were initialized using ImageNet-1K V1 weights. For ResNet50 and ResNet18, the final residual stage (layer4) and the replacement four-class classifier were made trainable, while the earlier backbone layers remained frozen. EfficientNet-B0 fine-tunes its final two feature blocks (features.7 and features.8), together with the replacement classifier. Similarly, MobileNetV3-Small fine-tunes its final three feature blocks (features.10, features.11, and features.12) and the replacement classifier. For the spatial-attention model, ResNet18 layer4, the spatial-attention module, and the replacement classifier were trainable, while the earlier backbone layers remained frozen. Batch-normalization statistics in frozen backbone layers were kept fixed by maintaining those layers in evaluation mode during training.

The Simple CNN was trained from scratch without pretrained weights. It consists of four convolutional blocks with 32, 64, 128, and 256 channels, respectively. Each block incorporates batch normalization and a rectified linear unit (ReLU) activation function. Max pooling is applied after each of the first three blocks to progressively reduce the spatial dimensions of the feature maps. The resulting feature representation is then processed using adaptive average pooling, followed by a dropout layer and a fully connected classifier that produces scores for the four target classes.

Given the differences in network structure, pretrained initialization, and the proportion of trainable parameters, the comparison considers complete model-and-training configurations rather than attributing performance differences exclusively to architectural design. This comparative evaluation provides the basis for investigating how different CNN designs and spatial attention perform on the proposed frame-level touch-event classification task.

D. Image Preprocessing and Training Strategies

To ensure consistency across the experiments, all input images were converted to RGB format and resized to (224 × 224) pixels. The images were then converted into tensors and normalized using the ImageNet mean values of [0.485, 0.456, 0.406] and standard deviation values of [0.229, 0.224, 0.225]. For validation and test sets, the same deterministic preprocessing operations were applied without additional data augmentation.

Five training strategies E1–E5 were investigated to examine the effects of data augmentation and class-imbalance handling on model performance. These strategies combine or isolate targeted data augmentation, class-weighted cross-entropy loss, and weighted random sampling. Their configurations are summarized in Table IV.

TABLE IV. SUMMARY OF THE EVALUATED TRAINING STRATEGIES

Strategy	Data augmentation	Class-weighted loss	Weighted random sampling
E1	No	No	No
E2	Yes	No	No
E3	No	Yes	No
E4	Yes	Yes	No
E5	No	No	Yes

E1 - Baseline strategy: The baseline configuration used the original training images without data augmentation, class-weighted loss, or weighted random sampling. Standard cross-entropy loss was used for model training.

E2 - Targeted data augmentation: This strategy applied targeted data augmentation without class-weighted loss or weighted random sampling. To increase the representation of minority classes during training, the training dataset was accessed using class-specific repetition multipliers. No Touch images were accessed once, Touch Head images five times, and Touch Hips and Touch Shoulders images three times each. These multipliers increased the frequency with which images were accessed during training rather than creating a fixed collection of augmented files. A new random transformation was applied whenever an image was accessed.

For the three touch classes, the augmentation operations included random horizontal flipping with a probability of 0.5, rotation within ±5°, brightness and contrast adjustments of 0.15, saturation adjustment of 0.1, hue adjustment of 0.03, and translation of up to 5% in each direction. A lighter augmentation policy was applied to the No Touch class, consisting of random horizontal flipping with a probability of 0.3 and brightness and contrast adjustments of 0.1.

E3 - Class-weighted loss: This strategy used the original training images without data augmentation or weighted random sampling. Class-weighted cross-entropy loss was employed to reduce the influence of the imbalanced class distribution. The weight assigned to each class was calculated using the inverse-frequency formulation:

$$w_k = \frac{K(n_k+\varepsilon)^{-1}}{\sum_{j=1}^{K}(n_j+\varepsilon)^{-1}}, \quad k = 1, \dots, K, \quad \varepsilon = 10^{-6} \qquad (1)$$

where w_k denotes the weight assigned to class k, n_k represents the number of training samples belonging to that class, and K is the total number of classes. This normalization ensures that the sum of the class weights equals the number of classes.

For a minibatch containing B samples, the weighted cross-entropy loss was calculated as:

$$\mathcal{L} = -\frac{\sum_{i=1}^{B} w_{y_i} \log p_{i,y_i}}{\sum_{i=1}^{B} w_{y_i}} \qquad (2)$$

where y_i denotes the ground-truth class of sample i, $p_{i,yi}$ is the predicted probability assigned to its ground-truth class, and w_{y_i} is the corresponding class weight. The denominator normalizes the weighted loss by the sum of the class weights within the minibatch, following the weighted-mean reduction used in PyTorch.

E4 - Combined augmentation and class-weighted loss: This strategy combined the targeted augmentation policy described in E2 with class-weighted cross-entropy loss. The same class-specific repetition multipliers were applied to increase the access frequency of minority-class images. Class weights were calculated using the class frequencies in the resulting repeated training-label sequence. Thus, both data augmentation and class-weighted loss were incorporated into the training process, without weighted random sampling.

E5 - Weighted random sampling: This strategy used the original training images without data augmentation or class-weighted loss. Instead, weighted random sampling was employed to increase the probability of selecting samples from underrepresented classes. Each sample was assigned a sampling weight inversely proportional to the frequency of its corresponding class. Sampling was performed with replacement, and the number of samples drawn in each epoch was kept equal to the size of the original training partition.

The five strategies were designed to investigate different approaches to handling class imbalance during model training. E1 serves as the baseline configuration using standard cross-entropy loss, while E2 and E3 separately examine targeted data augmentation and class-weighted loss, respectively. E4 evaluates the combination of data augmentation and class-weighted loss, whereas E5 investigates weighted random sampling as an alternative balancing mechanism. This experimental design enables a comparative assessment of the effects of different training strategies across the evaluated CNN architectures and data partitions.

E. Optimization and Model Selection

All models are optimized using Adam [10] with an initial learning rate of 10^{-4} and a batch size of 32. Training runs for at least five epochs and 50 epochs at most. The learning-rate scheduler reduces the learning rate by a factor of 0.5 when validation Macro F1 fails to improve for more than three consecutive epochs. Early stopping uses a patience of seven epochs and a minimum improvement threshold of 0.001. Gradient norms are clipped at 1.0, and the model checkpoint with the highest validation Macro F1 is restored. The ResNet18 baseline is compared directly with a variant that uses the same ResNet18 backbone followed by a spatial-attention module to

examine whether adding spatial attention changes classification performance. Corresponding runs use the same split seeds and matched initialization seeds within each experiment for aligned experimental conditions. The results are summarized as paired descriptive comparisons across the five seeds.

F. Evaluation Metrics

Performance is evaluated using accuracy, balanced accuracy, and macro F1. Accuracy measures the proportion of correctly classified frames, where N is the total number of evaluated examples.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{N} \quad (3)$$

balanced accuracy is the unweighted mean of per-class recall, and macro F1 is the unweighted mean of the four class-specific F1-scores [9]. For K classes, the metrics are defined as follows:

$$\text{Balanced Accuracy} = \frac{1}{K}\sum_{k=1}^{K} \text{Recall}_k \quad (4)$$

$$F1_k = \frac{2\,\text{Precision}_k\,\text{Recall}_k}{\text{Precision}_k + \text{Recall}_k}, \quad \text{Macro F1} = \frac{1}{K}\sum_{k=1}^{K} F\,1_k \quad (5)$$

The mean and sample standard deviation were calculated across five seeded runs for each architecture strategy configuration. Figures and tables report means with descriptive 95% t-intervals to show the variability observed across these runs.

IV. Results

This section presents the experimental results obtained from evaluating different CNN architectures and training strategies for four-class touch-event recognition. The evaluation focuses on overall classification performance across the tested model configurations, with particular emphasis on macro F1-score and balanced accuracy due to the imbalanced class distribution. The results are analyzed in two parts. First, the overall performance of the evaluated architectures and training strategies is compared to identify the configurations that achieved the strongest results. Second, a paired comparison is conducted to examine the effect of incorporating spatial attention into ResNet18 under different training strategies. All reported performance values are calculated across five seeded runs, with descriptive 95% t-intervals used to indicate variability in the estimated mean performance.

A. Overall Performance

The overall performance of the evaluated CNN architectures was examined across the five training strategies to assess their classification effectiveness and identify the configurations achieving the highest test performance.

Among the evaluated configurations, ResNet18 trained using class-weighted loss under E3 achieved the highest mean test macro F1-score of 0.344, with a descriptive 95% t-interval half-width of 0.103. This was closely followed by ResNet18 with spatial attention under E3, which achieved a mean macro F1-score of 0.342 (±0.127). Other competitive configurations included ResNet50 with weighted sampling under E5, with a mean macro F1-score of 0.340 (±0.085), and EfficientNet-B0 with class-weighted loss under E3, which achieved 0.339 (±0.080). Although these configurations demonstrated relatively similar mean performance, the interval widths indicate variation across the five seeded runs.

The results further demonstrate that the effectiveness of a training strategy depends on the underlying architecture. Class-weighted loss under E3 achieved the highest mean macro F1-score for ResNet18, EfficientNet-B0, MobileNetV3-Small, and ResNet18 with spatial attention. In contrast, weighted sampling under E5 produced the highest mean for ResNet50. This variation suggests that the interaction between model architecture and class-imbalance handling plays an important role in determining classification performance. The Simple CNN exhibited comparatively poor performance under E3 and E5, indicating that class-imbalance handling alone does not guarantee improved classification performance. The effectiveness of rebalancing techniques may depend on the representational capacity of the architecture and its ability to learn discriminative features from underrepresented classes.

A further observation concerns the difference between accuracy and macro F1-score across the evaluated configurations. Baseline ResNet18 achieved a mean accuracy of 0.809 but a considerably lower mean macro F1-score of 0.317. In comparison, ResNet18 with E3 achieved a lower mean accuracy of 0.765 but a higher mean macro F1-score of 0.344, together with a mean balanced accuracy of 0.364. This difference highlights the limitations of accuracy as a standalone evaluation metric in an imbalanced classification setting. In particular, the dominance of the No Touch class can contribute to relatively high accuracy without corresponding improvements in performance across all four classes. Therefore, macro F1-score and balanced accuracy provide complementary information for assessing the models' ability to recognize both majority and minority classes.

Fig. 3 illustrates the mean test macro F1-scores across the 30 evaluated configurations enabling a comparative assessment of the CNN architectures and training strategies. The results show differences in classification performance across configurations, highlighting that the choice of architecture and training strategy influences the achieved macro F1-score. The error bars represent descriptive 95% t-intervals around the estimated means and provide an indication of the variability observed across the five seeded runs. Wider intervals indicate greater variation in performance across runs, whereas narrower intervals suggest more consistent results. Notably, configurations with similar mean macro F1-scores may exhibit different levels of variability, indicating that the highest mean score does not necessarily correspond to the most consistent performance. Therefore, both the estimated mean performance and its variability should be considered when comparing the evaluated configurations. Table V summarizes the CNN architecture and training-strategy configurations with the strongest results in terms of mean accuracy, balanced accuracy, and macro F1-score, including the descriptive 95% t-intervals for all metrics.

TABLE V. CNN ARCHITECTURES AND TRAINING APPROACHES

Training strategy	Architecture	Accuracy (mean ± 95% t-interval)	Balanced accuracy (mean ± 95% t-interval)	Macro F1 (mean ± 95% t-interval)
E3 Weighted loss	ResNet18	0.765 ± 0.093	0.364 ± 0.126	0.344 ± 0.103
E3 Weighted loss	ResNet18 + spatial attention	0.693 ± 0.165	0.344 ± 0.130	0.342 ± 0.127
E5 Weighted sampling	ResNet50	0.709 ± 0.141	0.385 ± 0.160	0.340 ± 0.085
E3 Weighted loss	EfficientNet-B0	0.675 ± 0.114	0.376 ± 0.145	0.339 ± 0.080
E5 Weighted sampling	ResNet18 + spatial attention	0.703 ± 0.164	0.373 ± 0.172	0.335 ± 0.135
E1 Baseline	ResNet18	0.809 ± 0.053	0.313 ± 0.066	0.317 ± 0.094

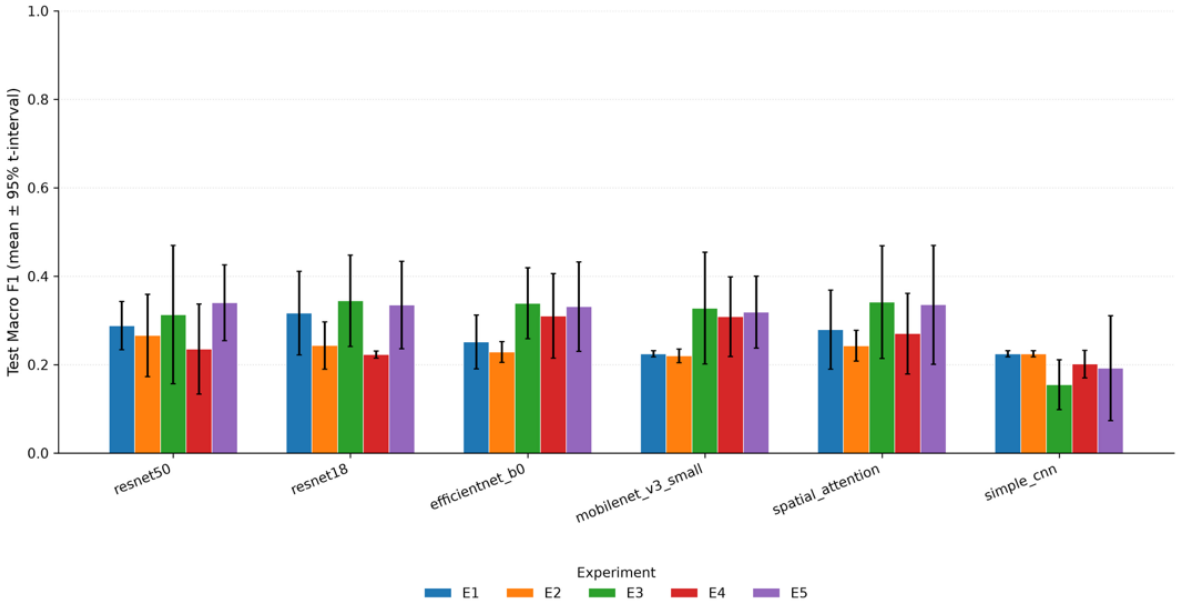

Fig. 3. Mean test macro F1 across the 30 configurations. Means are across five seeded runs, and error bars show descriptive 95% t-intervals.

B. Spatial-Attention Comparison

To investigate the contribution of spatial attention, ResNet18 was compared with its spatial-attention counterpart under each of the five training strategies. The comparison used matched split seeds and corresponding initialization seeds to ensure that both architectures were evaluated under aligned experimental conditions.

The results reveal that the effect of spatial attention varied across training strategies. Under E1, the baseline ResNet18 achieved a mean test macro F1-score of 0.317, compared with 0.279 for the spatial-attention model. Under E2, both models achieved an identical mean of 0.243. Similarly, under E5, both configurations obtained a mean macro F1-score of 0.335. For E3, the baseline and spatial-attention models achieved mean scores of 0.344 and 0.342, respectively. In contrast, under E4, the spatial-attention model achieved a higher mean of 0.270 compared with 0.222 for the baseline model.

These findings indicate that incorporating spatial attention did not consistently improve classification performance across all training strategies. Although the spatial-attention model achieved a higher mean under E4, its performance was lower under E1 and marginally lower under E3. The identical mean scores observed under E2 and E5 further suggest that the effect of spatial attention was limited under these configurations. Overall, the findings indicate that the contribution of spatial attention is dependent on the training strategy rather than providing a consistent improvement over the baseline architecture.

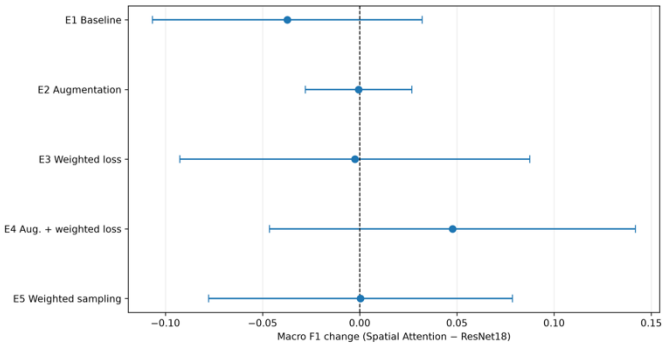

Fig. 4. Comparison of paired test macro-F1 differences between ResNet18 with spatial attention and the baseline ResNet18 across the five training strategies.

Fig. 4 illustrates the paired differences in test macro F1-score between ResNet18 with spatial attention and the baseline ResNet18 across the five training strategies. Positive differences indicate higher macro F1-scores for the spatial-attention model, whereas negative differences indicate better performance by the

baseline model. The dashed vertical line represents zero difference, indicating equivalent performance between the two configurations.

The results indicate that the effect of spatial attention varies across the five training strategies. The largest positive mean difference was observed under E4 +0.048, suggesting that spatial attention provided a potential performance benefit under this configuration. In contrast, E1 produced the largest negative difference –0.038, indicating comparatively lower performance for the spatial-attention model. The differences under E2 and E5 were zero, while E3 showed only a marginal negative difference –0.002. However, the confidence intervals should also be considered when interpreting these differences, as intervals crossing zero indicate that the observed effects are not clearly distinguishable from zero at the 95% confidence level.

V. Discussion

The experimental results demonstrate that frame-level touch-event recognition can be investigated using a consistent image classification framework, although the classification performance varies across the evaluated CNN architectures and training strategies. The comparison of five training strategies reveals that class-weighted cross-entropy loss achieved the highest mean test macro-F1 scores for ResNet18, EfficientNet-B0, MobileNetV3-Small, and the spatial-attention model.

In contrast, weighted random sampling produced the highest mean macro-F1 score for ResNet50, while targeted data augmentation alone did not provide a consistent performance improvement across the evaluated architectures. These findings indicate that no single training strategy was uniformly effective for all models. Instead, the effectiveness of class-imbalance handling depended on the architecture and training configuration. Therefore, evaluating class-sensitive metrics, particularly macro-F1 and balanced accuracy, alongside conventional accuracy is important when assessing models trained on highly imbalanced datasets.

Furthermore, the comparison between the ResNet18 baseline and its spatial-attention variant provides an opportunity to examine the effect of incorporating spatial feature weighting into the classification process. Although the spatial-attention module introduces a mechanism for assigning different weights to spatial locations in the feature maps, the experimental results did not demonstrate a consistent improvement over the ResNet18 baseline across the five training strategies. This suggests that incorporating spatial attention alone does not necessarily lead to better classification performance under the evaluated experimental conditions. Its observed effect also varied depending on the training strategy, indicating that architectural modifications and class-imbalance handling should be considered together when evaluating model performance. Similarly, the Simple CNN exhibited different responses to class-weighted loss and weighted random sampling, further illustrating that the effectiveness of rebalancing techniques can depend on the network architecture and the available training data.

The differences observed among the evaluated architectures should also be interpreted in the context of their respective model configurations. The pretrained networks, including ResNet50, ResNet18, EfficientNet-B0, and MobileNetV3-Small, use ImageNet-based feature representations and different fine-tuning configurations, whereas the Simple CNN learns its feature representations directly from the target dataset. In addition, the spatial-attention model introduces an additional feature-weighting component to the ResNet18 backbone. Consequently, the comparison reflects the performance of complete model-and-training configurations rather than the isolated effect of architectural design. The results therefore provide an empirical baseline for evaluating CNN-based frame-level touch-event recognition on the dataset used in this study, rather than establishing a universally preferred architecture or training strategy.

Despite these findings, several dataset and evaluation limitations should be considered when interpreting the results. The retained dataset contains 815 images and is substantially imbalanced, with 667 No Touch images compared with only 29 Touch Head images. Consequently, the minority classes contribute relatively few samples to individual test partitions, meaning that a small number of different predictions can noticeably affect class-specific performance metrics. The reported intervals reflect variation across the implemented experimental runs rather than independent observations from a wider population, since the five seeded partitions reuse the same limited image pool and their test sets may overlap. Furthermore, the partitioning procedure groups images by participant-video pairs rather than by participant identity. Therefore, a participant appearing in multiple videos may be represented in different partitions, and the current evaluation does not establish generalization to individuals entirely absent from the training data.

The duplicate-removal procedure also introduces a consideration regarding group assignment. The duplicate audit identified eight duplicate files associated with differing participant-video metadata groups. As the first filename in sorted order was retained, its associated metadata determined the group assignment for these cases. This may influence the separation of related visual content across the data partitions. Moreover, label preservation under the selected augmentation operations was not evaluated separately, and the suitability of these transformations for every touch-event image therefore remains an aspect for further investigation.

Finally, the source dataset was originally developed for dance-video analysis rather than as a dedicated touch-event recognition benchmark. This may limit the representativeness of the selected frames for touch events in other activities and real-world settings. These considerations should be taken into account when interpreting the reported performance and motivate further evaluation using larger, more diverse, and specifically annotated touch-event datasets.

VI. Conclusion

This study investigated frame-level touch-event recognition using individual RGB images, addressing a four-class classification task involving No Touch, Touch Head, Touch Hips, and Touch Shoulders. Each frame was classified independently without relying on temporal information from neighbouring frames. Given the limited number of available samples and the substantial class imbalance in the dataset,

evaluating the suitability of different convolutional neural network (CNN) architectures and training strategies is important for understanding their classification performance. Therefore, this study conducted a comparative evaluation of six CNN architectures, namely ResNet50, ResNet18, EfficientNet-B0, MobileNetV3-Small, a ResNet18-based spatial-attention model, and a Simple CNN. Five training strategies were also investigated to examine the effects of targeted data augmentation, class-weighted cross-entropy loss, and weighted random sampling on model performance. The experiments were conducted across five seeded participant-video-group partitions to provide a consistent basis for comparison.

The experimental results showed that ResNet18 with class-weighted loss achieved the highest mean test macro-F1 score of 0.344, followed by the spatial-attention model with class-weighted loss (0.342), ResNet50 with weighted random sampling (0.340), and EfficientNet-B0 with class-weighted loss (0.339). The findings demonstrate that the effectiveness of training strategies varies across architectures and that incorporating spatial attention does not consistently improve classification performance over the ResNet18 baseline. These results highlight the importance of considering macro-F1 and balanced accuracy alongside conventional accuracy when evaluating models on highly imbalanced datasets. Although the evaluated models provide a comparative baseline for frame-level touch-event recognition, further improvements require larger and more diverse datasets, particularly for minority touch classes, together with more comprehensive annotation documentation to support label reliability. Future research could also investigate participant-exclusive evaluation to assess generalization to previously unseen individuals, as well as the integration of pose information, improved spatial representations, and temporal sequences to determine whether additional visual and temporal information can provide more reliable evidence of touch events. These extensions would help establish the generalizability and practical applicability of frame-level touch-event recognition systems.

FUNDING STATEMENT

This work was supported by Universiti Malaysia Sarawak through the following grant scheme: PETRONAS P. Ramlee Research Chair (PRC) (UNI/F08/PRC/86580/2025).

REFERENCES

[1] I. Jiménez-Velasco, J. Zafra-Palma, R. Muñoz-Salinas, and M. J. Marín-Jiménez, “Proxemics-net++: classification of human interactions in still images,” Pattern Anal. Appl., vol. 27, no. 2, p. 49, Jun. 2024, doi: 10.1007/s10044-024-01270-3.

[2] S. Narasimhaswamy, T. Nguyen, and M. Hoai, “Detecting Hands and Recognizing Physical Contact in the Wild,” Oct. 19, 2020, arXiv: arXiv:2010.09676. doi: 10.48550/arXiv.2010.09676.

[3] F. Shafizadegan, A. R. Naghsh-Nilchi, and E. Shabaninia, “Multimodal vision-based human action recognition using deep learning: a review,” Artif. Intell. Rev., vol. 57, no. 7, p. 178, Jun. 2024, doi: 10.1007/s10462-024-10730-5.

[4] M. H. Azhar et al., “Intelligent Human Interaction Recognition with Multi-Modal Feature Extraction and Bidirectional LSTM,” Comput. Mater. Contin., vol. 87, no. 1, pp. 1–10, 2026, doi: 10.32604/cmc.2025.071988.

[5] Q. Chen, H. Li, R. Abu-Zhaya, A. Seidl, F. Zhu, and E. J. Delp, “Touch Event Recognition For Human Interaction,” Electron. Imaging, vol. 28, no. 11, pp. 1–6, Feb. 2016, doi: 10.2352/ISSN.2470-1173.2016.11.IMAWM-465.

[6] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, “Class-Balanced Loss Based on Effective Number of Samples,” 2019.

[7] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[8] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” Proc. IEEE, vol. 86, no. 11, pp. 2278–2324, Nov. 1998, doi: 10.1109/5.726791.

[9] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” 2019.

[10] Z. Liu, H. Mao, C.-Y. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, “A ConvNet for the 2020s,” in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA: IEEE, Jun. 2022, pp. 11966–11976. doi: 10.1109/CVPR52688.2022.01167.

[11] Y. Fu et al., “Long-tailed visual recognition with deep models: A methodological survey and evaluation,” Neurocomputing, vol. 509, pp. 290–309, Oct. 2022, doi: 10.1016/j.neucom.2022.08.031.

[12] Y. Zhang, B. Kang, B. Hooi, S. Yan, and J. Feng, “Deep Long-Tailed Learning: A Survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 9, pp. 10795–10816, Sep. 2023, doi: 10.1109/TPAMI.2023.3268118.

[13] W. Chen, K. Yang, Z. Yu, Y. Shi, and C. L. P. Chen, “A survey on imbalanced learning: latest research, applications and future directions,” Artif. Intell. Rev., vol. 57, no. 6, p. 137, May 2024, doi: 10.1007/s10462-024-10759-6.

[14] Y. Zhang, X.-S. Wei, B. Zhou, and J. Wu, “Bag of Tricks for Long-Tailed Visual Recognition with Deep Convolutional Neural Networks,” Proc. AAAI Conf. Artif. Intell., vol. 35, no. 4, pp. 3447–3455, May 2021, doi: 10.1609/aaai.v35i4.16458.

[15] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” Inf. Process. Manag., vol. 45, no. 4, pp. 427–437, Jul. 2009, doi: 10.1016/j.ipm.2009.03.002.

[16] Z. Yang, R. O. Sinnott, J. Bailey, and Q. Ke, “A survey of automated data augmentation algorithms for deep learning-based image classification tasks,” Knowl. Inf. Syst., vol. 65, no. 7, pp. 2805–2861, Jul. 2023, doi: 10.1007/s10115-023-01853-2.

[17] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, “CBAM: Convolutional Block Attention Module,” in Computer Vision – ECCV 2018, vol. 11211, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., in Lecture Notes in Computer Science, vol. 11211. , Cham: Springer International Publishing, 2018, pp. 3–19. doi: 10.1007/978-3-030-01234-2_1.

[18] I. Hipiny, H. Ujir, A. A. Alias, M. Shanat, and M. K. Ishak, “Who danced better? ranked tiktok dance video dataset and pairwise action quality assessment method,” Int. J. Adv. Intell. Inform., vol. 9, no. 1, p. 96, Mar. 2023, doi: 10.26555/ijain.v9i1.919.

[19] A. Howard et al., “Searching for MobileNetV3,” Nov. 20, 2019, arXiv: arXiv:1905.02244. doi: 10.48550/arXiv.1905.02244.

[20] L. Muller, A. A. A. Osman, S. Tang, C.-H. P. Huang, and M. J. Black, “On Self-Contact and Human Pose,” in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA: IEEE, Jun. 2021, pp. 9985–9994. doi: 10.1109/CVPR46437.2021.00986.

[21] M. R. Ibrahim and T. Lyons, “FaceTouch: Detecting hand-to-face touch with supervised contrastive learning to assist in tracing infectious diseases,” PLOS ONE, vol. 19, no. 6, p. e0288670, Jun. 2024, doi: 10.1371/journal.pone.0288670.

[22] H. Kaur, V. Rani, and M. Kumar, “Human activity recognition: A comprehensive review,” Expert Syst., vol. 41, no. 11, p. e13680, Nov. 2024, doi: 10.1111/exsy.13680.

[23] N. S. Kumar et al., “HARNet in deep learning approach—a systematic survey,” Sci. Rep., vol. 14, no. 1, p. 8363, Apr. 2024, doi: 10.1038/s41598-024-58074-y.

[24] Y. Zhang and Y. Wang, “A comprehensive survey on RGB-D-based human action recognition: algorithms, datasets, and popular applications,” EURASIP J. Image Video Process., vol. 2025, no. 1, p. 15, Aug. 2025, doi: 10.1186/s13640-025-00677-0.

[25] J. Plested, M. Phiri, and T. Gedeon, "Deep transfer learning for image classification: a survey," Artif. Intell. Rev., vol. 59, no. 3, p. 100, Jan. 2026, doi: 10.1007/s10462-026-11491-z.

[26] T.-H. Cheung and D.-Y. Yeung, "A Survey of Automated Data Augmentation for Image Classification: Learning to Compose, Mix, and Generate," IEEE Trans. Neural Netw. Learn. Syst., vol. 35, no. 10, pp. 13185–13205, Oct. 2024, doi: 10.1109/TNNLS.2023.3282258.