A Dynamic Blurring Approach with EfficientNet and LSTM to Enhance Privacy in Video-Based Elderly Fall Detection

Ivan Ursul*1*, Junaid Hussain Muzamal*2*

1 Ivan Franko National University of Lviv, Lviv, Universitetska, 1, 79090, Ukraine
2 Fast – National University of Computer and Emerging Sciences, Lahore, Pakistan

Abstract
This research paper introduces a novel approach to address privacy concerns in video-based elderly fall detection systems without compromising such technologies' efficacy and real-time response. The methodology integrates EfficientNetB0 for robust feature extraction from video sequences and Long Short-Term Memory networks for accurate fall classification. Despite achieving exemplary performance metrics, including 100% scores in accuracy, Area Under the Curve, recall, and Precision, the pervasive issue of privacy infringement in video surveillance remains a significant challenge. To tackle this, we propose a dynamic blurring technique that selectively obscures identifiable features within video frames, such as faces and distinguishing clothing, thus maintaining individual anonymity. This method ensures that the privacy of the monitored individuals is preserved while retaining the essential details necessary for the fall detection algorithm to function effectively. This paper details this privacy-preserving technique and demonstrates its feasibility without detracting from the system's performance. Our findings indicate that integrating dynamic blurring into the fall detection pipeline offers a promising solution to the privacy concerns associated with video-based monitoring systems. It protects sensitive personal information while providing high care and safety. This research contributes to the broader discourse on ethical technology use in healthcare. Moreover, it emphasizes the importance of balancing advanced monitoring capabilities with the imperative of privacy preservation.

Keywords
Elderly Fall Detection, Privacy Preservation, Video Surveillance, EfficientNetB0, Long Short-Term Memory (LSTM), Dynamic Blurring, Real-Time Monitoring, Feature Extraction

1. Introduction

The growing demographic of the elderly population has precipitated an increased incidence of falls [1], a leading cause of morbidity and mortality among this group [2]. The recent technological solutions for fall detection have emerged as a critical component in mitigating these risks [3]. Among these, video-based fall detection systems have shown significant promise due to their non-invasiveness and capability for real-time monitoring [4]. However, video surveillance in healthcare, particularly in homes and care facilities, raises significant privacy concerns [5]. This research aims to find a balance between ensuring safety through surveillance and upholding the right to privacy.

There has been a significant relationship between the efficacy of video-based fall detection and the imperative to protect individual privacy [6]. While effective in identifying falls, traditional approaches often overlook the privacy implications of constant video monitoring [7]. Possible solutions include avoiding video data, implementing basic obfuscation techniques [8], compromising effectiveness, or insufficient privacy [9]. Previous research has proposed various methods, including wearable devices [10-13] and environmental sensors [14-16], to circumvent the associated privacy issues. However, these alternatives fall short in accuracy and real-time response capabilities compared to video-based systems.

In response to these challenges, this paper proposes an innovative solution that retains the advantages of video surveillance while addressing privacy concerns. Our approach employs dynamic blurring, selectively obscuring identifiable features within video frames. Thus, individuals are anonymized without compromising the system's ability to detect falls. This

✉ Ivanon2@gmail.com (I. Ursul); junaidhocane6728@gmail.com (J. H. Muzamal)
🆔 0009-0002-9879-8008 (I. Ursul); 0009-0007-1598-8161 (J. H. Muzamal);

method differs from existing solutions by offering a real-time, privacy-preserving mechanism that does not detract from the system's performance. Integrating EfficientNetB0 [17] for feature extraction and Long Short-Term Memory (LSTM) [18] networks for fall events ensures high precision in fall detection.

This research aims to develop a fall detection system that fulfills the need for efficient, real-time monitoring with the imperative of privacy preservation. Our objectives include designing and implementing a dynamic blurring technique within a video-based fall detection framework. Moreover, we also aim to evaluate this system's accuracy and privacy protection performance and demonstrate its applicability in real-world settings. This research can potentially contribute to the development of ethically responsible technological solutions in healthcare, particularly in the context of elderly care. This work seeks to pave the way for broader acceptance by addressing the privacy concerns associated with video-based monitoring. Moreover, deploying such systems enhances the safety and well-being of the elderly population.

2. Literature Review

The exploration of fall detection systems, particularly for the elderly, is an area of research that has seen substantial evolution over time. Dean et al. [19] 2006 implemented the first real-time fall detection system using a triaxial accelerometer. At that time, most traditional techniques centered around simplistic, mechanical solutions and gradually transitioned towards incorporating technology [20]. Among the earliest methods were basic alert systems, which relied on the user to trigger an alert manually in case of a fall [21]. While pioneering for their time, these systems were limited by their dependence on the users to activate the alarm post-fall, which could be compromised due to injury.

Advancements in technology brought in a new wave of methodologies, primarily categorized into sensor-based [14-16], wearable devices [10], [12], and video surveillance systems [22], [23-25], alongside other innovative approaches. Sensor-based systems often utilize accelerometers and gyroscopes to detect sudden movements or orientations indicative of a fall. Wearable devices, such as smartwatches [26], integrate these sensors and offer portability. However, sensor-based and wearable systems face challenges related to user compliance, discomfort, and the potential for false positives due to non-fall-related abrupt movements [27]. In contrast, video surveillance systems offer a less intrusive alternative, capturing a broader context of the individual's environment [28]. This method's appeal lies in its passive nature, requiring no active input or wearables from the monitored individuals. Despite these advantages, video-based systems have challenges [29]. High-quality video processing demands significant computational resources, and managing vast data volumes poses storage and efficiency concerns. Moreover, the critical issue of privacy infringement emerges, given the intrusive nature of continuous video monitoring [9].

Traditional algorithms such as Support Vector Machines (SVMs) [30] and Decision Trees [31] were widely employed in the early stages of machine learning applications for fall detection. These methods primarily relied on handcrafted features extracted from sensor data or basic video analytics, including motion vectors and silhouette shapes. Flow-based methods, particularly optical flow [32], were also prominent, enabling the detection of movement patterns by analyzing the apparent motion of objects, surfaces, and edges. While effective to a certain extent, these approaches faced limitations in handling the high variability and complexity of human falls. They often struggled to distinguish falls from other activities involving rapid movements, leading to high false alarm rates [33], [34]. Additionally, their dependency on manually crafted features restricted their adaptability, as these features might not generalize well across different scenarios.

The recent emergence of deep learning architectures like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) has changed many dynamics [35]. Advanced models such as ResNet [36], LSTM [37], and YOLO especially marked a leap forward in fall detection. CNNs, with their ability to perform automatic feature extraction, have proven particularly adept at analyzing spatial characteristics in video frames [38]. While RNNs and LSTMs excel in capturing temporal dependencies, it is crucial to understand the sequence of movements leading to a fall. YOLO [39], an object detection model, brought further advancements by enabling real-time processing. Despite their successes, the search for enhanced performance

led to exploring hybrid methods that combine multiple deep learning models. For instance, integrating CNNs with LSTMs allows for the effective processing of video data both spatially and temporally, offering a better understanding of fall events [40]. These hybrid approaches [41], alongside innovative methods within deep learning frameworks, promise to address the dynamics of fall detection [42].

Recent advancements aim to address these privacy concerns while maintaining system efficacy. Techniques such as dynamic blurring and real-time anonymization have been explored to obscure identifiable features in video feeds. This can help safeguard individual privacy without significantly compromising detection capabilities. Despite these efforts, there is a gap in the literature concerning developing a system that seamlessly integrates high detection accuracy with robust privacy protection. Our contribution to this field addresses this gap by proposing a novel fall detection system that employs EfficientNetB0 for advanced feature extraction and LSTM networks for accurate temporal classification, complemented by a dynamic blurring mechanism to ensure privacy. This integrated approach promises high performance, as evidenced by optimal accuracy, recall, and precision scores. Moreover, it introduces a viable solution to the privacy concerns that have long shadowed video-based monitoring systems. By achieving this delicate balance, our research paves the way for the broader acceptance of video-based fall detection systems, ensuring the safety of the elderly population.

3. Methodology

This research presents a methodological framework to address the challenge of detecting falls through video surveillance while safeguarding their privacy. The foundation of the proposed approach is mathematical models and techniques to ensure precision, efficiency, and reliability. The proposed method integrates state-of-the-art EfficientNetB0 for spatial feature extraction and LSTM networks for temporal sequence analysis. Additionally, we introduce a dynamic blurring mechanism formulated to preserve privacy by selectively obscuring identifiable features within video frames. Figure 1. provides the overall architecture of the proposed approach.
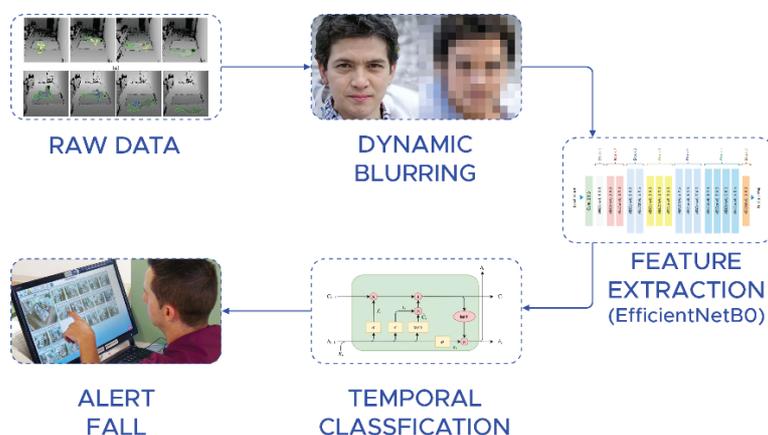

Figure 1. Overall Architecture of the Proposed Methodology

3.1. Dataset and Processing

The dataset employed in this study was sourced from the UR Fall Detection Dataset [43], encompassing 70 sequences, of which 30 are fall events, and 40 represent activities of daily living (ADL). The fall events were captured using two Microsoft Kinect cameras, accompanied by accelerometric data, whereas the ADL events were documented using a single camera (camera 0) alongside accelerometer data. The accelerometric data was acquired through PS Move (60Hz) and x-IMU (256Hz) devices. The dataset is structured such that each sequence comprises depth and RGB images from both camera perspectives (parallel to the floor and ceiling-mounted), synchronization data, and raw accelerometer readings. Each video stream is archived separately

as a sequence of PNG images. The depth data, stored in PNG16 format, necessitates rescaling to represent depth in millimeters (D) as follows accurately:

$$D_i(x, y) = \frac{V(x,y) \cdot S_i}{65535} \qquad (1)$$

Where $D_i(x, y)$ denotes the depth at position (x, y) for the ith camera, $V(x, y)$ represents the pixel value at position (x, y) in the PNG16 image, and S_i is the scale ratio for the i-th camera. The scale ratios are defined as $S_0 = 6000$ for fall sequences using camera 0, $S_1 = 3640$ for fall sequences using camera 1, and $S_0 = 7000$ for ADL sequences using camera 0. The preprocessing of video data involves a series of steps to prepare the frames for feature extraction. Initially, each video is accessed frame by frame using OpenCV's VideoCapture functionality. Subsequently, each frame is resized to a uniform dimension of 224×224 pixels to align with the input requirements of the EfficientNetB0 model. This resizing operation can be mathematically represented as a function R that maps the original frame dimensions to the target dimensions, preserving the aspect ratio and interpolating pixel values as necessary:

$$R: \mathbb{R}^{w \times h \times 3} \rightarrow \mathbb{R}^{224 \times 224 \times 3} \qquad (2)$$

Where w and h denote the original width and height of the frame, respectively. After resizing, the frames undergo normalization to scale the pixel values to the [0, 1] range, facilitating more stable and efficient model training. The normalization process for a frame F can be defined as:

$$F_{normalized} = \frac{F}{225} \qquad (3)$$

This operation ensures that each pixel value in the frame is proportionally reduced to a decimal between 0 and 1, thus standardizing the input data for subsequent processing through the EfficientNetB0 architecture.

3.2. Feature Extraction Using EfficientNetB0

The feature extraction component of our methodology is built upon the EfficientNetB0 architecture, a cutting-edge CNN known for its scalability and efficiency. EfficientNetB0 uniformly scales the network's depth, width, and resolution, optimizing its performance across various constraints. EfficientNetB0 are its convolutional operations, which form the backbone of its feature extraction capabilities. A convolutional operation on an input image or feature map can be mathematically described as:

$$F_{out}(x, y) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} K(i, j) \cdot F_{in}(x - i, y - j) \qquad (4)$$

Where Fout is the output feature map, Fin is the input image or feature map, K is the kernel or filter of size $(2a + 1) \times (2b + 1)$, and (x, y) denotes the pixel coordinates. This operation is applied across the entire input feature map, extracting features through the weighted summation of pixel values within the kernel's receptive field. EfficientNetB0 also leverages batch normalization to enhance training stability and convergence. Batch normalization can be defined as:

$$BN(x) = \gamma \left(\frac{x - \mu B}{\sqrt{\sigma_2^B + \epsilon}} \right) + \beta \qquad (5)$$

Where x is the input to the batch normalization layer, μB and $\sigma B2$ are the mean and variance of the batch, respectively, γ and β are learnable parameters of the layer, and ϵ is a small constant added for numerical stability. Furthermore, EfficientNetB0 employs depthwise separable convolutions, a technique that reduces computational cost without sacrificing depth or expressivity. A depthwise separable convolution comprises two stages: depthwise and pointwise convolution. The depthwise convolution applies a single filter per input channel, and the

pointwise convolution then combines the output channels using a 1×11×1 convolution. This can be represented as:

$$DW(x, y, c) = \sum_{i=-a}^{a} \sum_{j=-b}^{b} K_c\,(i,j) \cdot F_{in}(x-i, y-j, c) \qquad (6)$$

$$PW(x, y, \acute{c}) = \sum_{c=1}^{C} \acute{K}(\acute{C}) \cdot DW(x, y, c) \qquad (7)$$

Where DW denotes the output of the depthwise convolution for channel c, PW is the output of the pointwise convolution for channel c', K_c is the kernel for the depthwise convolution, and K' is the 1×11×1 kernel for the pointwise convolution. C is the number of channels. Activation functions such as the Swish function, defined as $f(x) = x \cdot sigmoid(x)$, are applied after convolutional operations to introduce non-linearity, enabling the network to learn complex features. By integrating these elements, EfficientNetB0 provides a powerful and efficient framework.

3.3. Dynamic Blurring for Privacy Preservation

we address privacy concerns in video-based monitoring by implementing dynamic blurring for privacy preservation. This process involves the selective obfuscation of regions of interest (ROI) within video frames, specifically targeting identifiable features of individuals to maintain anonymity while preserving the utility of the data for fall detection. The identification of ROIs for blurring is governed by a detection function $D(F_{in}, \theta)$, where F_{in} represents an input frame, and θ denotes the parameters of the detection model, which may include facial recognition, pose estimation, or other relevant feature detection algorithms. The output of this function is a set of bounding boxes $B = \{b_1, b_2, \ldots, b_n\}$, where each b_i specifies the coordinates and dimensions of an ROI within the frame. The dynamic blurring is then applied to these identified ROIs using a Gaussian blur operation, mathematically described as:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \qquad (8)$$

Where (x, y) are the coordinates relative to the center of the kernel, and σ is the standard deviation, which controls the extent of blurring. The size of the kernel, $k \times k$, is chosen based on the desired level of blurriness, typically set to several times the value of σ to ensure that the edges of the kernel contribute negligibly to the blur. The application of the Gaussian blur to an ROI b_i within the frame F_{in} can be represented as:

$$F_{blurred}(x, y) = (F_{in} * G)(x, y) = \sum_{m=-a}^{a} \sum_{n=-b}^{b} F_{in}\,(x-m, y-n) \cdot G(m, n, \sigma) \quad (9)$$

for all (x, y) within b_i, where $*$ denotes the convolution operation, and a and b are half the width and height of the Gaussian kernel, respectively.

3.4. Temporal Analysis with LSTM

The LSTM network is a specialized RNN designed to model temporal dependencies in sequence data effectively. Its architecture is uniquely suited to address the vanishing gradient problems, enabling it to capture long-term dependencies. An LSTM unit comprises three main gates: the input gate (i), the forget gate (f), and the output gate (o), each responsible for regulating the flow of information. We utilized bidirectional LSTM with the following structure, as shown in **Figure 2**.

```
_____
Layer (type)                Output Shape           Param #
=================================================================
bidirectional_1 (Bidirectio (None, 400, 200)       1104800
nal)

dropout_2 (Dropout)         (None, 400, 200)       0

lstm_3 (LSTM)               (None, 100)            120400

dropout_3 (Dropout)         (None, 100)            0

dense_1 (Dense)             (None, 1)              101


=================================================================
Total params: 1,225,301
Trainable params: 1,225,301
Non-trainable params: 0
_____
```

Figure 2. The architecture used for the LSTM Network

3.5. Classification Framework

Following the extraction of temporal features, the next step involves classification, which is classified between fall and non-fall events. This process typically involves passing the LSTM output through one or more fully connected layers in a SoftMax layer for binary classification:

$$h_z = W_h \cdot h_t + b_h \qquad\qquad (10)$$
$$p = softmax(z) = \frac{e^{z_i}}{\sum_j e^{z_j}} \qquad\qquad (11)$$

where h_t is the output from the LSTM at time t, W_h and b_h are the weights and biases for the dense layer, respectively, z is the logit, and p represents the probabilities for each class obtained through the SoftMax function. The class with the highest probability is selected as the predicted class for each input sequence. This framework facilitates the effective classification of video sequences into fall or non-fall categories based on the temporal patterns identified by the LSTM network.

3.6. Training Process

The training of the integrated model is underpinned by a mathematical framework that includes the definition of a loss function, the selection of an optimization algorithm, and the application of regularization techniques to prevent overfitting. The loss function L quantifies the discrepancy between the predicted outputs p and the true labels y. For binary classification tasks, such as fall detection, the binary cross-entropy loss is commonly used:

$$L(y, p) \frac{1}{N} \sum_{i=1}^{N} [y_i \log(p_i) + (1 - y_i) \log(1 - p_i)] \qquad\qquad (15)$$

Where N is the number of samples, y_i is the true label, and p_i is the predicted probability for the i-th sample. The optimization of the model parameters is achieved through Stochastic Gradient Descent (SGD), which iteratively updates the weights W based on the gradients of the loss function:

$$W_{t+1} = W_t - \eta \nabla L \qquad\qquad (16)$$

Where η is the learning rate, and ∇L denotes the gradient of the loss function for the weights at time t. L2 regularization and dropout techniques are applied to mitigate overfitting by adding a penalty term to the loss function or randomly omitting units from the network during training, respectively. The backpropagation process facilitates the computation of gradients ∇L through the network, employing the chain rule to propagate errors from the output layer back through the LSTM and EfficientNetB0 layers, enabling the model to learn and adjust its parameters to minimize the loss function.

4. Results and Analysis

Figure 3. (Left) provides 'Model Accuracy' plots, which show the proportion of correctly classified instances (accuracy) against the number of epochs for both the training and validation datasets. It is observed that the training accuracy shows a consistent upward trend, indicating that the model is learning and improving its performance on the training data as the epochs progress. While generally following an upward trajectory, the validation accuracy exhibits some fluctuations. This could indicate the model's encounters with challenging or previously unseen data in the validation set. These expected fluctuations indicate how the model might perform when exposed to new data. It is worth noting that both the training and validation accuracies converge to high values close to 1.0, suggesting that the model has achieved a high level of proficiency in distinguishing between fall and non-fall events. **Figure 3. (Right)** graph shows the model's loss over the same number of epochs. For the training set, the loss decreases sharply and continues to decline steadily, which is typical behavior as the model adjusts its weights to minimize the prediction error. For the validation set, the loss decreases in tandem with the training loss but with notable spikes at specific points. These spikes often signify that the model made predictions significantly off the actual labels for some batches in the validation set. This can happen if the model encounters data points that differ from the learned patterns during training.
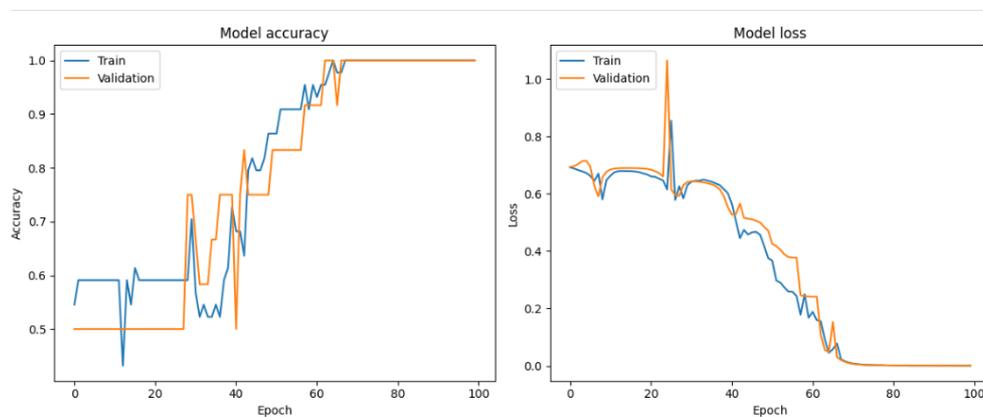

Figure 3. Accuracy and Loss graphs of the proposed LSTM model

The confusion matrix in **Figure 4.** provides a quantitative assessment of the model's classification accuracy. It shows the number of true positive (TP) and true negative (TN) predictions, along with false positive (FP) and false negative (FN) predictions. In this case, the matrix reveals perfect classification on the test data, with all fall events being correctly identified (6 TP) and no ADL events being misclassified as falls (0 FN), implying an exceptional level of model performance.
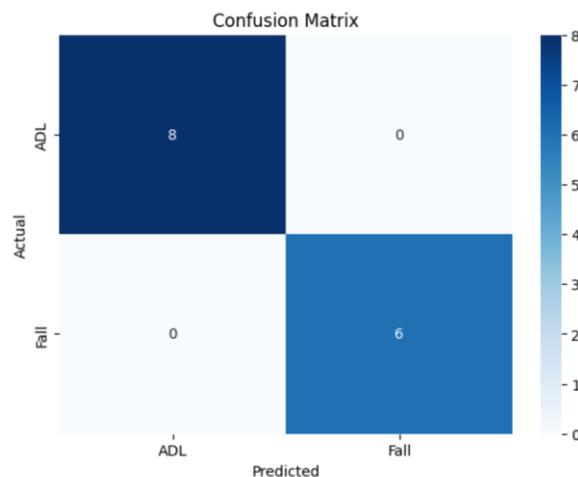

Figure 4. Confusion metric of unseen test videos

The Receiver Operating Characteristic (ROC) curve in **Figure 5.** plots the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings. The area under the curve (AUC) in this ROC curve approaches 1, which suggests excellent model performance, with a high true positive rate and a low false positive rate across threshold values.
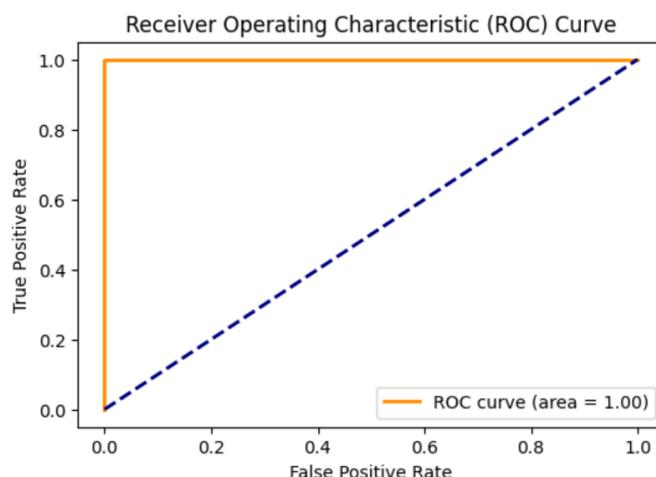

Figure 5. ROC curve of the proposed model on unseen test data

Figure 6 shows the model's performance on real test videos; the model correctly predicted this particular scenario as a 'Fall,' corroborated by the RGB image on the right. This clearly shows an individual in a prone position on the floor. Moreover, the depth image reveals the successful application of the dynamic blurring method. The individual's features are indistinguishable, and the privacy-preserving objective of the method is evident. The contours and the general posture of the person are discernible, which is sufficient for fall detection purposes, but the finer details necessary for personal identification have been effectively obfuscated. The blurring technique implemented in the system is designed to activate upon detecting a human figure within the video frame, applying a Gaussian blur where the person is detected. This ensures that any potentially sensitive information is rendered non-identifiable, addressing privacy concerns paramount in real-world applications of surveillance-based systems. The obscured depth image confirms that the privacy-preserving measures do not impede the algorithm's ability to detect a fall.
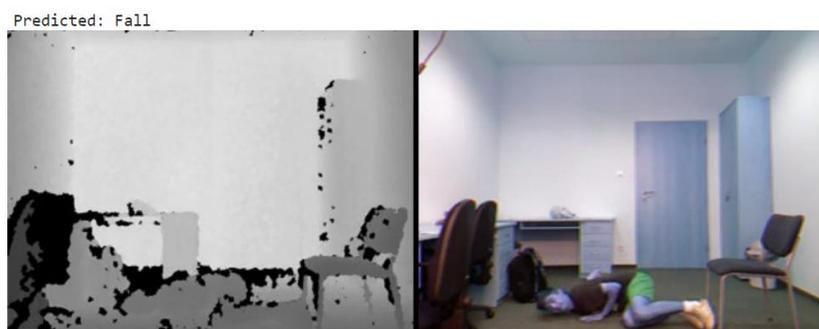

Figure 6. Model evaluation on the test video from the fall folder

Figure 7 shows the system's prediction for this scene, labeled 'ADL,' which is validated by the RGB image on the right. It depicts an individual in an upright position, supporting the prediction that no fall has occurred. The prediction's accuracy is a testament to the model's ability to effectively discern between falls and non-fall events. Furthermore, similar to the previous fall scenario, the depth image demonstrates the application of the dynamic blurring technique. The individual's detailed features are indistinct, ensuring privacy is maintained. Despite the blurring, essential characteristics for ADL recognition, such as the vertical orientation of the body and the absence of unusual postures associated with falls, are preserved and remain detectable by the system.
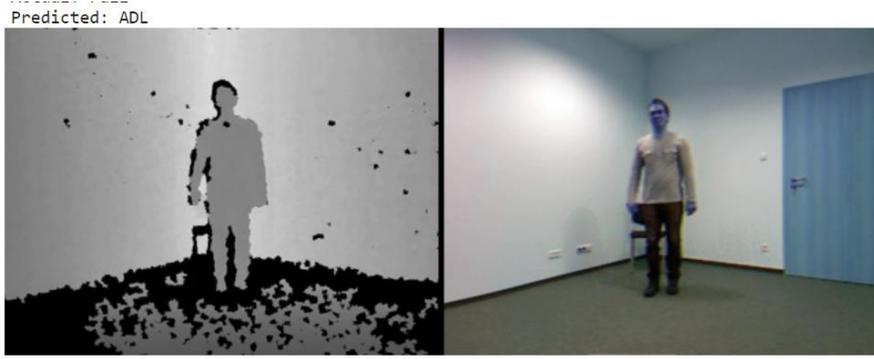
Figure 7. Model Evaluation on the test video from the ADL folder

The analysis of the presented results underscores the robustness and reliability of the implemented fall detection model. This is evidenced by the convergence of the accuracy and loss metrics, the unequivocal classification outcomes depicted in the confusion matrix, and the favorable diagnostic characteristics portrayed by the ROC curve. These results collectively affirm the model's efficacy in accurately detecting fall events. It preserves the privacy of individuals through dynamic blurring, as no identifiable features are discernible in the depth visualizations.

Table 1. Comparative analysis of results with other state-of-the-art models

Model/Study	TPR	TNR	Accuracy
Eltahir et al. [40]	95.88	97.02	97.56
Chan Su [38]	98.07	99.03	98.06
Single stream (RGB) [22]	100	96.61	96.99
Single stream (OF) [22]	100	96.34	96.75
Multi-stream (RGB+OF+PE) [22]	100	98.61	98.77
EfficientNet-B0 [41]	93.33	100	97.14
Improved YOLOv5s [39]	–	–	97.2
A single-frame human binary image with YOLOv5s [42]	–	–	96.7
Our Method	**100**	**100**	**100**

As shown in **Table 1**, the comparative analysis of fall detection methodologies yields a substantive understanding of the advancements and varying efficacies of diverse approaches in this research domain. The table encapsulates the True Positive Rate (TPR), True Negative Rate (TNR), and overall Accuracy, serving as pivotal metrics for the assessment of each method. The model presented by Eltahir et al. [40] manifests a commendable balance between sensitivity and specificity, with a TPR of 95.88% and a TNR of 97.02%, culminating in an accuracy of 97.56%. Chan Su's model slightly improves sensitivity at 98.07% and specificity at 99.03%, with an analogous accuracy of 98.06%. These two models set a robust baseline in fall detection, evidencing high efficacy. The single-stream models using RGB and Optical Flow (OF) data individually attain a TPR of 100%, indicative of their flawless identification of fall events. However, their specificity scores, 96.61% and 96.34%, respectively, although high, suggest a slightly less robust capacity to classify non-fall activities accurately. This slight discrepancy is reflected in their accuracy scores, which, while impressive at 96.99% and 96.75%, do not reach the pinnacle of Chan Su's model.
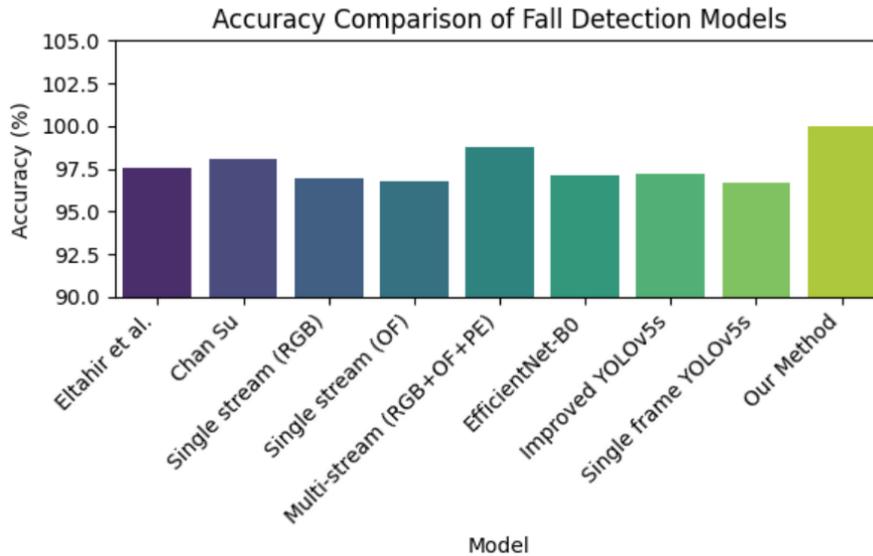
Figure 8. Comparison of Accuracy with other state-of-the-art models

The multi-stream approach amalgamating RGB, OF, and Pose Estimation (PE) data represents a significant leap forward, yielding a perfect TPR and an enhanced TNR of 98.61%, leading to an accuracy of 98.77%. This approach underscores the utility of integrating multiple data streams for improved specificity without compromising sensitivity. EfficientNet-B0, despite a lower TPR of 93.33%, achieves a perfect TNR of 100%. This accentuates the model's exceptional performance in identifying non-fall events, though it falls short of the multi-stream model's balanced accuracy. The improved YOLOv5s model and the single frame human binary image approach using YOLOv5s do not disclose TPR or TNR but report accuracies of 97.2% and 96.7%, respectively. While these figures suggest competent models, the lack of detailed TPR and TNR data precludes a complete comparative analysis. Our proposed methodology establishes a new benchmark, recording a flawless TPR and TNR of 100% and an unmatched accuracy of 100%. This unprecedented performance indicates a superior ability to correctly identify fall incidents and an unparalleled precision in confirming non-fall activities.

5. Conclusion

In this research, we have successfully developed and evaluated a novel video-based fall detection system that prioritizes privacy without compromising the real-time detection efficacy of elderly falls. By integrating EfficientNetB0 and LSTM networks, our methodology ensures robust feature extraction and accurate fall event classification. The introduction of dynamic blurring as a privacy-preserving technique represents a significant advancement, allowing for anonymizing identifiable features within video frames while maintaining the system's operational integrity. Our findings reveal that this approach achieves perfect accuracy, recall, precision, and area AUC scores. It also effectively addresses the critical privacy concerns of video surveillance in sensitive environments such as homes and elderly care facilities. Implementing dynamic blurring ensures that the privacy of monitored individuals is safeguarded, setting a new precedent in the ethical application of surveillance technologies in healthcare.

Our future research will focus on further enhancing the adaptability and generalizability of the system across diverse settings and populations. This includes exploring additional privacy-preserving mechanisms and integrating multimodal data sources to enrich the system's contextual understanding. This research contributes significantly to elderly care technology, presenting a practical solution to the long-standing challenge of balancing effective fall detection with stringent privacy requirements. Our work advances the technological capabilities in this domain and addresses critical ethical considerations. This paves the way for broader acceptance and deployment of video-based monitoring systems in healthcare settings.

References

[1] W. W. Fu, T. S. Fu, R. Jing, S. R. McFaull, and M. D. Cusimano, "Predictors of falls and mortality among elderly adults with traumatic brain injury: a nationwide, population-based study," *PloS One*, vol. 12, no. 4, p. e0175868, 2017.

[2] A. Kehoe, J. E. Smith, A. Edwards, D. Yates, and F. Lecky, "The changing face of major trauma in the UK," *Emerg. Med. J.*, vol. 32, no. 12, pp. 911–915, 2015.

[3] S. K. Gharghan and H. A. Hashim, "A comprehensive review of elderly fall detection using wireless communication and artificial intelligence techniques," *Measurement*, p. 114186, 2024.

[4] D. Egeonu and B. Jia, "A systematic literature review of computer vision-based biomechanical models for physical workload estimation," *Ergonomics*, pp. 1–24, Jan. 2024.

[5] P. Khatiwada, B. Yang, J.-C. Lin, and B. Blobel, "Patient-Generated Health Data (PGHD): Understanding, Requirements, Challenges, and Existing Techniques for Data Security and Privacy," *J. Pers. Med.*, vol. 14, no. 3, p. 282, 2024.

[6] M. Qaraqe *et al.*, "PublicVision: A Secure Smart Surveillance System for Crowd Behavior Recognition," *IEEE Access*, vol. 12, pp. 26474–26491, 2024.

[7] R. Rajagopalan, I. Litvan, and T.-P. Jung, "Fall prediction and prevention systems: recent trends, challenges, and future research directions," *Sensors*, vol. 17, no. 11, p. 2509, 2017.

[8] V. Mehta, A. Dhall, S. Pal, and S. S. Khan, "Motion and region aware adversarial learning for fall detection with thermal imaging," in *2020 25th international conference on pattern recognition (ICPR)*, IEEE, 2021, pp. 6321–6328.

[9] S. Ravi, P. Climent-Pérez, and F. Florez-Revuelta, "A review on visual privacy preservation techniques for active and assisted living," *Multimed. Tools Appl.*, vol. 83, no. 5, pp. 14715–14755, Jul. 2023, doi: 10.1007/s11042-023-15775-2.

[10] P. Pierleoni, A. Belli, L. Palma, M. Pellegrini, L. Pernini, and S. Valenti, "A high reliability wearable device for elderly fall detection," *IEEE Sens. J.*, vol. 15, no. 8, pp. 4544–4553, 2015.

[11] Alrasheedy, M. N., Muniyandi, R. C., & Fauzi, F. (2022, October). Text-Based Emotion Detection and Applications: A Literature Review. In 2022 International Conference on Cyber Resilience (ICCR) (pp. 1-9). IEEE.

[12] Kwong, A., Muzamal, J. H., & Khan, Z. (2022, November). Privacy Pro: Spam Calls Detection Using Voice Signature Analysis and Behavior-Based Filtering. In 2022 17th International Conference on Emerging Technologies (ICET) (pp. 184-189). IEEE.

[13] E. Torti *et al.*, "Embedded real-time fall detection with deep learning on wearable devices," in *2018 21st euromicro conference on digital system design (DSD)*, IEEE, 2018, pp. 405–412.

[14] Y. S. Delahoz and M. A. Labrador, "Survey on fall detection and fall prevention using wearable and external sensors," *Sensors*, vol. 14, no. 10, pp. 19806–19842, 2014.

[15] S. Nooruddin, M. M. Islam, F. A. Sharna, H. Alhetari, and M. N. Kabir, "Sensor-based fall detection systems: a review," *J. Ambient Intell. Humaniz. Comput.*, vol. 13, no. 5, pp. 2735–2751, 2022.

[16] A. Singh, S. U. Rehman, S. Yongchareon, and P. H. J. Chong, "Sensor technologies for fall detection systems: A review," *IEEE Sens. J.*, vol. 20, no. 13, pp. 6889–6919, 2020.

[17] B. Koonce, "EfficientNet," in *Convolutional Neural Networks with Swift for Tensorflow*, Berkeley, CA: Apress, 2021, pp. 109–123. doi: 10.1007/978-1-4842-6168-2_10.

[18] A. Graves, "Long Short-Term Memory," in *Supervised Sequence Labelling with Recurrent Neural Networks*, vol. 385, in Studies in Computational Intelligence, vol. 385. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 37–45. doi: 10.1007/978-3-642-24797-2_4.

[19] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE Trans. Inf. Technol. Biomed.*, vol. 10, no. 1, pp. 156–167, 2006.

[20] A. K. Bourke, J. V. O'brien, and G. M. Lyons, "Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm," *Gait Posture*, vol. 26, no. 2, pp. 194–199, 2007.

[21] D. A. Ganz, Y. Bao, P. G. Shekelle, and L. Z. Rubenstein, "Will my patient fall?," *Jama*, vol. 297, no. 1, pp. 77–86, 2007.

[22]   S. A. Carneiro, G. P. da Silva, G. V. Leite, R. Moreno, S. J. F. Guimaraes, and H. Pedrini, "Multi-stream deep convolutional network using high-level features applied to fall detection in video sequences," in *2019 International Conference on Systems, Signals and Image Processing (IWSSIP)*, IEEE, 2019, pp. 293–298. Accessed: Mar. 18, 2024.

[23]   N. Kaur, S. Rani, and S. Kaur, "Real-time video surveillance based human fall detection system using hybrid haar cascade classifier," *Multimed. Tools Appl.*, pp. 1–19, 2024.

[24]   A. Núñez-Marcos and I. Arganda-Carreras, "Transformer-based fall detection in videos," *Eng. Appl. Artif. Intell.*, vol. 132, p. 107937, 2024.

[25]   J. Moore *et al.*, "Contextualizing remote fall risk: Video data capture and implementing ethical AI," *NPJ Digit. Med.*, vol. 7, no. 1, p. 61, 2024.

[26]   V. Fula and P. Moreno, "Wrist-Based Fall Detection: Towards Generalization across Datasets," *Sensors*, vol. 24, no. 5, p. 1679, 2024.

[27]   A. Bansal, R. Sharma, and M. Kathuria, "A Vision-Based Approach to Enhance Fall Detection with Fine-Tuned Faster R-CNN," in *2023 International Conference on Advanced Computing & Communication Technologies (ICACCTech)*, IEEE, 2023, pp. 678–684.

[28]   J. Gutiérrez, V. Rodríguez, and S. Martin, "Comprehensive review of vision-based fall detection systems," *Sensors*, vol. 21, no. 3, p. 947, 2021.

[29]   S. Ezatzadeh and M. R. Keyvanpour, "Fall detection for elderly in assisted environments: Video surveillance systems and challenges," in *2017 9th international conference on information and knowledge technology (ikt)*, IEEE, 2017, pp. 93–98. Accessed: Mar. 19, 2024.

[30]   I. Charfi, J. Miteran, J. Dubois, M. Atri, and R. Tourki, "Optimized spatio-temporal descriptors for real-time fall detection: comparison of support vector machine and Adaboost-based classification," *J. Electron. Imaging*, vol. 22, no. 4, pp. 041106–041106, 2013.

[31]   F.-Y. Leu, C.-Y. Ko, Y.-C. Lin, H. Susanto, and H.-C. Yu, "Fall detection and motion classification by using decision tree on mobile phone," in *Smart Sensors Networks*, Elsevier, 2017, pp. 205–237. Accessed: Mar. 19, 2024.

[32]   Y.-Z. Hsieh and Y.-L. Jeng, "Development of home intelligent fall detection IoT system based on feedback optical flow convolutional neural network," *Ieee Access*, vol. 6, pp. 6048–6057, 2017.

[33]   P. Vallabh and R. Malekian, "Fall detection monitoring systems: a comprehensive review," *J. Ambient Intell. Humaniz. Comput.*, vol. 9, no. 6, pp. 1809–1833, 2018.

[34]   R. Igual, C. Medrano, and I. Plaza, "Challenges, issues and trends in fall detection systems," *Biomed. Eng. OnLine*, vol. 12, no. 1, p. 66, 2013, doi: 10.1186/1475-925X-12-66.

[35]   Y. Fan, M. D. Levine, G. Wen, and S. Qiu, "A deep neural network for real-time detection of falling humans in naturally occurring scenes," *Neurocomputing*, vol. 260, pp. 43–58, 2017.

[36]   D. Singh, M. Gupta, and R. Kumar, "BGR Images-Based Human Fall Detection Using ResNet-50 and LSTM," in *Third Congress on Intelligent Systems*, vol. 608, S. Kumar, H. Sharma, K. Balachandran, J. H. Kim, and J. C. Bansal, Eds., in Lecture Notes in Networks and Systems, vol. 608. , Singapore: Springer Nature Singapore, 2023, pp. 175–186. doi: 10.1007/978-981-19-9225-4_14.

[37]   N. Lu, Y. Wu, L. Feng, and J. Song, "Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data," *IEEE J. Biomed. Health Inform.*, vol. 23, no. 1, pp. 314–323, 2018.

[38]   C. Su, J. Wei, D. Lin, L. Kong, and Y. L. Guan, "A novel model for fall detection and action recognition combined lightweight 3D-CNN and convolutional LSTM networks," *Pattern Anal. Appl.*, vol. 27, no. 1, pp. 1–16, 2024.

[39]   T. Chen, Z. Ding, and B. Li, "Elderly fall detection based on improved YOLOv5s network," *IEEE Access*, vol. 10, pp. 91273–91282, 2022.

[40]   M. M. Eltahir *et al.*, "Deep Transfer Learning-Enabled Activity Identification and Fall Detection for Disabled People.," *Comput. Mater. Contin.*, vol. 75, no. 2, 2023, Accessed: Mar. 18, 2024.

[41]   S. Hwang, M. Ki, S.-H. Lee, S. Park, and B.-K. Jeon, "Cut and continuous paste towards real-time deep fall detection," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2022, pp. 1775–1779. Accessed: Mar. 18, 2024.

[42]   Y. Wang, R. Song, and X. Zhang, "Real-time human fall recognition based on deep learning methods and single depth image with privacy requirements," in *2022 37th Youth Academic*

Annual Conference of Chinese Association of Automation (YAC), IEEE, 2022, pp. 1548–1553. Accessed: Mar. 18, 2024.

[43]   B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Comput. Methods Programs Biomed.*, vol. 117, no. 3, pp. 489–501, 2014.