

Towards Robust Neuromorphic/Event Vision-Based Fall Detection: Cross-Dataset Generalisation Benchmarking

Muhammad Zeeshan Masood^a, Muhammad Hamza Zafar^b, Syed Kumayl Raza Moosavi^b, Florenc Demrozi^c, Furqan Shaukat^d and Filippo Sanfilippo^b

^aUniversité Jean-Monnet-Saint-Étienne, Saint-Étienne, 42100, France

^bDepartment of Engineering Sciences, University of Agder, Grimstad, 4879, Norway

^cDepartment of Electrical Engineering and Computer Science, University i Stavanger, Stavanger, 4021, Norway

^dElectrical Engineering Department, UET Taxila, taxila, 47050, Pakistan

ARTICLE INFO

Keywords:

Neuromorphic/Event Vision
Fall Detection
Cross-Dataset Generalisation
Privacy Preserving

ABSTRACT

Fall detection is a critical safety concern in many real-world environments, including smart homes, healthcare and assisted living facilities, workplaces, and other indoor/outdoor environments where individuals may be exposed to situations that increase the risk of accidental falls. Vision-based systems have emerged as a promising solution due to their non-intrusive nature and strong recognition performance. However, traditional RGB-based approaches suffer from limitations, including sensitivity to lighting conditions, occlusions, and privacy concerns. In contrast, event cameras capture asynchronous brightness changes with high temporal resolution and low latency, making them more robust in challenging conditions. Despite this potential, progress in event-based fall detection is limited by the lack of large-scale datasets and standardized evaluation protocols. In this paper, we present a benchmarking study based on four widely used RGB fall detection datasets—CAUCA Fall, GMDCSA24, Fall Vision, and Multiple Cameras Fall Dataset—converted into event data using the Metavision Simulator. These datasets are used to train and evaluate nine different deep learning models (i.e., I3D, 3DCNN, ConvLSTM, MobileNetV3-LSTM, TCN, SwinTransformer, TimeSFormer, EfficientViT, and TemporalMotionNet). To evaluate cross-domain generalization, the trained models are tested on a real-world event dataset, the Pedestrian Detection, Action Recognition, and Fall Detection (PAF) dataset. Our study highlights the challenges of simulation-to-real transfer in event-based fall detection and provides a comparative evaluation across synthetic and real event datasets. The results show that while TemporalMotionNet achieves high performance on synthetic data (98.65% accuracy), performance drops significantly on real data, with a 17.25% gap (81.4% accuracy on PAF). The converted datasets, together with the evaluation protocol, are publicly released (<https://github.com/AIBCRG/EventFallDetection>) to foster further research in this domain.

1. Introduction

Falls represent a pervasive and serious safety hazard across diverse populations and environments, including clinical settings (hospitals, assisted living facilities), residential and smart homes, athletic and recreational activities, and industrial and workplace settings, frequently resulting in significant injuries, functional decline, loss of independence, and substantial healthcare and economic costs (James et al., 2020). Consequently, automatic fall detection has become a long-standing research topic, with approaches spanning wearable sensors, ambient sensing, and vision-based systems (Xeferis et al., 2021). Among these, vision-based fall detection has attracted sustained interest due to its ability to capture rich spatio-temporal information without requiring users to wear or maintain dedicated devices (Gutiérrez et al., 2021; Singh et al., 2020). Early camera-based methods relied

on silhouette extraction (Mirmahboub et al., 2012), posture analysis (Lai et al., 2010), and motion history features (Feng et al., 2017), while more recent systems employ deep learning architectures to model complex temporal dynamics in video streams (Auvinet, Multon, et al., 2010; Espinosa et al., 2019).

Despite significant progress, conventional RGB camera-based fall-detection systems still face persistent limitations. Performance often degrades under challenging lighting conditions, motion blur, and occlusions, while privacy concerns restrict deployment in real-world healthcare and residential settings (Asif et al., 2020; Gutiérrez et al., 2021). Depth cameras alleviate some illumination-related issues but remain sensitive to occlusion and have limited operational range, and their use still raises privacy concerns due to explicit shape reconstruction (Gallego et al., 2020). These challenges motivate the exploration of alternative sensing modalities that can offer robustness, efficiency, and privacy-aware perception.

In contrast, event cameras (also referred to as neuromorphic vision sensors) represent a fundamentally different sensing paradigm. Instead of recording full image frames at fixed intervals, event cameras asynchronously report per-pixel brightness changes with microsecond temporal resolution, high dynamic range, and low latency (Rebecq et al.,

Muhammad Zeeshan Masood and Muhammad Hamza Zafar Contributed Equally in this Work.

*Corresponding author: Filippo Sanfilippo

✉ muhammad.zeeshan.masood@etu.univ-st-etienne.fr (M.Z. Masood); muhammad.h.zafar@uia.no (M.H. Zafar); syed.k.moosavi@uia.no (S.K.R. Moosavi); florenc.demrozi@uis.no (F. Demrozi); furqan.shoukat@uettaxila.edu.pk (F. Shaukat); filippo.sanfilippo@uia.no (F. Sanfilippo)
ORCID(s):

2018). These properties make event-based vision particularly attractive for monitoring fast, subtle human motion under variable lighting, while the absence of texture and appearance information inherently reduces privacy leakage. As a result, event-based perception has gained increasing attention in robotics, autonomous systems, and human activity recognition (Hu et al., 2021; Rebecq et al., 2018).

However, the adoption of event-based vision for fall detection remains constrained by the limited availability of large-scale, labeled neuromorphic datasets. While RGB-based fall detection benefits from multiple public benchmarks and standardized evaluation protocols, equivalent resources for event-based fall detection are scarce and fragmented (Zhang et al., 2024; Zhu et al., 2018). Such data scarcity poses a significant barrier to training deep models and conducting reproducible benchmarking. In such a context, event simulation frameworks—particularly including RGB-to-event conversion tools—have therefore emerged as a practical solution to scale event-based datasets by converting conventional videos into synthetic event streams that approximate real sensor output (Perot et al., 2020; Rebecq et al., 2019).

In this work, we investigate whether event-based fall detection models trained on RGB-derived simulated event data can generalize effectively to real event-camera data. Specifically, we convert multiple established RGB fall detection benchmarks into event-based datasets using a realistic video-to-event simulation pipeline, train fall detection models under consistent protocols, and evaluate their generalisation on a purely event-camera-based fall detection dataset. By releasing the converted datasets publicly, this work aims to establish a standardized benchmarking framework for neuromorphic fall detection and to provide the first systematic study of cross-dataset and sim-to-real generalization in this domain.

2. Literature Review

A structured literature search was conducted on Scopus using the query (*“event camera” OR “neuromorphic camera” OR “event-based vision” OR DVS OR “dynamic vision sensor”*) AND (*“fall detection” OR “fall recognition” OR “fall monitoring”*), with the goal of identifying all relevant studies at the intersection of neuromorphic vision and fall detection. Despite the intentionally broad search formulation, only a limited number of publications were retrieved, indicating that event-based fall detection remains a relatively nascent and under-explored domain, as shown in Table 1. The existing literature is largely centered around a small set of native event-camera datasets, such as DVS UR Fall and DVSFall, which have enabled initial feasibility studies in controlled environments (Prasad et al., 2022, 2023). These works typically employ 3D convolutional neural networks or hybrid deep learning pipelines and report strong within-dataset performance, often exceeding 97% accuracy, particularly in binary fall detection scenarios. In addition, benchmarking efforts have explored adapting conventional

video-based architectures—including MViT, I3D, and Slow-Fast—to event-based representations, achieving competitive results but still confined to within-dataset evaluation settings (Krishnan & Krishnan, 2022). Complementing these efforts, foundational contributions such as the Pedestrian Detection, Action Recognition, and Fall Detection (PAF) dataset provide annotated event-based fall sequences across multiple activity classes, supporting early-stage dataset development in this domain (Miao et al., 2019).

To mitigate the scarcity of large-scale event datasets, several studies have adopted RGB-to-event simulation pipelines, where conventional datasets such as URFall and Le2i are converted into synthetic event streams (Liang & ongsheng Ma, 2024). These approaches often integrate spiking neural networks for preprocessing, followed by object detection frameworks like YOLOv8 and temporal modeling using recurrent architectures such as LSTMs, enabling end-to-end real-time pipelines. Notably, such methods frequently report extremely high accuracy—sometimes reaching 100%—but these results are typically limited to simulated environments and lack validation on real event-camera data. Alongside this, neuromorphic computing paradigms have been investigated to better exploit the asynchronous nature of event streams, with studies exploring spiking neural networks (SNNs) and recurrent SNNs (RSNNs) as energy-efficient alternatives to conventional deep learning models (Wang et al., 2023). However, comparative analyses suggest that while SNN-based approaches offer computational efficiency, they often underperform relative to CNN-based counterparts in terms of accuracy. Hybrid frameworks combining CNN feature extraction with spiking representations have also been proposed to balance performance and efficiency (Prasad et al., 2023), further reflecting the ongoing exploration of suitable architectures for event-based data. Despite these advances, a consistent trend across the literature is the transformation of event data into frame-like representations to fit conventional models, rather than fully leveraging its native asynchronous characteristics. Moreover, most studies evaluate performance under within-dataset settings, with limited attention to cross-dataset generalisation or sim-to-real transfer, thereby restricting insights into robustness and real-world applicability (Krishnan & Krishnan, 2022; Liang & ongsheng Ma, 2024; Miao et al., 2019; Prasad et al., 2022, 2023; Wang et al., 2023).

2.1. Research Gaps and Contributions

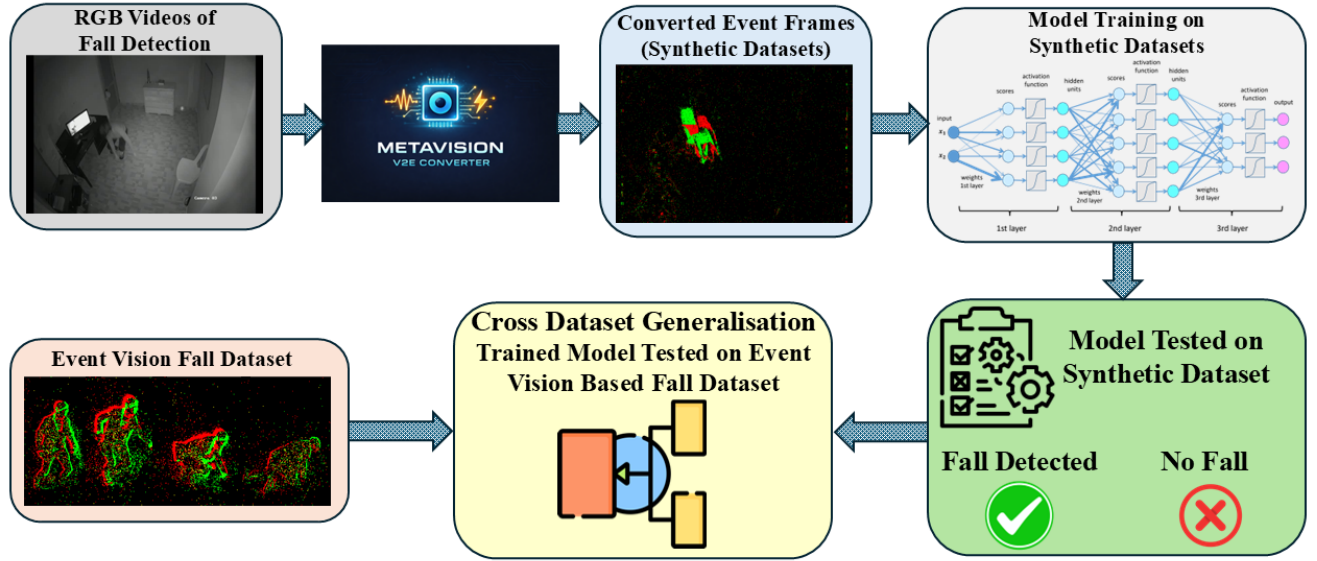
From the above review, several key research gaps emerge.

1. There is no standardized, large-scale benchmarking framework for event-based fall detection comparable to existing RGB benchmarks. While multiple RGB datasets enable reproducible evaluation and fair comparison, event-based fall detection research remains fragmented due to limited data availability and inconsistent protocols.
2. Cross-dataset and sim-to-real generalisation in neuromorphic fall detection is largely unexplored. Although event simulation is widely used, prior studies rarely

Table 1

Comparison of event-based fall detection literature and the proposed cross-dataset benchmarking framework.

Work	Year	Event Data	Dataset(s)	Model / Pipeline	Scope	Reported Performance
(Krishnan & Krishnan, 2022)	2024	Native event	Neuromorphic fall + action benchmark	Conventional video backbones (MViT, I3D, SlowFast family)	Within-dataset benchmarking	Best Acc $\approx$ 95.8%
(Prasad et al., 2022)	2022	Native event	DVS UR Fall	Lightweight 3D-CNN (edge-oriented)	Real-time binary detection	Acc 98.6%; Sens 96.7%; Spec 100%
(Liang & ongsheng Ma, 2024)	2024	Simulated events	URFall + Le2i (RGB→event)	SNN preprocessing + YOLOv8 + LSTM	Real-time pipeline demo	Acc 100% (reported)
(Prasad et al., 2023)	2021	Native event	DVSFall (multi-view dataset)	Hybrid 3D-CNN + SNN pipeline	Dataset within-dataset study	Acc 97.84%
(Miao et al., 2019)	2019	Native event	Pedestrian Detection, Action Recognition, and Fall Detection (PAF) dataset (4 classes)	Dataset + encoding framework	Dataset contribution	N/A
(Wang et al., 2023)	2023	Simulated events	Wearable RGB (single subject)	RSNN vs LSTM comparison	Efficiency trade-off study	CNN: 99.7%; RSNN: 76–81%
This work	2024	Simulated + Native event	4 RGB datasets → event + PAF native	TemporalMotionNet + cross benchmarking	Standardised sim-to-real benchmark	Synthetic: 98.65%; Real: 81.4%

**Figure 1:** Graphical Abstract of the Proposed Cross-Dataset generalisation Framework for Fall Detection using Neuromorphic/Event Camera

evaluate whether models trained on simulated events can transfer to real event-camera data, particularly across multiple datasets.

3. The role of dataset diversity in improving the robustness of event-based fall detection remains unclear. In RGB-based fall detection, training on diverse datasets has been shown to improve generalisation, yet this question has not been systematically addressed in the event-based domain.

Motivated by these gaps, this paper makes the following contributions:

- **RGB-to-Event Fall Detection Benchmark Suite:** We convert multiple established RGB fall detection datasets into event-based datasets using a realistic video-to-event simulation pipeline and release them publicly to support reproducible benchmarking.

Table 2
Comprehensive Comparison of Fall Detection Event-Based Datasets

Dataset	GMDCSA-2 (Alam, 2024; Alam et al., 2023, 2024)	Multiple Cameras (Auvinet, Rougier, et al., 2010)	CAUCAFall (Eraso et al., 2022; Guerrero et al., 2022)	FallVision (Rahman et al., 2024, 2025)	PAF (Miao et al., 2019)
Total Videos/Sequences	160	24	100	11,732	180
Fall Samples	81	Varies	50	6,002	180*
Non-Fall Samples	79	Varies	50	5,730	15
Subjects	4	1	10	58	346 × 260
Resolution	0.92 MP (30 FPS)	Variable	1080 × 960 (23 FPS)	720p/1080p (30 FPS)	DAVIS346redColor
Camera Type	Event Camera	8 IP Cameras	HIKVISION IR	Consumer Devices	Empty Office
Recording Environments	3 Home Environments	1 Controlled Room	Uncontrolled	Various	4 Classes**
Fall Categories	3 Types	1 (with 4 phases)	5 Types	3 Positions	Action Labels
ADL Categories	5 Categories	9 Position States	5 Categories	N/A	Basic Action Class
Annotations	Frame-level CSV	3D + Multi-camera	Per-frame YOLO Binary	17-point Pose	Neuromorphic sensor
Annotation Detail	Activity Boundaries	Ground Truth Positions	20,002 Annotated Frames	Keypoint Coordinates	Preddefined actions
Key Features	Multiple clothing/lighting	Calibrated 3D reconstruction	IR + RGB, Realistic	Large-scale, Pose keypoints	
Challenges Included	Variable conditions	Compression, shadows, occlusion	Variable lighting, occlusion	Multi-angle perspectives	

- **Cross-Dataset Benchmarking Protocol:** We establish a consistent training and evaluation protocol for event-based fall detection, including single-dataset and multi-dataset training scenarios.
- **Sim-to-Real Generalisation Study:** We conduct a systematic evaluation of models trained on simulated event data by testing them on a purely event-camera-based fall detection dataset, quantifying the generalisation gap and identifying effective training strategies.

The graphical description of the proposed work is shown in Fig.1. **To the best of our knowledge, no prior work provides a standardized cross-dataset and sim-to-real benchmarking framework for event-based fall detection under a unified evaluation protocol.**

3. Dataset Description and Methodology

For cross-dataset benchmarking and evaluation, we have converted 4 benchmark RGB datasets into Event data, and the description of these RGB datasets is presented in Table 2. The GMDCSA-24, Multiple Cameras, CAUCAFall, and FallVision RGB datasets are converted to event-stream format using the Metavision Video-to-Event (V2E) converter with standard DVS parameters, enabling the generation of synthetic event data from diverse RGB sources. GMDCSA-24 comprises 160 sequences (81 falls, 79 ADL) from 4 subjects with frame-level temporal annotations, offering a compact dataset (1.95 GB) suitable for edge deployment scenarios (Alam, 2024; Alam et al., 2023, 2024). The Multiple Cameras dataset provides 24 calibrated multi-camera scenarios with 9 position states and explicit temporal phase annotations for fall decomposition (Auvinet, Rougier, et al., 2010). CAUCAFall contains 100 videos with 20,002 per-frame binary annotations from 10 subjects recorded in uncontrolled environments, capturing realistic challenges such as variable lighting and occlusions (Eraso et al., 2022; Guerrero et al., 2022). FallVision, the largest dataset, comprises 11,732 videos from 58 subjects with detailed 17-point pose keypoint annotations across multiple recording angles and positions (Rahman et al., 2024, 2025). For cross-dataset generalisation, we used the native event camera-based dataset, i.e., the Pedestrian Detection, Action Recognition, and Fall Detection (PAF) dataset in which 180 neuromorphic recordings captured natively with the DAVIS346redColor event camera from 15 subjects performing four action classes (falling, bending, slumping, tying-shoes) (Miao et al., 2019).

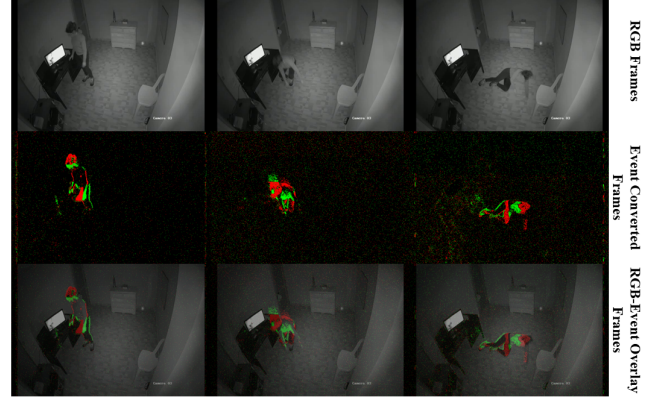


Figure 2: Temporal Sequence Samples of RGB to Event Converted Data

3.1. Event Camera Dataset Conversion Process

The RGB video datasets were converted to neuromorphic event streams using a video-to-event simulator based on the Metavision SDK. The conversion process operates as follows: for each input video file (supporting both .avi and .mp4 formats), the frame sequence is processed temporally by an event simulator that models the response of a Dynamic Vision Sensor (DVS) by computing per-pixel intensity changes between consecutive frames. When the absolute pixel intensity change exceeds a configurable contrast sensitivity threshold (default $C_p = C_n = 0.35$), an event is generated with coordinates (x, y) , timestamp t , and polarity $p \in \{0, 1\}$ representing brightness increase (ON events) or decrease (OFF events). The simulator accumulates events into fixed-size buffers of $n_{events} = 15,000$ events per processing window, with configurable refractory period (1 ms), shot noise rate (10 Hz), and temporal filtering (cutoff frequency). For each event buffer, three synchronized outputs are generated and saved: (1) the original RGB frame as a PNG image, (2) a custom event visualization with ON events rendered as green pixels and OFF events as red pixels, and (3) the raw event data saved as NumPy arrays containing the (x, y, t, p) tuples. This conversion preserves the temporal dynamics and motion information inherent in the original videos while generating sparse, asynchronous event representations suitable for event-based deep learning architectures, with output directory structures mirroring the source folder organization to maintain dataset hierarchy and enable scalable processing of large multi-dataset collections.

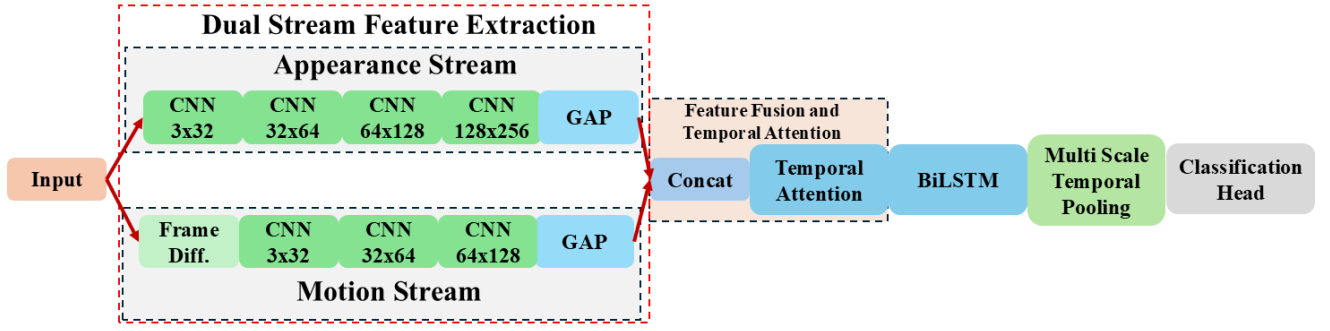


Figure 3: Detailed Architecture of Proposed TemporalMotionNet Model

The sample image of the rgb to event converted data is presented in Fig. 2. Considering the extensive processing time i.e. approximately 8 weeks and computational resources required for large-scale RGB-to-event conversion, all generated datasets are publicly released to accelerate research progress and avoid redundant data preparation efforts.

3.2. Proposed Model

To assess cross-dataset benchmarking, we first evaluate several state-of-the-art models (Section IV-A). Building on the observed limitations, we propose TemporalMotionNet, a lightweight dual-stream deep learning architecture tailored for real-time fall detection on event-based neuromorphic data. The model combines explicit motion encoding through temporal differencing with appearance-based spatial feature extraction, integrated through temporal attention mechanisms and bidirectional LSTM layers, as shown in Fig. 3. The architecture achieves superior performance while maintaining fewer than 3M parameters, enabling deployment on resource-constrained edge devices (Raspberry Pi, Jetson Nano).

3.2.1. Dual-Stream Feature Extraction

The model processes input video sequences $\mathbf{X} \in \mathbb{R}^{B \times C \times T \times H \times W}$ where B is batch size, $C = 3$ color channels, $T = 20$ temporal frames, and $(H, W) = (112, 112)$ spatial dimensions.

Appearance Stream: For each frame $\mathbf{x}_t$ at time t , we extract spatial features using a compact convolutional encoder:

$$\mathbf{a}_t = \text{AvgPool}(\text{Conv}_4(\dots \text{Conv}_1(\mathbf{x}_t) \dots)) \in \mathbb{R}^{256} \quad (1)$$

where Conv_i denotes successive convolutional blocks with $(32 \rightarrow 64 \rightarrow 128 \rightarrow 256)$ channels and batch normalization, followed by global average pooling to produce fixed-size appearance features $\mathbf{a}_t$ for all T frames.

Motion Stream: Explicit motion information is extracted via per-pixel temporal differencing:

$$\mathbf{d}_t = \mathbf{x}_{t+1} - \mathbf{x}_t \quad \forall t \in [1, T-1] \quad (2)$$

Each difference frame $\mathbf{d}_t$ is encoded through a separate lightweight convolutional stream:

$$\mathbf{m}_t = \text{AvgPool}(\text{Conv}_3(\dots \text{Conv}_1(\mathbf{d}_t) \dots)) \in \mathbb{R}^{128} \quad (3)$$

Motion features are zero-padded temporally to match the appearance stream: $\bar{\mathbf{m}}_1 = \mathbf{0}, \bar{\mathbf{m}}_{t>1} = \mathbf{m}_{t-1}$.

3.2.2. Feature Fusion and Temporal Attention

Appearance and motion features are concatenated at each time step:

$$\mathbf{f}_t = [\mathbf{a}_t \oplus \bar{\mathbf{m}}_t] \in \mathbb{R}^{384} \quad (4)$$

where $\oplus$ denotes concatenation. A temporal attention module weights frames based on their discriminative importance for fall detection:

$$\begin{aligned} \alpha_t &= \text{Softmax}(\text{MLP}(\mathbf{f}_t)) \\ &= \frac{\exp(w_2 \tanh(w_1 \mathbf{f}_t + b_1) + b_2)}{\sum_{t'=1}^T \exp(w_2 \tanh(w_1 \mathbf{f}_{t'} + b_1) + b_2)} \end{aligned} \quad (5)$$

Attended features are obtained as:

$$\hat{\mathbf{f}}_t = \alpha_t \odot \mathbf{f}_t \quad (6)$$

where $\odot$ denotes element-wise multiplication. This mechanism enables the model to focus on critical motion patterns during falls while suppressing non-informative frames (e.g., static ADL activities).

3.2.3. Temporal Modeling with Bidirectional LSTM

The attended feature sequence $\{\hat{\mathbf{f}}_1, \dots, \hat{\mathbf{f}}_T\}$ is processed by a bidirectional LSTM to capture long-range temporal dependencies:

$$\mathbf{h}_t^{\text{forward}}, \mathbf{h}_t^{\text{backward}} = \text{BiLSTM}(\hat{\mathbf{f}}_t) \quad (7)$$

where each direction has hidden dimension 256. The forward LSTM processes frames chronologically ($t : 1 \rightarrow T$) to capture temporal progression of falls, while the backward LSTM processes frames in reverse ($t : T \rightarrow 1$) to capture anticipatory patterns. Concatenating both directions yields:

$$\mathbf{h}_t = [\mathbf{h}_t^{\text{forward}} \oplus \mathbf{h}_t^{\text{backward}}] \in \mathbb{R}^{512} \quad (8)$$

3.2.4. Multi-Scale Temporal Aggregation

We employ dual pooling strategies to capture both rapid motion events and long-duration trends:

$$\mathbf{s}_{\text{short}} = \text{MaxPool}(\{\mathbf{h}_1, \dots, \mathbf{h}_T\}) \in \mathbb{R}^{512} \quad (9)$$

$$\mathbf{s}_{\text{long}} = \text{AvgPool}(\{\mathbf{h}_1, \dots, \mathbf{h}_T\}) \in \mathbb{R}^{512} \quad (10)$$

Short-term pooling captures peak motion values characteristic of impact events, while long-term pooling captures overall temporal trends. Combined features are:

$$\mathbf{s} = [\mathbf{s}_{\text{short}} \oplus \mathbf{s}_{\text{long}}] \in \mathbb{R}^{1024} \quad (11)$$

Table 3

TemporalMotionNet parameter breakdown by component

Component	Parameters	Percentage
Appearance Stream	485,152	14.7%
Motion Stream	289,408	8.8%
Temporal Attention	50,113	1.5%
BiLSTM (2 layers, 256 hidden)	1,640,448	49.7%
Classification Head	667,394	20.2%
Batch Normalization	91,520	2.8%
Total	3,304,035	100%

Table 4

Hyperparameters for TemporalMotionNet Model

Hyperparameter	Value
Appearance Stream	Conv2D (32, 64, 128, 256 channels)
Motion Stream	Conv2D (32, 64, 128 channels)
LSTM Hidden Size	256
LSTM Layers	2 (Bidirectional)
Dropout Rate	0.3
Number of Classes	2 (Fall, Non-Fall)
Global Pooling	Adaptive AvgPool2D
Temporal Attention Input	384 (Appearance + Motion)
LSTM Input Size	384

3.2.5. Classification Head

Final predictions are obtained through a lightweight multi-layer perceptron:

$$\mathbf{z} = \text{MLP}(\mathbf{s}) = \text{Linear}_2(\text{ReLU}(\text{Dropout}(\text{Linear}_1(\mathbf{s})))) \quad (12)$$

where $\text{Linear}_1 : \mathbb{R}^{1024} \rightarrow \mathbb{R}^{256}$ and $\text{Linear}_2 : \mathbb{R}^{256} \rightarrow \mathbb{R}^2$ map combined temporal features to class logits. Binary classification (Fall vs. Non-Fall) is achieved through:

$$\hat{y} = \arg \max \text{softmax}(\mathbf{z}) \quad (13)$$

The architecture and the parameter count is presented in Table 3 and Table 4.

4. Results and Discussion

4.1. Benchmark Models

For comparative analysis, we evaluate several established neural network architectures spanning different design paradigms for video and event-based action recognition. 3DCNNs and I3D (Caruana et al., 2017; Ji et al., 2012) capture spatiotemporal features through volumetric convolutions with ImageNet pre-training, ConvLSTM (Shi et al., 2015) preserves spatial structure while modeling temporal sequences through recurrent convolutional cells, and MobileNetV3-LSTM (Howard et al., 2019) provides a lightweight hybrid approach for edge deployment. Temporal Convolutional Networks (TCN) (Bai et al., 2018) employ dilated 1D convolutions to capture multi-scale temporal patterns with efficient parallel processing, while Swin Transformer and TimeSformer (Bertasius et al., 2021; Liu

et al., 2021) leverage hierarchical vision transformers with factorized attention mechanisms to balance computational efficiency with global context modeling. EfficientViT (Cai et al., 2022) further optimizes transformer architectures for resource-constrained environments through lightweight backbones and cascaded attention. These models represent diverse approaches—from CNN-based feature learning to recurrent temporal modeling and transformer-based attention mechanisms—enabling comprehensive evaluation of different inductive biases for action recognition on event-based and frame-based video data. All nine models evaluated in this study were trained and tested under completely identical experimental conditions to ensure a fair and unbiased comparison, as shown in Table 5. Specifically, all models received the same input representations, were trained on the same combined synthetic event dataset, used the same 80/20 clip-level random split, and were evaluated on the same real PAF dataset for cross-dataset generalization.

4.2. Ablation Study of Proposed Model

The ablation study systematically evaluates the contribution of each architectural component in TemporalMotionNet by removing one module at a time while keeping all other components intact, as presented in Table 6. M1 represents the complete TemporalMotionNet architecture with all components, serving as the upper bound for comparison. M2 removes the Appearance Stream, relying solely on the Motion Stream for spatial feature extraction, which means the model loses the ability to capture static spatial context from individual frames. M3 removes the Motion Stream,

Table 5
Unified Experimental Configuration Applied to All Models

Parameter	Value
Clip Length	20 frames
Input Spatial Resolution	112 × 112
Train/Test Split Ratio	80/20
Random Seed	42
Batch Size	16
Number of Classes	2 (Fall / Non-Fall)
Normalization Mean	(0.45, 0.45, 0.45)
Normalization Std	(0.225, 0.225, 0.225)
Input Channels	3
Data Shuffling (Train)	Yes (random seed 42)
Data Shuffling (Test)	No
Number of Workers	4
Datasets Used	CAUCAFall, GMDCSA24, FallVision, Multiple Cameras
Evaluation Metrics	Accuracy, Precision, Recall, F1

eliminating the explicit temporal differencing and forcing the model to rely only on appearance features without any explicit motion encoding. M4 removes the Temporal Attention module, meaning all frames in the clip are treated with equal importance during temporal modeling regardless of their discriminative value for fall detection. M5 removes the Multi-Scale Pooling, replacing the combined max and average pooling strategy with a single pooling operation, which reduces the model’s ability to simultaneously capture both rapid motion events and long-duration temporal trends.

The ablation study results on the real PAF event camera dataset, summarised in Table. 7, confirm the importance of each architectural component in TemporalMotionNet. The full model (M1) achieves the best performance across all metrics with 81.42% accuracy and 73.91% F1 score. Removing the Appearance Stream (M2) results in a notable drop to 76.14% accuracy and 68.91% F1 score, demonstrating that spatial context from individual frames contributes meaningfully to cross-dataset generalisation. The most significant performance degradation is observed when the Motion Stream is removed (M3), where accuracy drops to 72.38% and F1 score falls to 64.98%, directly confirming that explicit motion encoding through temporal differencing is the single most critical component for effective sim-to-real transfer. This finding is consistent with the cross-dataset generalisation analysis in Section IV, where models with implicit temporal modeling showed substantially larger domain gaps. Removing the Temporal Attention module (M4) leads to a moderate drop to 78.91% accuracy and 71.87% F1 score, suggesting that while the BiLSTM can partially compensate for the missing attention mechanism, the ability to selectively focus on fall-critical frames still provides a meaningful performance benefit. Finally, removing Multi-Scale Pooling (M5) results in the smallest performance drop to 79.83% accuracy and 72.98% F1 score, indicating that while combining short-term and long-term temporal aggregation is beneficial, single-scale pooling retains most of the

temporal information captured by the BiLSTM. Overall, the ablation results validate that each component of TemporalMotionNet contributes positively to the final performance, with the Motion Stream being the most critical for robust cross-dataset generalisation.

4.3. Performance Evaluation and Generalisation Analysis

We conducted a comprehensive two-stage evaluation to assess both within-dataset performance and cross-dataset generalization capability.

Stage 1 (Within-Dataset) involved training and testing all nine models on combined RGB-to-Event converted datasets (GMDCSA-24, CAUCAFall, FallVision, and MC-Fall) with binary classification (Fall vs. Non-Fall).

Stage 2 (Cross-Dataset Generalization) tested the trained models on real neuromorphic event camera data i.e. PAF dataset, to evaluate synthetic-to-real transfer learning performance.

4.3.1. Stage 1: Within-Synthetic Datasets

Within the controlled environment of synthetic event data, TemporalMotionNet achieved the highest accuracy of **98.65%**, substantially outperforming all benchmark models. The performance ranking, as shown in Table 8, are as follows: (1) TemporalMotionNet (98.65%), (2) ConvLSTM (96.68%), (3) I3D (96.91%), (4) MobileNetV3-LSTM (97.64%), (5) EfficientViT (76.43%), (6) 3DCNN (95.66%), (7) TCN (89.07%), (8) TimeSFormer (75.29%), (9) Swin-Transformer (74.39%). The superior performance of TemporalMotionNet is attributed to its explicit motion encoding via temporal differencing ($\mathbf{d}_t = \mathbf{x}_{t+1} - \mathbf{x}_t$), which directly aligns with the structure of event-based data where temporal changes are the primary information source. The high recall of 99.40% indicates that TemporalMotionNet captures nearly all fall events within the synthetic dataset, while the precision of 98.23% demonstrates excellent specificity (few false positives). Notably, transformer-based models (Swin-Transformer: 74.39%, TimeSFormer: 75.29%) significantly underperform despite their state-of-the-art status on natural video benchmarks, suggesting that transformer architectures may not be optimally suited for event-based action recognition tasks.

The within-dataset results reveal important patterns. Recurrent and convolutional architectures (ConvLSTM: 96.68%, I3D: 96.91%, MobileNetV3-LSTM: 97.64%) achieve strong performance (96 – 98%), while transformer architectures show substantial degradation (74 – 76%), suggesting that the explicit temporal modeling through convolution and recurrence is more effective than self-attention mechanisms for event-based data. The TCN architecture (89.07%) shows intermediate performance, indicating that while dilated causal convolutions effectively model temporal dependencies, they lack the spatial-temporal integration of 3D convolutions. TemporalMotionNet’s performance advantage (98.65%) over MobileNetV3-LSTM (97.64%), despite comparable parameter counts (3.3M vs. 5.8M),

Table 6

Ablation Study: Effect of Removing Individual Components from TemporalMotionNet

Variant	Appearance Stream	Motion Stream	Temporal Attention	Multi-Scale Pooling
Full Model (M1)	✓	✓	✓	✓
W/O Appearance Stream (M2)	×	✓	✓	✓
W/O Motion Stream (M3)	✓	×	✓	✓
W/O Temporal Attention (M4)	✓	✓	×	✓
W/O Multi-Scale Pooling (M5)	✓	✓	✓	×

Table 7

Ablation Study Results: Cross-Dataset Generalisation on Real PAF Event Camera Dataset

Variant	Accuracy	Precision	Recall	F1 Score
M1	0.8142	0.7943	0.7402	0.7391
M2	0.7614	0.7421	0.6983	0.6891
M3	0.7238	0.7102	0.6541	0.6498
M4	0.7891	0.7712	0.7201	0.7187
M5	0.7983	0.7821	0.7312	0.7298

Table 8

Stage 1 Within-Dataset Performance: Models trained and tested on combined RGB-to-Event converted datasets using 80/20 train/test split.

Model	Accuracy	Precision	Recall	F1 Score
3DCNN	0.9566	0.9562	0.9682	0.9611
I3D	0.9691	0.9705	0.9759	0.9699
ConvLSTM	0.9668	0.9638	0.9779	0.9708
MVNetLSTM	<u>0.9764</u>	<u>0.9761</u>	<u>0.9807</u>	<u>0.9744</u>
TCN	0.8907	0.8922	0.8988	0.8961
SwinTransformer	0.7439	0.7422	0.7501	0.7499
TimeSFormer	0.7529	0.7606	0.7641	0.7306
EfficientViT	0.7643	0.7643	0.7533	0.7215
TempMotionNet	0.9865	0.9823	0.994	0.9881

Legend: **best results**, second best results

demonstrates that explicit motion encoding outweighs other architectural factors.

4.3.2. Stage 2: Cross Dataset Generalisation

When tested on real PAF event data, TemporalMotionNet maintained superior performance with **81.4%** accuracy, substantially outperforming the second-best model (MobileNetV3-LSTM: 75.2%) by a significant 6.2% margin. The performance ranking on real hardware differs markedly from within-dataset rankings, as shown in Table 9: TemporalMotionNet (81.4%) > MobileNetV3-LSTM (75.2%) > TCN (71.6%) > I3D (68.1%) > EfficientViT (63.5%) > 3DCNN (64.8%) > ConvLSTM (65.4%) > TimeSFormer (61.7%) > SwinTransformer (58.0%). This reordering is particularly striking for transformer architectures, which showed worst-in-class performance on real hardware despite moderate performance on synthetic data. The precision-recall trade-off for TemporalMotionNet on real hardware

Table 9

Stage 2 Cross-Dataset Generalization: Models trained on synthetic events, tested on real event camera dataset i.e. PAF

Model	Accuracy	Precision	Recall	F1 Score
3DCNN	0.648	0.6411	0.6561	0.6122
I3D	0.6814	0.6631	0.6828	0.6547
ConvLSTM	0.6539	0.6423	0.6522	0.6691
MVNetLSTM	<u>0.7521</u>	0.676	0.7004	0.6847
TCN	0.7164	<u>0.7022</u>	<u>0.7109</u>	<u>0.6917</u>
SwinTransformer	0.5802	0.5739	0.5822	0.5638
TimeSFormer	0.6168	0.6093	0.6022	0.5958
EfficientViT	0.6352	0.6283	0.6401	0.6269
TempMotionNet	0.8142	0.7943	0.7402	0.7391

Legend: **best results**, second best results

(79.4% precision, 74.0% recall) reveals a challenging safety trade-off: high precision (79.4%) minimizes false alarms, while the recall of 74.0% indicates that approximately 26% of actual falls are missed, below the industry standard of > 90% recall for safety-critical applications.

We identify a surprising and important pattern: **models that explicitly encode motion show dramatically better domain transfer than models that learn temporal patterns implicitly, as shown in Fig. 4.** Comparing models by their temporal modeling approach:

- **Explicit Motion Encoding (TemporalMotionNet):** Synthetic 98.65% → Real 81.4% (gap: 17.25%)
- **Implicit Temporal via Recurrence (ConvLSTM):** Synthetic 96.68% → Real 65.4% (gap: 31.28%)
- **Implicit Temporal via Convolution (3DCNN):** Synthetic 95.66% → Real 64.8% (gap: 30.86%)
- **Implicit Temporal via Attention (SwinTransformer):** Synthetic 74.39% → Real 58.0% (gap: 16.39%)

4.4. Experimental Validation

We deployed the TemporalMotionNet model on a Raspberry Pi 5 (8GB) interfaced with the GenX320 event camera to validate real-time edge performance, as shown in Fig. 5 and the summary of edge deployment is shown in Table 10. Across 212 consecutive inferences, the model demonstrated stable performance with a mean inference latency of 227.97 ± 30.46 ms and a coefficient of variation of

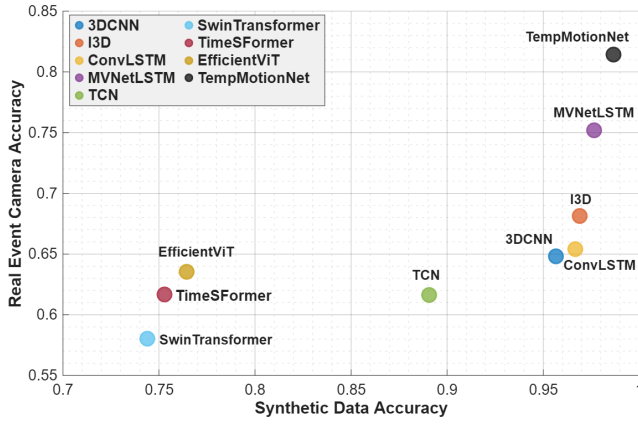


Figure 4: Domain Gap Analysis for Synthetic Data to Real Data Model Transfer

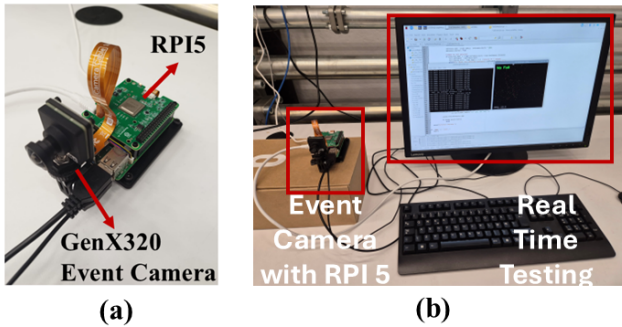


Figure 5: Description of Experimental Setup used for Real Time Validation (a) GenX320 Event Camera Attached with RPi5 (b) Complete Real Time Integration of System

13.4%, indicating consistent and predictable execution on resource-constrained hardware. Peak memory consumption remained at 16.11 MB, well within the RPi5's available resources, confirming the feasibility of edge deployment without requiring external computational offloading.

Real-time capability was rigorously validated against the 100 FPS temporal budget requirement. The input to the model is 20 consecutive frames with stride 1 and a clip budget of 200.0 ms per frame sequence and a mean processing time of 227.97 ms, the system achieved a latency headroom of 439.03 ms, corresponding to a 55.8% safety margin—well above the threshold required for reliable real-time operation. Fall and non-fall detection accuracy was evaluated across temporal window configurations ($N=1$, $N=20$, $N=40$), with optimal balanced performance achieved at $N=20$ (0.82 non-fall accuracy, 0.18 fall error rate), as shown in Fig. 6.

4.5. Discussion

The experimental validation reveals a critical tension between synthetic-to-real domain transfer and edge deployment feasibility for event camera-based fall detection. While TemporalMotionNet achieves exceptional within-dataset performance (98.65% on synthetic events), the cross-dataset evaluation on real PAF neuromorphic event camera data exposes a substantial 17.25% performance gap

(81.4% accuracy), highlighting the fundamental domain shift between RGB-to-event conversion and native event camera streams. This synthetic-to-real gap is primarily driven by three key factors: (1) the fixed contrast threshold ($C_p=C_n=0.35$) used in simulation does not capture the dynamic threshold variations of real sensors under changing lighting conditions, (2) real event cameras exhibit background activity noise and non-ideal sensor characteristics that are difficult to fully replicate in simulation, and (3) differences in camera viewpoints, sensor resolutions, and fall action definitions between the synthetic training datasets and the real PAF dataset further contribute to the domain shift. This synthetic-to-real gap, though significantly smaller than competing models (ConvLSTM: 31.28%, 3DCNN: 30.86%), suggests that explicit motion encoding through temporal differencing provides inherent robustness to event camera sensor characteristics and polarity-based encoding artifacts. However, the 81.4% accuracy on real event hardware raises a critical safety concern: with 74.0% recall, the system misses approximately 26% of actual fall events—substantially below the industry safety threshold of >90% recall for assistive robotics applications. This trade-off between precision (79.4%) and recall (74.0%) reflects the fundamental challenge in event camera-based fall detection, where false negatives pose direct safety risks and demand substantially higher recall even at the cost of increased false alarms. Promising future directions to address this gap include introducing domain randomization during simulation by randomly varying contrast thresholds and noise rates, fine-tuning the model on native event camera recordings, and exploring advanced domain adaptation strategies to bridge the sim-to-real divide.

The edge deployment on Raspberry Pi 5 with GenX320 event camera successfully validates real-time feasibility with a 55.8% latency margin (227.97 ms vs. 667 ms budget at 100 FPS), demonstrating that TemporalMotionNet's efficiency gains do not compromise computational tractability on resource-constrained platforms. The 200ms clip budget refers to the total accumulation time required to collect 20 consecutive event frames at a 10ms accumulation window per frame, while the 227.97ms covers the model inference time exclusively. The preprocessing overhead of aggregating raw discrete events (x, y, t, p) into frame representations is handled by the highly optimized Metavision SDK and remains negligible compared to the inference time, ensuring the overall end-to-end latency stays well within the real-time budget. Yet this technical success masks a deeper systemic challenge: achieving 90%+ recall on real event camera data while maintaining real-time edge constraints remains unsolved. The optimal temporal window configuration ($N=20$) in the deployment shows reasonable accuracy (82% non-fall), but this performance evaluation lacks the challenging domain transfer conditions of real neuromorphic event data.

Table 10
TemporalMotionNet Edge Deployment Summary

Category	Metric	Value
Model	Parameters	3.304M
	Size (FP32)	13.21
	Compression	4.0x
	Peak Memory	16.11 MB
Performance	Mean Inference	227.97 $\pm$ 30.46 ms
	Median Inference	217.68 ms
	Coefficient of Variation	13.4%
	Throughput	4.39 clips/s
Real-time	Clip Budget at 10ms accumulation time	200.0 ms
	Processing Time	227.97 ms
	Latency Headroom	439.03 ms
	Capability	Yes (55.8% margin)
Testing	Hardware	RPi5 (8GB)
	Inferences	212
	Success Rate	85%

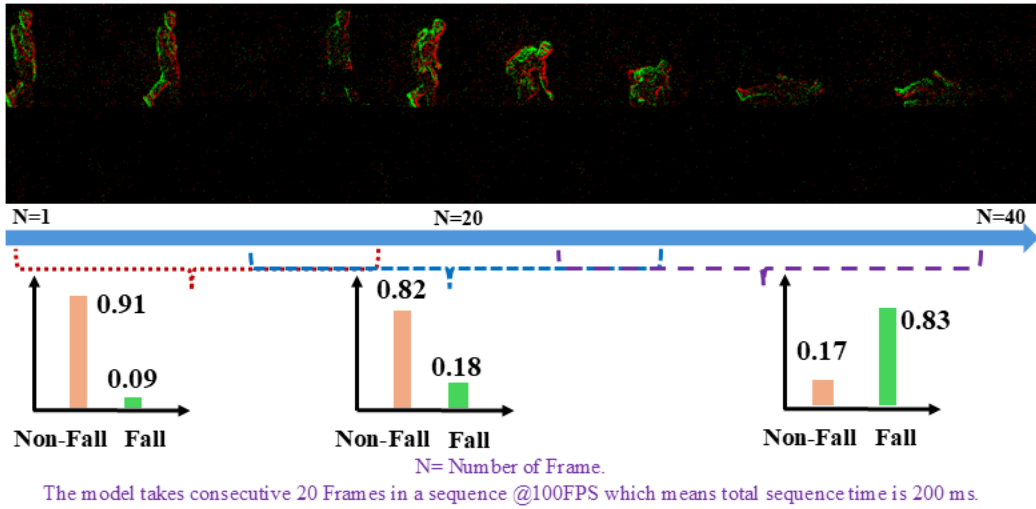


Figure 6: Real Time Fall and Non-Fall Detection Probabilities over frames

4.6. Future Work and Potential Research Directions:

While this work establishes a standardized benchmarking framework for event-based fall detection, several important directions remain open for future research. The most critical priority is improving recall on real event camera data, as the current 74.0% recall on the PAF dataset remains below the industry safety threshold of >90% required for assistive and healthcare applications. To address this, future work should explore domain adaptation techniques and fine-tuning on native event camera recordings to reduce the 17.25% sim-to-real performance gap. Introducing domain randomization during the simulation phase, such as randomly varying contrast thresholds and noise rates, is a promising direction to improve model robustness to real sensor characteristics without requiring additional data collection. Collecting larger and more diverse native event camera datasets across varied environments, viewpoints, and lighting conditions will also be essential for

training more generalizable models and enabling subject-independent evaluation protocols. In addition to that, exploring alternative event representations such as event count maps, time surfaces, and voxel grids as model inputs could reduce dependency on pseudo-color rendering and improve generalization to real hardware. Finally, the publicly released converted datasets provide a common benchmark for the community to investigate sensitivity analyses of simulation parameters, advanced fusion strategies with complementary sensing modalities, and more robust architectures tailored to the unique characteristics of neuromorphic event data.

5. Conclusion

This work establishes the first standardized cross-dataset and sim-to-real benchmarking framework for event-based fall detection. By converting four widely-used RGB fall detection datasets into event-based formats and evaluating on native event camera data (PAF dataset), we provide critical insights for advancing neuromorphic fall detection toward practical deployment. Our comprehensive evaluation

demonstrates that explicit motion encoding through temporal differencing is essential for effective domain transfer, with TemporalMotionNet achieving 81.4% accuracy on real event hardware while substantially outperforming comparison baselines that rely on implicit temporal modeling. However, it is important to acknowledge that while TemporalMotionNet achieves the best performance among all compared models, the 74.0% recall on real PAF event camera data remains insufficient for safety-critical fall detection applications, where the industry standard requires greater than 90% recall. This limitation highlights that the current work should be considered as a benchmarking study and a foundation for future research, rather than a deployment-ready solution.

The proposed TemporalMotionNet architecture successfully balances accuracy with computational efficiency, comprising fewer than 3.5M parameters while achieving competitive performance on both synthetic and real event data. Real-time edge deployment is validated through implementation on Raspberry Pi 5 with GenX320 event camera, achieving 227.97 ms mean inference latency with a 55.8% safety margin, confirming the technical viability of neuromorphic fall detection on resource-constrained devices. However, technical feasibility alone does not translate to clinical or assistive deployment readiness, as achieving reliable recall on real event camera data under diverse real-world conditions remains an open challenge. The standardized benchmarking framework and converted datasets released with this work provide a reproducible foundation for future research in this domain.

Acknowledgement

This research is supported by the Artificial Intelligence, Biomechatronics, and Collaborative Robotics research group at the Top Research Center Mechatronics (TRCM), University of Agder (UiA), Norway.

References

- Alam, E. (2024, July). *ekramalam/GMDCSA24-A-Dataset-for-Human-Fall-Detection-in-Videos: 2.0* (Version v2.0). Zenodo. <https://doi.org/10.5281/zenodo.12921216>
- Alam, E., Sufian, A., Dutta, P., & Leo, M. (2023). Real-time human fall detection using a lightweight pose estimation technique. *International Conference on Computational Intelligence in Communications and Business Analytics*, 30–40.
- Alam, E., Sufian, A., Dutta, P., Leo, M., & Hameed, I. (2024). Gmdcsa24: A dataset for human fall detection in videos. *Data in Brief*.
- Asif, U., Mashford, B., Von Cavallar, S., Yohanandan, S., Roy, S., Tang, J., & Harrer, S. (2020). Privacy preserving human fall detection using video data. *Machine Learning for Health Workshop*, 39–51.
- Auvinet, E., Multon, F., Saint-Arnaud, A., Rousseau, J., & Meunier, J. (2010). Fall detection with multiple cameras: An occlusion-resistant method based on 3-d silhouette vertical distribution. *IEEE transactions on information technology in biomedicine*, 15(2), 290–300.
- Auvinet, E., Rougier, C., Meunier, J., St-Arnaud, A., & Rousseau, J. (2010). Multiple cameras fall dataset. *DIRO-Université de Montréal, Tech. Rep.*, 1350, 24.
- Bai, S., Kolter, J. Z., & Koltun, V. (2018). An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*.
- Bertasius, G., Wang, H., & Torresani, L. (2021). Is space-time attention all you need for video understanding? *International Conference on Machine Learning*, 813–824.
- Cai, X., Zhou, Y., Han, H., Ruan, C., Li, Y., Wei, Z., & Wang, H. (2022). Efficientvit: Memory efficient vision transformers with hybrid structure. *arXiv preprint arXiv:2205.14756*.
- Caruana, R., Karpukhin, V., Yuille, A., & Zhu, F. (2017). Quo vadis, action recognition? a new model and large-scale datasets. 2017, 6299–6308.
- Eraso, J. C., Muñoz, E., Muñoz, M., & Pinto, J. (2022). Dataset caucafall. <https://doi.org/10.17632/7w7fccy7ky.4>
- Espinosa, R., Ponce, H., Gutiérrez, S., Martínez-Villaseñor, L., Brieva, J., & Moya-Albor, E. (2019). A vision-based approach for fall detection using multiple cameras and convolutional neural networks: A case study using the up-fall detection dataset. *Computers in biology and medicine*, 115, 103520.
- Feng, Q., Gao, C., Wang, L., Zhang, M., Du, L., & Qin, S. (2017). Fall detection based on motion history image and histogram of oriented gradient feature. 2017 *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, 341–346.
- Gallego, G., Delbrück, T., Orchard, G., Bartolozzi, C., Taba, B., Censi, A., Leutenegger, S., Davison, A. J., Conradt, J., Daniilidis, K., et al. (2020). Event-based vision: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(1), 154–180.
- Guerrero, J. C. E., España, E. M., Añasco, M. M., & Lopera, J. E. P. (2022). Dataset for human fall recognition in an uncontrolled environment. *Data in brief*, 45, 108610.
- Gutiérrez, J., Rodríguez, V., & Martín, S. (2021). Comprehensive review of vision-based fall detection systems. *Sensors*, 21(3), 947.
- Howard, A., Sandler, M., Chu, G., Chen, L.-C., Wang, B., Zhang, B., Oskotsky, H., & Krishnan, R. (2019). Searching for mobilenetv3. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 1314–1324.
- Hu, Y., Liu, S.-C., & Delbruck, T. (2021). V2e: From video frames to realistic dvs events. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1312–1321.
- James, S. L., Lucchesi, L. R., Bisignano, C., Castle, C. D., Dingels, Z. V., Fox, J. T., Hamilton, E. B., Henry, N. J., Krohn, K. J., Liu, Z., et al. (2020). The global burden of falls: Global, regional and national estimates of morbidity and mortality from the global burden of disease study 2017. *Injury prevention*, 26(Suppl 2), i3–i11.
- Ji, S., Xu, W., Yang, M., & Yu, K. (2012). 3d convolutional neural networks for human action recognition. *IEEE transactions on pattern analysis and machine intelligence*, 35(1), 221–231.
- Krishnan, K. S., & Krishnan, K. S. (2022). Benchmarking conventional vision models on neuromorphic fall detection and action recognition dataset. 2022 *IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC)*, 0518–0523.
- Lai, C.-F., Huang, Y.-M., Chao, H.-C., & Park, J. H. (2010). Adaptive body posture analysis using collaborative multi-sensors for elderly falling detection. *IEEE Intelligent Systems*, 24(6), 20–30.
- Liang, W., & ongsheng Ma, Y. (2024). Event-based real-time fall detection using yolov8 and lstm. 2024 *IEEE SENSORS*, 1–4.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., & Guo, B. (2021). Swin transformer: Hierarchical vision transformer using shifted windows. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 10012–10022.
- Miao, S., Chen, G., Ning, X., Zi, Y., Ren, K., Bing, Z., & Knoll, A. (2019). Neuromorphic vision datasets for pedestrian detection, action recognition, and fall detection. *Frontiers in neurorobotics*, 13, 38.
- Mirmahboub, B., Samavi, S., Karimi, N., & Shirani, S. (2012). Automatic monocular system for human fall detection based on variations

- in silhouette area. *IEEE transactions on biomedical engineering*, 60(2), 427–436.
- Perot, E., De Tournemire, P., Nitti, D., Masci, J., & Sironi, A. (2020). Learning to detect objects with a 1 megapixel event camera. *Advances in Neural Information Processing Systems*, 33, 16639–16652.
- Prasad, S. S., Mehta, N. K., Banerjee, A., Kumar, H., Saurav, S., & Singh, S. (2022). Real-time privacy-preserving fall detection using dynamic vision sensors. *2022 IEEE 19th India Council International Conference (INDICON)*, 1–6.
- Prasad, S. S., Mehta, N. K., Kumar, H., Banerjee, A., Saurav, S., & Singh, S. (2023). Hybrid snn-based privacy-preserving fall detection using neuromorphic sensors. *Proceedings of the Fourteenth Indian Conference on Computer Vision, Graphics and Image Processing*, 1–9.
- Rahman, N. N., Mahi, A. B. S., Mistry, D., Al Masud, S. M. R., Saha, A. K., Rahman, R., & Islam, M. R. (2025). Fallvision: A benchmark video dataset for fall detection. *Data in Brief*, 59, 111440.
- Rahman, N. N., Mahi, A. B. S., Mistry, D., Masud, S. M. R. A., Saha, A. K., Rahman, R., & Islam, M. R. (2024). *Fall Vision: A Benchmark Video Dataset for Advancing Fall Detection Technology*. <https://doi.org/10.7910/DVN/75QPKK>
- Rebecq, H., Gehrig, D., & Scaramuzza, D. (2018). Esim: An open event camera simulator. *Conference on robot learning*, 969–982.
- Rebecq, H., Ranftl, R., Koltun, V., & Scaramuzza, D. (2019). Events-to-video: Bringing modern computer vision to event cameras. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3857–3866.
- Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., & Woo, W.-c. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. *Advances in neural information processing systems*, 28, 802–810.
- Singh, A., Rehman, S. U., Yongchareon, S., & Chong, P. H. J. (2020). Sensor technologies for fall detection systems: A review. *IEEE Sensors Journal*, 20(13), 6889–6919.
- Wang, X., Risi, N., Talavera, E., Chicca, E., Karastoyanova, D., & Azopardi, G. (2023). Fall detection with event-based data: A case study. *International Conference on Computer Analysis of Images and Patterns*, 33–42.
- Xeferis, V.-R., Tsanousa, A., Meditskos, G., Vrochidis, S., & Kompatsiaris, I. (2021). Performance, challenges, and limitations in multi-modal fall detection systems: A review. *IEEE sensors Journal*, 21(17), 18398–18409.
- Zhang, Z., Cui, S., Chai, K., Yu, H., Dasgupta, S., Mahbub, U., & Rahman, T. (2024). V2ce: Video to continuous events simulator. *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 12455–12461.
- Zhu, A. Z., Yuan, L., Chaney, K., & Daniilidis, K. (2018). Ev-flownet: Self-supervised optical flow estimation for event-based cameras. *arXiv preprint arXiv:1802.06898*.