



OPEN SPARK: sparse-perception action recognition with keyframes for quadruped robots

Sehun Park & Andrew Jaeyong Choi✉

This study introduces a lightweight Human Action Recognition (HAR) model designed for computational efficiency and real-world applications. Faced with the challenge of processing large scale video data, the proposed approach strategically selects only the most informative keyframes, thereby, significantly reducing data redundancy. The model leverages a high-performing pre-trained DaViT backbone for feature extraction, combined with a Temporal Transformer that effectively captures both spatial details and temporal dynamics from sparse keyframes. The proposed method reduces the high computational cost associated with traditional architectures such as 3D CNNs, LSTMs that process every single frame. To validate its practical utility, the proposed model was deployed on a quadruped robot, establishing an efficient inference pipeline in which the robot captures video and performs on-device action recognition. The proposed method demonstrates a significant step towards applying complex HAR tasks in resource-constrained, robotic environments.

Keywords Human action recognition, Temporal transformer, Quadruped robot

The explosive growth of video data across countless domains has underscored the critical importance of Human Action Recognition (HAR). From intelligent surveillance systems¹ and advanced human-computer interaction² to content-based retrieval, the ability to automatically understand and classify human actions is a core technology. However, the inherent temporal dimension of video introduces a significant computational challenge³. Conventional deep learning models, such as 3D Convolutional Neural Networks (3D CNNs)⁴ or Long Short-Term Memory (LSTM)⁵ networks, often process every single frame of a video clip to capture spatiotemporal information. While highly effective, this exhaustive approach is resource-intensive, demanding substantial computational power and memory, which ultimately renders it impractical for many real-time applications on constrained devices like mobile robots.⁶

To address these limitations, this paper proposes a novel, lightweight HAR⁷ framework that balances recognition accuracy with computational efficiency. The overall architecture of our system, illustrating the distinct offline training and online inference pipelines, is shown in Fig. 1. Our research deviates from the frame-by-frame analysis paradigm. Instead, we introduce a keyframe selection mechanism that distills a video into a small, information-rich set, as conceptualized in Fig. 2. This strategy drastically reduces the amount of data to be processed without losing the essential context of the action.

The core of our model architecture synergizes two high-performing components: a pre-trained Vision Transformer (DaViT⁸) backbone and a Temporal Transformer. By using the DaViT model as a feature extractor, we harness its robust, pre-trained knowledge to derive detailed spatial features from each keyframe. These features are then fed into a Temporal Transformer, which is specifically designed to model the long-range dependencies and temporal relationships between the sparsely selected keyframes. This combination allows for a comprehensive understanding of an action's 'what' and 'when' using minimal data.

Finally, to demonstrate the true potential and practicality of our lightweight model, we have deployed it in a real-world scenario on a quadruped robot. We developed a complete inference pipeline where the robot's onboard camera captures live video and our model performs efficient action recognition. This successful implementation validates our model not only as an accurate and efficient architecture but also as a viable solution for embodied AI, establishing a foundation for increasingly advanced and autonomous robotic applications.

Robots that truly 'see' human gestures need vision models that run at the edge. Conventional 3D CNNs demand over 100 GFLOPs per video—draining batteries in minutes. We tackle this by reducing each approximately 10-second action (roughly 250 frames) to just eight essential keyframes, an approximately 97%

School of Computing, Gachon University, Seongnam 13120, Republic of Korea. ✉email: andrewjchoi@gachon.ac.kr

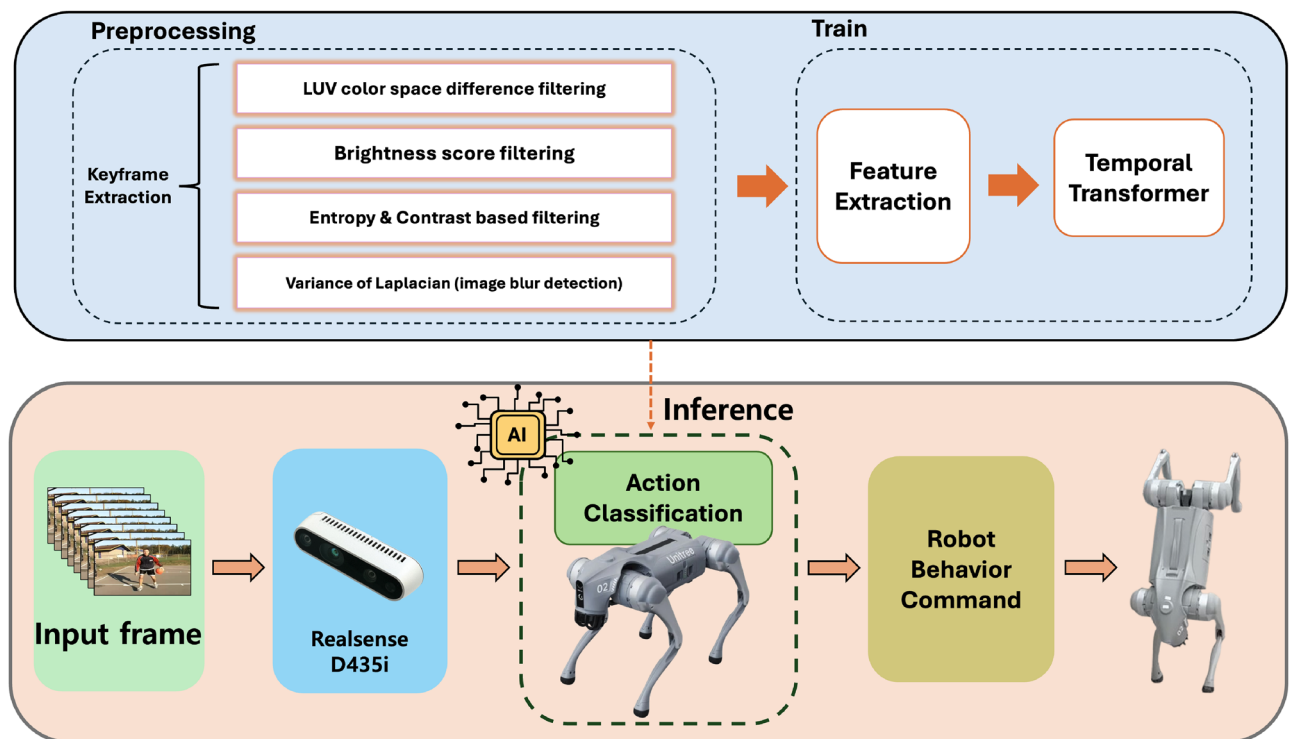


Fig. 1. System architecture showing the distinct offline training and online inference pipelines. The training pipeline (top) consists of a preprocessing stage for keyframe extraction and a training stage where features are extracted and fed into a Temporal Transformer to build the HAR model. The inference pipeline (bottom) is deployed on the robot, where it captures real-time frames, performs on-device action classification, and generates a behavior command.

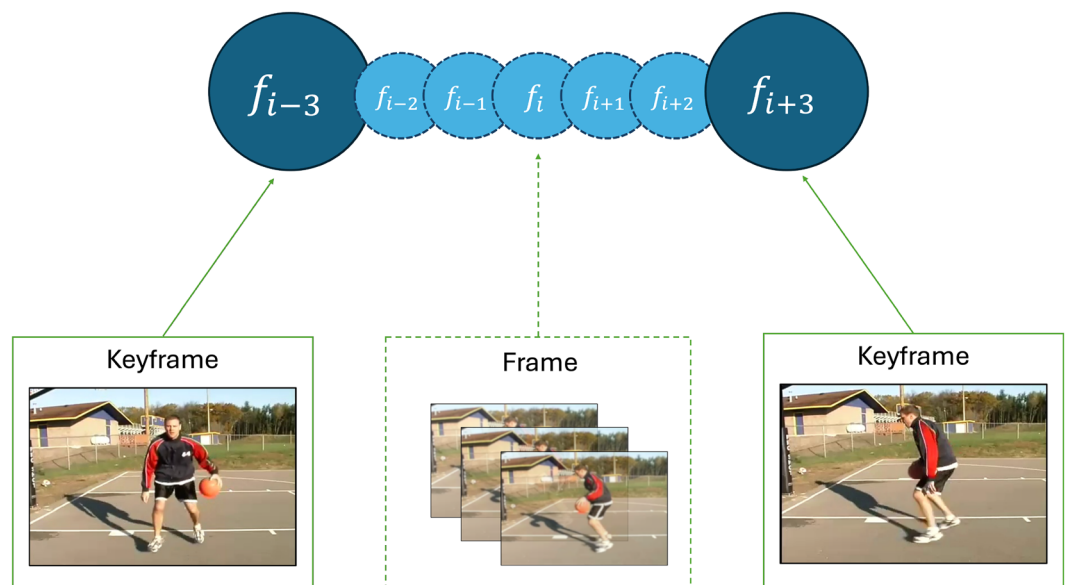


Fig. 2. Keyframe selection concept. The visual example shown is from the public Kinetics-400 dataset. Keyframes (left and right) are defined as frames containing the most semantically significant information within an action sequence, representing pivotal moments such as a change in posture, the peak of a movement, or a state transition. In contrast, the intermediate frames (center) often have high temporal redundancy. Our model selectively processes only these keyframes to ensure computational efficiency. The dataset is available at the following link https://huggingface.co/datasets/sehunpark/k400_keyframe.

reduction in data throughput. Our model then extracts spatial cues with a frozen DaViT and models time with a lightweight Temporal Transformer. The result: 68.5% Top-1 on Kinetics-400⁹ and an effective processed-frame throughput of 124.6 frames/s on an NVIDIA Orin NX Developer Kit (FP16, warm-start; 8 frames per inference step), enabling our Go2 Edu quadruped to react instantly to claps or finger-snaps.

Related work

The evolution of video-based action recognition

Early paradigms in Human Action Recognition (HAR) centered on the design of hand-crafted features. While foundational, these methods often exhibited limited generalization capabilities when faced with the variability of real-world video data. The emergence of deep learning marked a paradigm shift, leading to two-stream architectures that processed spatial information with 2D CNNs and temporal context with Recurrent Neural Networks (RNNs)¹⁰ or LSTMs. However, the disjointed nature of this approach often failed to capture complex spatiotemporal dependencies effectively. To address this, 3D CNNs, such as C3D and I3D¹¹, were introduced to learn spatiotemporal features jointly from video volumes. Despite achieving significant performance gains, these models are characterized by their immense computational and memory requirements, posing a substantial barrier to practical deployment. More recent advancements have seen the rise of video transformer architectures, like TimeSformer¹², which excel at modeling long-range dependencies. Nevertheless, the immense computational and memory requirements of both 3D CNNs and large-scale Transformers pose a substantial barrier to their deployment on resource-constrained platforms like mobile robots.

Approaches for efficient action recognition

To temper the high computational cost of deep learning-based HAR models, research focusing on “efficiency” has been actively pursued. The most representative strategy is the keyframe-based approach. This method dramatically reduces the computational load by processing only a few keyframes that capture the most important moments of an action, instead of every single frame.

Lightweight keyframe extraction often relies on appearance-based heuristics (e.g., blur, brightness, entropy, and contrast) to discard low-quality or redundant frames. While computationally attractive, such heuristics do not guarantee preservation of action-discriminative temporal cues, motivating empirical validation of the keyframe budget and stronger temporal aggregation for robust recognition.

Keyframe selection and representation granularity

Selecting a small set of representative frames can be viewed as compressing a video into a compact summary while retaining task-relevant information. Related directions in video understanding (e.g., video captioning) have explored how representation granularity and global–local cues affect the ability to capture salient temporal content. For example, GL-RG investigates global–local representation granularity for video captioning, and Prompt Learns Prompt explores knowledge-aware prompt collaboration to extract informative content from videos^{13,14}. Although these works target captioning rather than action recognition, they highlight a shared challenge: balancing compact representations with sufficient temporal semantics, which motivates careful design and validation of keyframe-based pipelines in our setting.

For instance, some methods based on channel averaging can lead to spatial information loss, while the use of RNNs for temporal processing risks information degradation. Crucially, these limitations make it difficult to build a system that is both robustly accurate and responsive enough for real-time interaction on a mobile robot, which requires a more intelligent selection of frames and a more powerful temporal model. This study aims to overcome this specific challenge.

Combining transfer learning and robotics vision

The success of modern computer vision relies heavily on the transfer learning paradigm, which utilizes models pre-trained on large-scale datasets for specific downstream tasks. Vision Transformer (ViT)¹⁵ and its variants have proven the high-performing generalization capabilities of such architectures. By using a pre-trained DaViT with its weights frozen purely as a feature extractor, our research efficiently obtains high-quality features without the cost of training a massive model from scratch. Furthermore, this research distinguishes itself by deploying the model on an actual robot, moving beyond theoretical performance evaluation. While most HAR research stops at measuring accuracy on standard datasets, this work builds a real-time inference pipeline by integrating the lightweight model onto a quadruped robot. This is a significant attempt to solve the practical challenges in the fields of Embodied AI or Robotics Vision. In summary, this work aims to bridge the gap between academic accuracy and real-world applicability by combining an efficient yet powerful pre-trained model, a holistic keyframe selection strategy, and a Temporal Transformer for temporal context.

Efficient and robust perception for mobile robots

Recent robotics and intelligent transportation literature highlights that real-world deployment requires not only accuracy but also robustness and efficiency under sensor and environmental constraints. For example, feature-level knowledge distillation has been explored to transfer representational capacity to compact models for place recognition¹⁶. Multi-modal perception and attention mechanisms have also been studied for navigation in intelligent vehicles, emphasizing robust decision-making from heterogeneous sensor inputs¹⁷. Comprehensive surveys further discuss open challenges in multi-modal place recognition, including domain shift, robustness, and compute constraints¹⁸. Robust visual place recognition under adverse illumination has been addressed via end-to-end enhancement networks¹⁹. Although these studies center on navigation and place recognition, they motivate a shared requirement in embodied systems: deployment-friendly perception under strict on-device

resource constraints. Our work is aligned with this objective by compressing video input into sparse keyframes and enabling low-latency inference on a quadruped platform.

Methodology

This study proposes a pipeline for human action recognition (HAR) that balances efficiency and performance. As illustrated in Fig. 3, the proposed methodology is composed of three sequential stages: (1) information-driven keyframe extraction to reduce computational load, (2) feature representation using a pre-trained and frozen vision backbone, and (3) temporal modeling and classification using a Temporal Transformer.

Keyframe extraction

Processing every frame in a video is inefficient, as many frames are redundant or irrelevant to the action itself. To address this, we select only the most information-rich keyframes using a multi-stage filtering process based on the Katna library²⁰ (LUV Color Space Filtering, Brightness Score Filtering, Entropy and Contrast Filtering, and Image Blur Detection). This drastic reduction in data is an essential step for enabling real-time processing on the robot's limited computing resources. To this end, we set the keyframe budget to $K=8$ as a practical compromise: it substantially reduces the frame budget while retaining sufficient information for accurate action classification. While many video recognition baselines sample 16–32 frames, our goal is single-view, keyframe-based inference under resource-limited deployment; accordingly we fix the budget to $K=8$.

Implementation note

Equations (1)–(5) summarize representative formulations of the heuristics used within Katna. In our implementation, these criteria are applied internally by Katna to generate and score candidate frames; we then enforce a fixed cardinality by selecting the top- K frames by Katna's score and reordering the selected keyframes chronologically (by frame index) before feeding them to the model. All Katna configuration parameters used in this work are reported in Table 1 for reproducibility.

1. **LUV Color Space Filtering** To measure visual changes more closely aligned with human perception, each frame is converted to the LUV color space. Frames where the absolute difference from the previous frame exceeds a set threshold are selected as candidates, reducing temporal redundancy. The difference is calculated as:

$$D(t) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W |LUV_t(i, j) - LUV_{t-1}(i, j)| \quad (1)$$

2. **Brightness Score Filtering** To ensure the quality of candidate frames, a brightness score is calculated for each frame. Frames that are too dark or overly bright, falling outside a predefined range $[B_{min}, B_{max}]$ are discarded to prevent distorted or uninformative features.

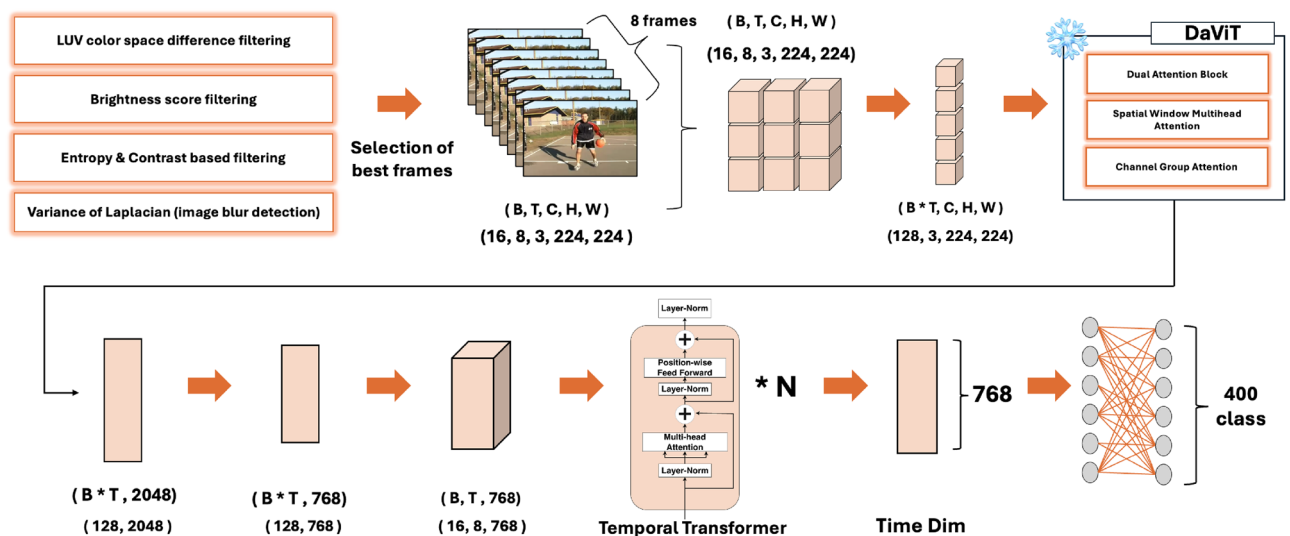


Fig. 3. The overall architecture of our efficient HAR pipeline. (1) First, a multi-stage filtering module selects a sparse set of 8 keyframes from the input video to reduce redundancy. (2) These frames are then passed through a frozen, pre-trained DaViT backbone and a linear projection layer to extract powerful 768-dimensional feature vectors. (3) The resulting sequence of feature vectors is fed into a Temporal Transformer to model spatiotemporal dependencies. (4) Finally, the sequence is summarized and classified into one of 400 action classes.

Component / Parameter	Value
Candidate proposal (Katna FrameExtractor)	
USE_LOCAL_MAXIMA	True
len_window	20
window_type	hanning
max_frames_in_chunk	500
Appearance-based selection (Katna ImageSelector)	
min_brightness_value	5.0
max_brightness_value	95.0
brightness_step	1.5
min_entropy_value	0.5
max_entropy_value	10.0
entropy_step	0.25
Output enforcement	
Final keyframe count K	8 (Top- K by Katna score)
Chronological reordering	Yes (by frame index)

Table 1. Katna configuration used for offline keyframe extraction. We use permissive thresholds to obtain a sufficiently large candidate pool, and then select exactly $K=8$ keyframes by ranking candidates with Katna’s score.

$$B(t) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W LUV_t(i, j)_L \tag{2}$$

3. *Entropy and Contrast Filtering* To quantify the level of detail within a frame, we measure its entropy and contrast. Frames with low entropy or weak contrast, indicating a lack of detailed information, are effectively removed.

$$E(t) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W E_L(i, j) \tag{3}$$

$$C(t) = \sqrt{\frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W (LUV_t(i, j)_L - B(t))^2} \tag{4}$$

4. *Image Blur Detection* Finally, to ensure keyframe quality, we evaluate the sharpness of each frame using Laplacian variance. Blurry frames with a variance below a certain threshold are removed from the final keyframe set, ensuring that only visually clear and informative frames are chosen.

$$\text{Var}(\Delta L_t) = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W (\Delta L_t(i, j) - \overline{\Delta L_t})^2 \tag{5}$$

This process reduces the frame budget from dense clips (typically hundreds of frames) to eight keyframes, substantially lowering input redundancy and I/O overhead. This massive data reduction is the primary enabler of the model’s efficient training and inference, despite its large parameter capacity.

Require: Video V , target keyframes $K=8$, Katna config Θ (Table 1)

Ensure: Chronologically ordered keyframes $\mathcal{K}$ (size K)

1: $(\mathcal{C}, s) \leftarrow \text{KATNAEXTRACT}(V; \Theta)$

2: $\mathcal{C}_K \leftarrow$ Top- K frames from $\mathcal{C}$ sorted by $s(f)$ (descending)

3: $\mathcal{K} \leftarrow$ sort $\mathcal{C}_K$ by frame index (ascending)

4: **return** $\mathcal{K}$

$\triangleright s(f)$: Katna internal score

Algorithm 1. Offline keyframe extraction (Katna + Top- K enforcement)

Algorithm 1 summarizes the exact offline keyframe extraction procedure used in this work.

Frame-wise feature representation

To convert the selected keyframes into rich and semantically meaningful feature vectors, our methodology centers on the strategic use of a pre-trained vision model. Our choice of backbone is DaViT (Dual Attention Vision Transformer), a powerful architecture that also serves as the foundation for Microsoft's Florence-2²¹ model. This selection is deliberate, as DaViT's hybrid design is uniquely suited for action recognition. It synergizes two distinct attention mechanisms: a CNN-like spatial window attention for capturing fine-grained local details, such as hand or object positions, and a Transformer-style channel group attention for modeling the global, long-range dependencies of the entire scene. This dual-attention approach allows DaViT to create a rich feature hierarchy that understands both the micro-level details of an action and its macro-level context, which is critical for distinguishing between similar actions.

A core component of our strategy is utilizing this DaViT backbone, pre-trained on ImageNet-1K²², in a frozen state where its weights are not updated during training. This decision is fundamental to achieving a lightweight and efficient model. Primarily, it drastically reduces the number of trainable parameters by over 90%, which in turn shortens training time, lowers memory requirements, and mitigates the risk of overfitting. Furthermore, this approach preserves the powerful, generalized visual knowledge from the massive ImageNet-1K dataset, providing a robust feature foundation without costly relearning. Consequently, the inference pipeline is significantly simplified and accelerated, satisfying a critical prerequisite for real-time operation on a mobile robot.

The resulting 768-dimensional feature vector, f'_t , from the frozen DaViT is then passed through a lightweight, trainable linear projection layer. This final layer acts as a task-specific adapter, mapping the general-purpose features into a new space that is optimized for our final action classification task.

Sequence modeling and classification

The final stage of our model is to interpret the sequence of feature vectors $F = (f_1, f_2, \dots, f_N)$ to classify the action. We employ a Temporal Transformer architecture for this purpose. The sequence is passed through $L=6$ standard Transformer encoder layers to effectively model the temporal order, patterns, and long-range dependencies between the keyframes. The output from the Transformer, a sequence of context-aware feature vectors, is then summarized into a single vector using Temporal Average Pooling. This summary vector, which represents the entire action, is fed into a final fully connected layer with a Softmax function to produce the probability distribution over the action classes.

By using a lightweight yet powerful Temporal Transformer, the model can effectively learn temporal context even from a limited number of keyframes, making this approach highly suitable for our efficiency-focused robotic application.

Asymmetric inference pipeline

It is crucial to distinguish our online inference strategy from the offline training phase. During offline training, we employ the information-driven keyframe extraction (LUV, Entropy, etc.) to force the model to learn from the most semantically rich features. However, for the online robotic inference, applying these heavy filters is impractical due to variable processing times and sensitivity to robot ego-motion.

Therefore, we adopt a lightweight uniform sampling strategy during inference. The robot maintains a sliding window buffer, selecting frames at a fixed stride (e.g., every 4th frame) to construct an 8-frame input tensor. This ensures:

1. Robustness to Ego-motion: By bypassing pixel-difference filters, the system remains stable even when the robot walks or turns.
2. Deterministic Latency: The processing pipeline guarantees a fixed execution time per inference step.
3. Consistency: A fixed number of 8 frames is always guaranteed, preventing tensor shape mismatch errors.

Experiment and results

To empirically validate our proposed model, we conducted a series of experiments, including a quantitative performance analysis and a real-world deployment demonstration.

Experimental setup

Dataset: The model was trained and evaluated on the Kinetics-400 dataset, a large-scale benchmark for action recognition comprising 400 human action classes. For each video clip, our keyframe extraction module was configured to select a sequence of 8 frames.

Implementation Details: The model was trained for a maximum of 50 epochs with a cosine annealing learning rate scheduler. We employed an early stopping strategy, monitoring the validation accuracy after each epoch and saving the checkpoint with the best performance. The best performing model was obtained at epoch 17, which was used for all subsequent evaluations.

Component ablation study

To address the constraints of robotic applications and justify our specific architectural choices, we conducted a component ablation study as summarized in Table 2. This experiment isolates the contributions of the sampling strategy, spatial backbone, and temporal aggregation module.

Config	Sampling Strategy	Backbone	Temporal Aggregation	Top-1 Acc (%)	Improvement (vs. Config 1, %p)
1	Random 8	DaViT (Frozen)	Avg Pooling	42.31	–
2	Keyframe	DaViT (Frozen)	Avg Pooling	56.20	+ 13.89
3	Keyframe	ResNet-50 (Frozen)	Avg Pooling	45.58	+ 3.27
4	Keyframe	ResNet-50 (Frozen)	Temporal Transformer	51.59	+ 9.28
5(ours)	Keyframe	DaViT (Frozen)	Temporal Transformer	68.50	+ 26.19

Table 2. Component ablation study on Kinetics-400. All backbones are pretrained on ImageNet and frozen during training.

Temporal Module	Params	Top-1 Acc(%)	Top-5 Acc(%)
Avg Pooling	95.7M	56.2	80.3
LSTM	97.5M	57.5	81.0
GRU	95.1M	60.4	83.3
Temporal Transformer (Ours)	102.2M	68.5	87.5

Table 3. Comparison of temporal aggregation modules on Kinetics-400 with sparse keyframes. Significant values are in bold.

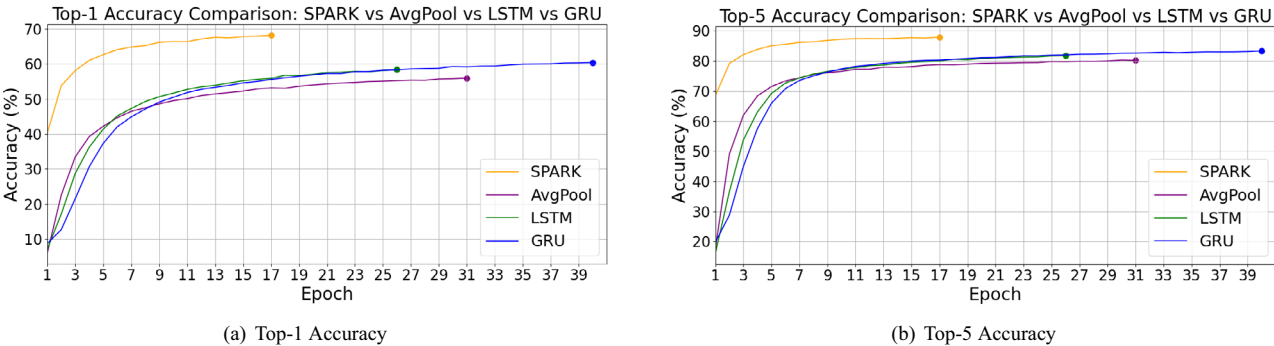


Fig. 4. Training curves for the baseline AvgPool model and the proposed SPARK model.

Necessity of Informative Keyframes (1 vs. 2): Replacing random sampling with keyframe selection improves Top-1 accuracy from 42.31% to 56.20% (+13.89%p), confirming that under a strict 8-frame budget, selecting semantically informative frames is crucial.

Importance of Strong Spatial Backbone (3 vs. 2): Under identical sampling (keyframes) and aggregation (average pooling), DaViT outperforms ResNet-50 by +10.62%p (56.20% vs. 45.58%), indicating that discriminative frame-level representations dominate performance in sparse-frame regimes.

Synergy with Temporal Transformer (3 vs. 4, 2 vs. 5(ours)): Temporal Transformer improves ResNet-50 from 45.58% to 51.59% (+6.01%p) and further boosts DaViT from 56.20% to 68.50% (+12.30%p). This suggests that temporal contextualization is beneficial, but its impact is substantially larger when built on high-quality spatial features.

Comparison of temporal modeling modules

Following the architectural validation, we further analyze temporal aggregation for sparse keyframes by comparing our Temporal Transformer with standard baselines, including Average Pooling (AvgPool), LSTM, and GRU. As summarized in Table 3, the Temporal Transformer achieves 68.5% Top-1 accuracy, improving over AvgPool (56.2%) and the recurrent baselines (LSTM: 57.54%, GRU: 60.39%). Top-5 accuracy also increases from 80.3% (AvgPool) to 87.5% (Transformer). These results suggest that when inputs are temporally sparse and potentially semantically discontinuous, recurrent models may be less effective at capturing long-range relationships, whereas global self-attention can more directly relate any pair of keyframes regardless of their temporal distance.

We additionally examine training dynamics in Fig. 4. The Temporal Transformer reaches its peak validation accuracy at epoch 17, earlier than AvgPool (epoch 31), LSTM (epoch 26), and GRU (epoch 40). Overall, these observations are consistent with the view that self-attention provides a useful inductive bias for sparse video representations, leading to both higher accuracy and earlier attainment of peak validation performance.

Model (Backbone)	Params (M)	Total GFLOPs (Per-view $\times$ Views)	Top-1 (%)	Training / Val Data	Inference (Frames/View $\times$ Views = Total)
TSN (R101)	43.3	48950 (195.8 $\times$ 250)	75.9	Full K400 Videos	8 frames $\times$ 250 views = 2000 frames
R(2+1)D (R34)	63.8	6390 (213.0 $\times$ 30)	75.5	Full K400 Videos	32 frames $\times$ 30 views = 960 frames
TSM (R50)	23.9	2632 (32.9 $\times$ 80)	73.2	Full K400 Videos	8 frames $\times$ 80 views = 640 frames
SlowFast 8 $\times$ 8 (R50)	34.6	1983 (66.1 $\times$ 30)	76.8	Full K400 Videos	32 frames $\times$ 30 views = 960 frames
I3D (NL, R50)	35.4	1779 (59.3 $\times$ 30)	74.8	Full K400 Videos	32 frames $\times$ 30 views = 960 frames
TPN (R50)	91.5	1515 (50.5 $\times$ 30)	74.2	Full K400 Videos	8 frames $\times$ 30 views = 240 frames
TANet (R50)	25.6	1032 (43.0 $\times$ 24)	76.3	Full K400 Videos	8 frames $\times$ 24 views = 192 frames
C2D (R50)	24.3	990 (33.0 $\times$ 30)	73.4	Full K400 Videos	8 frames $\times$ 30 views = 240 frames
TimeSformer (ViT-B)	122.0	588 (196.0 $\times$ 3)	77.7	Full K400 Videos	8 frames $\times$ 3 views = 24 frames
X3D-M (X3D)	3.8	186 (6.2 $\times$ 30)	76.4	Full K400 Videos	16 frames $\times$ 30 views = 480 frames
SPARK (DaViT-B)	102.0	122 (122.0 $\times$ 1)	68.5	Sparse K400 (8 Keyframes)	8 frames $\times$ 1 view = 8 frames

Table 4. Comprehensive Efficiency Comparison on Kinetics-400. We report Top-1/Top-5 accuracy on Kinetics-400 validation together with computational cost (GFLOPs $\times$ views) and the effective number of frame evaluations (frames per view $\times$ views). Standard baselines are trained/validated on dense full-video clips and typically use multi-view aggregation (spatial/temporal crops) at test time, which substantially increases total cost. In contrast, SPARK is trained and validated using only 8 sparse keyframes per video and is evaluated in a strictly single-view setting, requiring 8 input frames and 122 GFLOPs per inference step while achieving 68.5% Top-1 offline on Kinetics-400. This highlights SPARK's favorable test-time compute cost under severe temporal sparsity. Significant values are in bold.

Efficiency trade-off and data sparsity analysis

To contextualize SPARK's performance, we compare its accuracy and inference cost against 10 established baselines in Table 4. A key experimental difference is that all baselines are trained and validated on full Kinetics-400 videos, whereas SPARK is trained and offline-validated using only eight sparsely extracted keyframes per video. This extreme reduction in temporal information likely contributes to the observed accuracy gap while enabling substantially lower test-time cost.

Under the reported multi-view evaluation protocols, competitive accuracy is typically obtained by aggregating predictions over many views, which increases the effective inference workload. For example, C2D and TPN each evaluate 30 views, resulting in 240 frame evaluations (8 frames/view $\times$ 30) and total costs of 990 GFLOPs (33.0 $\times$ 30) and 1515 GFLOPs (50.5 $\times$ 30), respectively; TANet uses 24 views (192 frame evaluations; 1032 GFLOPs), and X3D-M, while efficient per view (6.2 GFLOPs), still relies on 30-view testing (480 frame evaluations; 186 GFLOPs). Heavier clip-based models are markedly more demanding (see Table 4).

In contrast, SPARK operates strictly in a single-view setting during both offline validation and on-device inference, processing exactly 8 frames per inference step with a total cost of 122 GFLOPs (122.0 $\times$ 1). For real-world deployment, we employ lightweight uniform sampling with a temporal stride of 4 (e.g., indices 0,4,8,12,16,20,24,28), which ensures deterministic latency while covering a temporal window sufficient for our gesture set. Overall, SPARK offers a practical accuracy–efficiency trade-off and defines a more realistic operating regime for on-device inference.

To provide a compact summary of accuracy per unit compute, Table 4 also reports *Top-1 per kGFLOPs*, defined as $\text{Top-1}(\%) \times 1000 / \text{Total GFLOPs}$. Figure 5 further visualizes the same accuracy–compute trade-off by plotting Top-1 accuracy against total inference GFLOPs for all methods.

Real-world deployment on quadruped robotics

To validate SPARK's practical utility, we deployed it on a Go2 Edu quadruped robot and implemented a complete on-device inference pipeline. The robot's onboard camera streams video frames, and the model performs action recognition locally on the embedded platform. We quantify robustness to ego-motion by measuring gesture recognition success rates under static- and moving-camera conditions on the robot (Table 6). In practice, the moving-camera setting is intentionally challenging and is not explicitly targeted by standard Kinetics-400 training; therefore, these results are best interpreted as an in-the-wild deployment stress test rather than a benchmark-equivalent evaluation.

To quantify on-device efficiency, we measure per-step latency on an NVIDIA Orin NX under FP32 and FP16 inference. We emulate real-time operation by constructing an 8-frame input clip via uniform sampling with a temporal stride of 4.

Under a cold-start condition, the average latency is 175.34 ms (FP32) and 110.57 ms (FP16). After a brief warm-up (10 consecutive inferences on the same 8-frame clip), latency stabilizes at 139.07 ms (FP32) and 64.18 ms (FP16), corresponding to 7.19 and 15.58 inference steps/s, respectively.

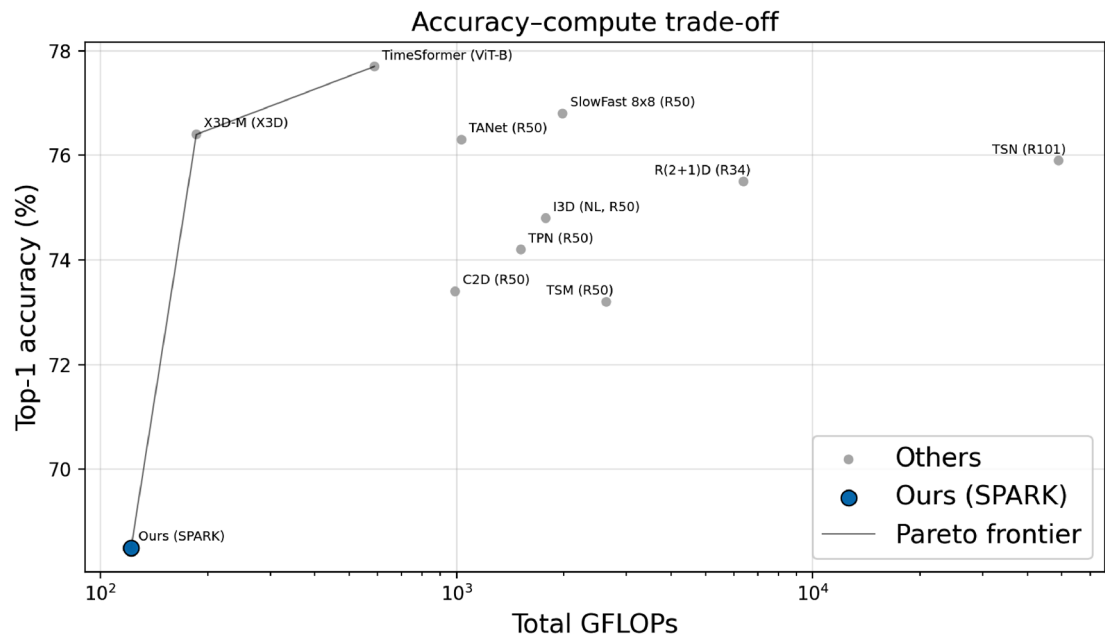


Fig. 5. Accuracy–compute trade-off on Kinetics-400. We plot Top-1 accuracy against total inference compute (Total GFLOPs, log scale) for SPARK and established baselines. The solid line denotes the Pareto frontier, i.e., models that are not dominated in both accuracy and compute. Points closer to the upper-left indicate a more favorable trade-off (higher accuracy at lower compute).

Model (Backbone)	Total GFLOPs	Top-1 (%)	Top-1 per kGFLOPs
TSN (R101)	48950	75.9	1.55
R(2+1)D (R34)	6390	75.5	11.82
TSM (R50)	2632	73.2	27.81
SlowFast 8 × 8 (R50)	1983	76.8	38.73
I3D (NL, R50)	1779	74.8	42.05
TPN (R50)	1515	74.2	48.98
TANet (R50)	1032	76.3	73.93
C2D (R50)	990	73.4	74.14
TimeSformer (ViT-B)	588	77.7	132.14
X3D-M (X3D)	186	76.4	410.75
SPARK (DaViT-B)	122	68.5	561.48

Table 5. Compute-normalized accuracy on Kinetics-400 under the unified evaluation protocol 4. We report *Top-1 per kGFLOPs*, defined as $\text{Top-1}(\%) \times 1000 / \text{Total GFLOPs}$, to summarize accuracy per unit of inference compute. Significant values are in bold.

Since each inference step processes 8 frames, the effective processed-frame throughput is 57.5 frames/s (FP32) and 124.6 frames/s (FP16).¹ We also report the peak memory footprint (RSS/VmRSS) as 1.041 GB (FP32) and 1.133 GB (FP16).

In a gesture-based control scenario, the robot approaches a person upon recognizing a “clapping” gesture and halts upon recognizing a “finger snapping” gesture. These demonstrations indicate that the proposed pipeline operates reliably under tight resource constraints and is responsive enough to support real-time command interfaces.

Conclusion

In this paper, we presented SPARK, a lightweight human action recognition framework designed for deployment on resource-constrained quadruped robots. SPARK compresses video input into a small set of informative

¹Processed-frame throughput is computed as (steps/s) × 8 frames per step and should not be confused with the camera frame rate.

Condition	#Trials	Clap SR	Snap SR	Overall SR
Static camera	25	19/25 (76%)	20/25 (80%)	39/50 (78%)
Moving camera (ego-motion)	25	14/25 (56%)	15/25 (60%)	29/50 (58%)

Table 6. Real-world deployment success rates on the Go2 Edu quadruped robot under camera ego-motion. Success rate (SR) is reported as correct recognitions over trials (25 trials per gesture). The moving-camera condition captures ego-motion during robot walking/turning while using the online uniform sampling inference pipeline (stride 4, 8 frames per step).

frames and combines a frozen DaViT backbone with a Temporal Transformer to model temporal context under a strict 8-frame budget.

On Kinetics-400, SPARK achieves 68.5% Top-1 accuracy while operating in an ultra-low compute regime (122 total GFLOPs in a strictly single-view setting). Compared with common multi-view evaluation protocols used by prior baselines, SPARK defines a more deployment-realistic operating point by processing exactly 8 frames per inference step, and the resulting accuracy–compute trade-off is summarized in Tables 4, 5, and Fig. 5.

We further validated real-world feasibility by deploying SPARK on a Go2 Edu robot equipped with an NVIDIA Orin NX. Under the online uniform-sampling inference pipeline (stride 4; 8 frames per step), SPARK achieves stable warm-start latencies of 139.07 ms (FP32) and 64.18 ms (FP16), corresponding to 7.19 and 15.58 inference steps/s, respectively (57.5 and 124.6 processed frames/s). In addition, we report gesture recognition success rates of 78% (static camera) and 58% (moving camera/ego-motion), providing an in-the-wild stress test of robustness during deployment (Table 6). Future work will focus on improving robustness under stronger ego-motion and challenging imaging conditions, for example through motion-aware sampling, lightweight stabilization, or in-domain fine-tuning for gesture-centric interactions, while preserving deterministic latency and on-device efficiency.

Data availability

The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request. The dataset is available at the following link https://huggingface.co/datasets/sehupark/k400_keyframe.

Received: 5 August 2025; Accepted: 20 April 2026

Published online: 08 May 2026

References

- Xie, B. J., Li, H., Luan, Z., Li, X. X. & Lei, Z. A lightweight coal mine pedestrian detector for video surveillance systems with multi-level feature fusion and channel pruning. *Sci. Rep.* **15**, 5757. <https://doi.org/10.1038/s41598-025-87157-7> (2025).
- Liu, Q., Ji, Z., Xu, W., Liu, Z. & Wang, L. One-shot learning-driven autonomous robotic assembly via human-robot symbiotic interaction. *npj Adv. Manuf.* **2**, 22. <https://doi.org/10.1038/s44334-025-00030-3> (2025).
- Aly, M. & Fathi, I. S. Recognizing american sign language gestures efficiently and accurately using a hybrid transformer model. *Sci. Rep.* **15**, 20253. <https://doi.org/10.1038/s41598-025-06344-8> (2025).
- Tran, D., Bourdev, L., Fergus, R., Torresani, L. & Paluri, M. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, 4489–4497. <https://doi.org/10.1109/ICCV.2015.510> (2015).
- Hochreiter, S. & Schmidhuber, J. Long short-term memory. *Neural Comput.* **9**, 1735–1780. <https://doi.org/10.1162/neco.1997.9.8.1735> (1997).
- Chen, L., Li, C., Fahmy, A. & Sienn, J. Gesturero: an algorithm for autonomous mobile robot teleoperation based on gesture recognition. *Sci. Rep.* **14**, 6199. <https://doi.org/10.1038/s41598-024-54488-w> (2024).
- Huan, S. et al. A lightweight hybrid vision transformer network for radar-based human activity recognition. *Sci. Rep.* **13**, 17996. <https://doi.org/10.1038/s41598-023-45149-5> (2023).
- Ding, M. et al. Davit: Dual attention vision transformers. In *Computer Vision—ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXIV*, 74–92 (Springer, 2022).
- Kay, W. et al. The kinetics human action video dataset (2017). [arXiv:1705.06950](https://arxiv.org/abs/1705.06950).
- Sherstinsky, A. Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network. *Physica D: Nonlinear Phenomena* **404**, 132306. <https://doi.org/10.1016/j.physd.2019.132306> (2020).
- Carreira, J. & Zisserman, A. Quo vadis, action recognition? A new model and the kinetics dataset (2018). [arXiv:1705.07750](https://arxiv.org/abs/1705.07750).
- Bertasius, G., Wang, H. & Torresani, L. Is space-time attention all you need for video understanding? In *Proceedings of the International Conference on Machine Learning (ICML)* (2021).
- Yan, L. et al. Gl-rg: Global-local representation granularity for video captioning. In Raedt, L. D. (ed.) *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2769–2775. <https://doi.org/10.24963/ijcai.2022/384> (International Joint Conferences on Artificial Intelligence Organization, 2022). Main Track.
- Yan, L., Han, C., Xu, Z., Liu, D. & Wang, Q. Prompt learns prompt: Exploring knowledge-aware generative prompt collaboration for video captioning. In Elkind, E. (ed.) *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, (International Joint Conferences on Artificial Intelligence Organization, 2023). Main Track. 1622–1630. <https://doi.org/10.24963/ijcai.2023/180>
- Dosovitskiy, A. et al. An image is worth 16x16 words: Transformers for image recognition at scale (2021). [arXiv:2010.11929](https://arxiv.org/abs/2010.11929).
- Li, Z., Xu, P., Dong, Z., Zhang, R. & Deng, Z. Feature-level knowledge distillation for place recognition based on soft-hard labels teaching paradigm. *IEEE Trans. Intell. Transp. Syst.* **26**, 2091–2101. <https://doi.org/10.1109/TITS.2024.3495656> (2025).
- Li, Z., Shang, T. & Xu, P. Multi-modal attention perception for intelligent vehicle navigation using deep reinforcement learning. *IEEE Trans. Intell. Transp. Syst.* **26**, 8657–8669. <https://doi.org/10.1109/TITS.2025.3535885> (2025).
- Li, Z., Shang, T., Xu, P. & Deng, Z. Place recognition meet multiple modalities: A comprehensive review, current challenges and future directions (2025). [arXiv:2505.14068](https://arxiv.org/abs/2505.14068).

19. Li, Z., Shang, T., Xu, P., Deng, Z. & Zhang, R. Toward robust visual place recognition for mobile robots with an end-to-end dark-enhanced net. *IEEE Trans. Ind. Inform.* **21**, 1359–1368. <https://doi.org/10.1109/TII.2024.3476574> (2025).
20. Kepler Lab. Katna: An automatic video keyframe extraction and summarization library. <https://github.com/keplerlab/katna> (2022). Accessed 2025 July 31.
21. Xiao, B. et al. Florence-2: Advancing a unified representation for a variety of vision tasks (2023). [arXiv:2311.06242](https://arxiv.org/abs/2311.06242).
22. Deng, J. et al. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. <https://doi.org/10.1109/CVPR.2009.5206848> (2009).

Acknowledgements

This work was supported by the Gachon University research fund of 2024, Republic of Korea (GCU-202500670001)

Author contributions

S.P. Conceptualization, Methodology, Software, Formal analysis, Data curation, Writing – original draft. A.J.C. Investigation, Methodology, Formal analysis, Supervision, Project administration, Funding acquisition. All the authors reviewed the manuscript.

Funding

This work was supported by the Gachon University research fund of 2024, Republic of Korea (GCU-202500670001).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to A.J.C.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2026