mathematics

MDPI

Article

Bayesian Prototypical Pruning for Transformers in Human–Robot Collaboration [†]

Bohua Peng [1] **and Bin Chen** [2,*]

[1] School of Automation, Northwestern Polytechnical University, Xi'an 710072, China; bohua_peng@mail.nwpu.edu.cn

[2] School of Electrical and Electronic Engineering, The University of Sheffield, Sheffield S1 4DP, UK

[*] Correspondence: bin.chen@sheffield.ac.uk

[†] This paper is an extended version of our paper published in 2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA), Padova, Italy, 10–13 September 2024; pp. 1–7.

Abstract: Action representations are essential for developing mutual cognition toward efficient human–AI collaboration, particularly in human–robot collaborative (HRC) workspaces. As such, it has become an emerging research direction for robots to understand human intentions with video Transformers. Despite their remarkable success in capturing long-range dependencies, local redundancy in video frames can add up to the inference latency of Transformers due to overparameterization. Recently, token pruning has become a computationally efficient solution that selectively removes input tokens with minimal impact on task performance. However, existing sparse coding methods often have an exhaustive threshold searching process, leading to intensive hyperparameter search. In this paper, Bayesian Prototypical Pruning (ProtoPrune), a novel end-to-end Bayesian framework, is proposed for token pruning in video understanding. To improve robustness, ProtoPrune leverages prototypical contrastive learning for fine-grained action representations, bringing sub-action level supervision to the video token pruning task. With variational dropout, our method bypasses the exhaustive threshold searching process. Experiments show that the proposed method can achieve a pruning rate of 37.2% while retaining 92.9% of task performance using Uniformer and ActionCLIP, which significantly improves computational efficiency. Convergence analysis ensures the stability of our method. The proposed efficient video understanding method offers a theoretically grounded and hardware-friendly solution for deploying video Transformers in real-world HRC environments.

Keywords: spatial–temporal modeling; sparse coding; human–robot collaboration; action recognition; inference optimization

MSC: 62F15

check for updates

Academic Editor: Liangxiao Jiang

Received: 21 March 2025

Revised: 19 April 2025

Accepted: 23 April 2025

Published: 25 April 2025

Citation: Peng, B.; Chen, B. Bayesian Prototypical Pruning for Transformers in Human–Robot Collaboration. *Mathematics* **2025**, *13*, 1411. https://doi.org/10.3390/math13091411

1. Introduction

The semantic representation of human actions is essential for agile planning in human–AI collaboration [1]. A significant research question for human–robot collaborative (HRC) systems [2] is to explore human cognition [3] and behaviors in manufacturing more profoundly and develop robust action recognition methods to understand human–robot collaboration. Recently, Transformers [4] have been emerging as promising models for action representation learning because of their ability to capture long time range dependency in videos [5]. However, semi-automatic assembly lines often have humans performing similar yet distinct sub-actions [6], as tasks are often sequential, hierarchical, and context dependent [7]. These sub-actions are parts of overarching actions, and are often unlabeled due to labeling cost [8], hence urging data-driven modeling for disambiguation. For example, a shared control action may include command, control, and manipulation, where a human first instructs through GUIs, then adjusts the movement via joysticks and finally interacts directly (see Figure 1). One important research question is to automatically learn useful action representations for action recognition. Supervised representation learning [9] has achieved state-of-the-art performance by learning generalizable features with robust fault tolerance via multi-view contrastive learning [10]. Recently, contrastive multi-view coding (CMC) [11] shows a promising direction to representation learning across multi-views [12]. Despite its superior performance, the method demands numerous contrastive action pairs for pulling similar samples together and pushing dissimilar actions apart. Acknowledging multi-facets of actions in collaborative manufacturing may produce a more comprehensive understanding of human actions, whose usability is not well investigated in weakly supervised learning [13] where actions are more fine-grained and unlabeled.

Another problem is the applicability of modern neural networks, stemming from massive overparameterization, which requires graphic processing units for real-time inference. The computational complexity of Transformer models grows quadratically with the length of an action sequence, leading to high CPU latency and time delay, which limits the model's ubiquity in edge devices. For instance, ViT-Base [14] can take around 10 s per image for a 224 × 224 input image on a high-end CPU, while convolutional neural networks only take 10 ms on the same device [15]. Real-time inference is critical in an AI assembly line [10] because delays in action recognition can lead to communication mismatches or even safety risks. To address this, one feasible solution is to enforce the sparsity of the deep features to reduce the insignificant dimensions for computational efficiency. Token pruning [16] is a dimensionality reduction method in which input tokens with limited effects on predictions are pruned. In computer vision, token pruning can actually improve the timing of Transformers on edge devices as unnecessary input features are removed from the computational graph. However, naively applying token pruning to action recognition poses difficulties in two aspects. Primarily, existing pruning methods [17,18] that employ the Bayesian setup depend on fixed layer-wise thresholds to decide whether to retain or discard a token, founded by either tuning expertise or grid search. Furthermore, action class distributions can significantly vary between synthetic and real-world environments, imposing severe viewpoint variance during sim-to-real knowledge transfer [19]. Specifically, the features of human–AI collaborative workspaces can be influenced by illumination and occlusion [20]. These noisy environmental factors can challenge the robustness or even convergence of an action recognition algorithm, which was underexplored in previous work.

In this work, we propose a Prototypical Pruning method, namely (ProtoPrune), to select task-relevant features and extract sub-action concepts with end-to-end learning. With a weakly labeled action sequence, ProtoPrune initiates the sub-action optimization with K-means clustering. Then, the method iteratively optimizes the sub-action representation and feature selection with prototypical contrastive learning regularized by an L-1 term. To avoid exhaustively searching the shrinkage thresholds, we apply variational dropout [21,22] to a pretrained self-attention mechanism using Gumbel-Softmax tricks [23]. In particular, we integrate task awareness and between-frame similarity into the selective attention mechanism, thus enforcing top-to-bottom supervision. Finally, we show that ProtoPrune is a theoretically grounded approach for video feature selection yielding a high compression rate and accuracy. The proposed method can be used in practical applications such as safety monitoring [24] and adaptive workspace planning [25] in human–robot collaborative environments.

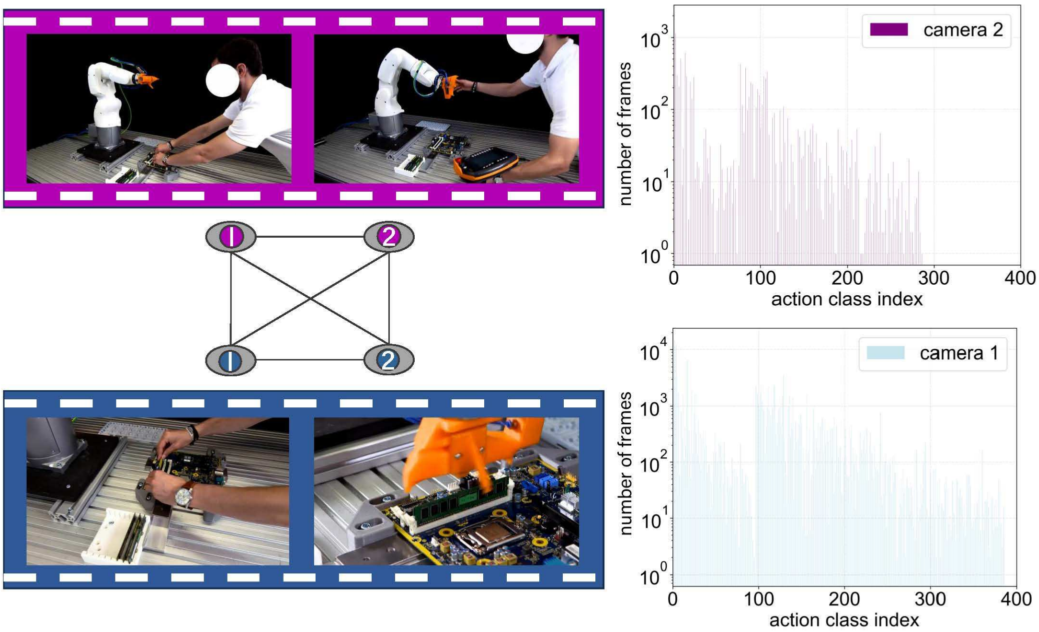


Figure 1. Spatio-temporal graphs from dual-view cameras in human–robot collaboration.

To summarize, the contributions of this paper are as follows:

- ProtoPrune: a Bayesian token pruning method that automatically selects task-relevant features with a refined self-attention mechanism.
- Theoretical analysis for the convergence of the Prototypical Pruning method with mathematical proof.
- Experimental analysis on two off-the-shelf video Transformers, demonstrating that task-awareness supervision can efficiently guide token pruning.

The remainder of this work is organized as follows: Section 2 contextualizes ProtoPrune with recently proposed methods. Section 3 introduces prototypes to remove task-irrelevant inputs with a deep Bayesian learning framework. Section 4 gives a theoretical analysis of the convergence of the proposed method. The experiments in Section 5 demonstrate that prototype-aware token pruning is the major component of inference speedup, while Section 6 concludes the paper.

2. Related Works

This section will discuss related works in human–robot collaboration, keyframe extraction, automatic action recognition, optical flow and action recognition, and structured pruning. These topics are essential for understanding the context of this paper.

2.1. Human–Robot Collaboration

Human–robot collaboration (HRC) in a manufacturing context allows humans to work with robots in close proximity [26]. In the last decade, numerous studies have explored HRC applications in manufacturing [27], including assembly [28], material handling, welding, picking-and-placing, and more, promoting HRC applications for human safety [29], operator assistance, and robust adaptive control [30]. The human–robot co-working of Industry 5.0 [31] needs to learn generalizable knowledge and anticipate human actions to enable humans and robots to execute ergonomic operations [32]. With evolving task arrangements, cobot video processing should adapt to egocentric video [33] to proactively coordinate with humans. In this work, we leverage prototypes to model weakly labeled sub-actions, providing a fine-grained pruning solution for inference acceleration.

2.2. Keyframe Extraction

An unsupervised clustering [34] proposed the adaptive method for feature selection in video signals, choosing those nearest to the cluster centers as keyframes. VSUMM [35] selects typical features using k-means clustering based on color features. Delaunay clustering [36] groups frames into clusters based on their geometric proximity in a high dimensional space. Work in [37] incorporates caption information into frame representation. Graph regularized matrix factorization [38] has integrated structure information into keyframe extraction. Meanwhile, Transformer models [4] have gained considerable attention across various domains. While Transformer models offer highly expressive representations, their computational cost scales quadratically with the input sequence length, making them prohibitively expensive for many applications. This limitation is significant in energy-sensitive environments, such as human–robot collaboration (HRC) systems where computational efficiency is a critical concern [39].

2.3. Automatic Action Recognition

Video sequence has local redundancy in both space and time [40]. To learn representations for action recognition, pioneering works, such as 3D ConvNet [41] and I3D [42], leverage 3D convolutions to extract features from video frames. However, the fixed receptive fields of convolution operations hinder model expressiveness for global dependency [43]. While Transformers can compute global correspondence, they are prone to learn shortcuts such as scenes or viewpoints. Ref. [44] mitigates the scene bias of extracted features by maximizing an adversarial loss to scene labels. The skeleton-based method [45] can address bias by removing scene-related background but requires extra 3D skeleton extraction. The methods mentioned above mainly consider single-view videos instead of multi-view videos. Multi-view systems [46] offer practical solutions to address occlusion challenges by leveraging diverse perspectives to ensure fault tolerance in dynamic environments. This approach is often improved through synthetic or real-world data augmentation [47]. In this paper, we introduce an orthogonal method that integrates fine-grained action representations into token pruning, further improving the responsiveness and adaptability of HRC systems.

2.4. Optical Flow and Action Recognition

Optical flow [48] describes motion vectors for each pixel within a video frame. Optical flow contributes to dynamic motion pattern recognition with its invariance to appearance across different scenes. Ref. [49] introduces a dual-stream architecture, which separately processes RGB images and optical flow through spatial and temporal convolutional networks, respectively. TSN [50] extends this framework by extracting multiple short video snippets and training the networks with model consensus as the labels. However, optical flow is optimized with end-point error [51], which correlates poorly with action recognition. Unlike these methods, ProtoPrune implicitly encodes optical flow information using self-attention mechanisms and further measures redundancy between sampled frames by computing similarity scores with action prototypes.

2.5. Structured Pruning

Compressive sensing [52] has laid the groundwork for expressing signals with sparse representations [53]. Pruning is the practice of removing redundant components from a model. Pruning can be broadly categorized as structured pruning [54] and unstructured pruning [55]. Structured pruning is a model compression technique that removes entire groups of neural network parameters (e.g., neurons, filters, channels, or layers) based on their importance scores while preserving the overall architecture of the model. Unlike unstructured pruning, which removes individual weights independently and often results in irregular sparsity, structured pruning maintains the dense matrix structure, making it more hardware-friendly and computationally efficient [15]. Token pruning [56], with its origins in NLP, can effectively improve throughput by removing inputs irrelevant to the task.

In computer vision, EViT [57] leverages self-attention fusion to merge unimportant tokens into a meta-token, which can be a bottleneck to model expressiveness. K-centered patch sampling [58] leverages K-center Search for structured patch sampling. Nevertheless, its performance is sensitive to a fixed hyperparameter—the number of cluster centers. ToMe [59] adopts a soft bipartite matching algorithm to pair and merge tokens adaptively, but this proportional attention relies on sorting to find paired tokens and a fixed threshold to keep top-r paired tokens, inevitably increasing latency in real-time inference. To address this limitation, ToFu [60] dynamically combines token pruning and token merging to select the most suitable strategy based on "functional linearity". Adjust [61] leverages the Gradient Aware Scaling (GAS) operation to adjust the pruning rate, but its top-k pruning process relies on a fixed hyperparameter to select the more important tokens, which might lead to inaccurate gradient estimation and affect algorithm convergence.

Similar to EViT, ProtoPrune is also an attention-based method that dynamically selects important video tokens that jointly consider three factors—attention activation, prototypical similarity, and frame redundancy—enabling its sparsity to adapt to input content. This Bayesian pruning method offers greater flexibility by combining the advantages of both token merging and token matching, without using fixed hyperparameters. In this way, the proposed method learns a dynamic token selection strategy rather than relying on a predefined token selection function. Additionally, the prototypical learning algorithm inherently facilitates data mining, allowing it to discover sub-actions in action recognition tasks. Compared to K-centered approaches, the proposed method leverages iterative clustering only during training to learn prototypes, while using static prototypes during inference, thereby adding negligible computational complexity. The key elements of these methods are summarized in Table 1.

Table 1. Key elements of the ProtoPrune and related works.

Study	**Summarization**	**Key Element**
EVit [57]	Token merging via fusion	Self-attention Fusion
K-centered [58]	Efficient patch-based sampling	K-center Search
ToMe [59]	Image token merging	Bipartite Matching
ToFu [60]	Combine merging and pruning	Bipartite Soft Matching
Adjust [61]	Adaptive pruning rate	Gradient Aware Scaling
ProtoPrune (ours)	Pruning weakly labeled sub-actions	Prototypical learning

3. Methodology

In this section, we will first introduce the mathematical notations of action recognition and formulate it as a sparse coding problem. We consider video sequence classification as a sequence classification problem because a video is inherently a sequence of frames that unfolds over time. Unlike static image classification, where a single image is analyzed, video classification requires models to understand how the frames evolve within the time; additionally, small chunks in the same sequence may provide fine-grained hierarchical structures into the overall activity. Then, we will introduce a Bayesian action recognition method by learning prototypical representations for multi-views. After that, we will derive a Bayesian token pruning method to sparsify action representations without the need for exhaustively searching pruning thresholds. Figure 2 shows an overview of the Prototypical Token Pruning method (ProtoPrune). The Transformer model encodes prototypes (μ) and samples (x) into a deep latent space, where distances to subclass prototypes corresponds to categorical probabilities. The sparse attention mechanism reuses attention weights as importance scores, removing tokens with less impact with Gumbel-Softmax to speed up inference. For the convenience of method introductions, we provide Table 2 to show the math symbols used in this paper.

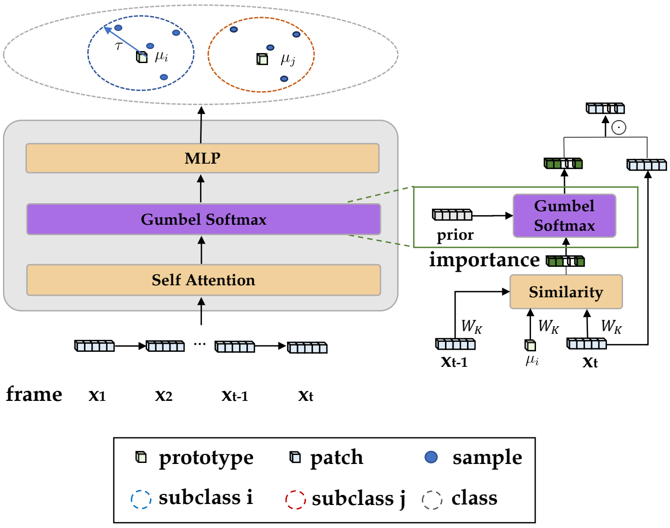


Figure 2. The computational graph of Prototypical Pruning. Gumbel-Softmax learns a binary mask based on the semantic similarity with the prototype and redundancy between adjacent frames (marked by red). Prototypes μ_i and μ_j are fine-grained sub-actions of the same class. Video tokens are iteratively sparsified by reusing learned self-attention.

Table 2. Mathematical notations of the methodology

Symbol	Description
$\mathbf{V}$	A video clip recording one action
e_i	The embedding of the i-th frame, with $x_i \in \mathbb{R}^d$
d	The number of hidden dimensions
x	Feature summed over the hidden dimension
μ	The keyframe (prototype) in multi-views
T	Hard threshold for token pruning operations
ρ	Temperature for Gumbel-Softmax operations
α	The margin to discriminate the representations
τ	Concentration level of the feature distribution in prototypes
f	The raw likelihood function used in the Bayesian inference
f_{proto}	The modified likelihood function in Bayesian Prototypical Pruning
ϵ	Gumbel sample from a prior distribution
w	Aggregating feature magnitudes on the l th layer
CLS	The [CLS] token from a video Transformer
PRR	Performance retention ratio measures accuracy of pruned models
GFLOPs	Giga floating point operations measure inference computation

3.1. Problem Formulation

Given a video clip $\mathbf{V} \in \mathbb{R}^{M\times H\times W\times 3}$, where M, H and W are the frame number, the height, and width of a frame, respectively. Following ViT, each frame is split into $N = \frac{H}{P} \times \frac{W}{P}$ patches, and the patch size is denoted as $P \times P$.

$$\mathbf{V} = [I_1, I_2, \cdots, I_m, \cdots, I_M] \tag{1}$$

$$I_m = [I_{m,1}, I_{m,2}, \cdots, I_{m,N}] \tag{2}$$

Then, this sequence of image patches is encoded with an embedding layer W. An additional token e_{cls} is concatenated at the beginning of the sequence of patches to learn the action representation of the current frame. The encoding process can be written as

$$e_m = e_{m,cls} + W_e^\top [x_{m,\text{cls}}, x_{m,1}, x_{t,2}, \cdots, x_{t,N}] + \mathbf{e}^{\text{spatial}} \tag{3}$$

$$= e_{m,cls} + W_e^\top [x_{m,\text{cls}}, x_{m,1}, x_{t,2}, \cdots, x_{t,N}] + [1, 2, \cdots, N] \tag{4}$$

$$e = [e_1, e_2, \cdots, e_M] \tag{5}$$

where $e_m \in R^{d\times N}$ denotes the frame embedding and $e \in R^{d\times(N\times M)}$ is the embedding of the whole video clip. In the literature of activation-based feature importance, importance can often be measured by the sum of magnitude across all hidden dimensions on that pixel. In this paper, we extend this idea to video processing by summing across the hidden dimension to measure token importance, written as

$$e' = \text{Attention}(e) \ \in R^{d\times(N\times M)} \tag{6}$$

$$w = [1, 1, ..., 1] \in R^{1\times d} \tag{7}$$

$$x = we' \ \in R^{1\times(N\times M)} \tag{8}$$

where e' is the embedding after the Transformer's self-attention, and w is an all-one vector that sums across the hidden dimension of video embeddings.

Let $x_i \in R^{N_i}, x_j \in R^{N_j}$ be the token embedding after the ith and jth iterations. The optimization goal is to reduce the sequence length $N_j \leq N_i \ll (N \times M)$ when $i < j$, thus increasing the throughput of video Transformers. The process is illustrated in Figure 3, which illustrates the temporal and data flow of the proposed methodology. The diagram shows how a video clip consisting of four frames is encoded into embeddings. The Bayesian token pruning module, employing Gumbel-Softmax, is integrated between the self-attention blocks to improve computational efficiency. Additionally, the diagram highlights the importance of attention-based learnable masks, sequentially removing less significant frames, generating a 16-fold efficiency improvement.

Formally, we formulate this ℓ_1 norm regularized classification problem in a well-established sparse coding framework [62], written as

$$F(\mu, x) = f(\mu, x) + g(x_i) \tag{9}$$

$$f(\mu, x) = \|\mu - Wx\|_2 \tag{10}$$

$$g(x) = \|x\|_1 \tag{11}$$

where $||\cdot||_2$ denotes the ℓ_2 norm of the difference between projected feature vectors Wx and its prototype μ. $||\cdot||_1$ denotes the ℓ_1 norm of the deep feature vector x. $f(\mu, x)$ is the ℓ_2 distance indicating the error of matching feature x to the prototype μ, and $g(x)$ is the ℓ_1 regularization term that enforces the sparsity of the feature x. Since the ℓ_1 term has a nondifferentiable point, we can optimize F with the proximal gradient descent algorithm [63], written as

$$\begin{aligned} W_t &= W_{t-1} - \nabla_W F(f, g; W_{t-1}) \\ x_t &= \eta_T(x_{t-1} - \nabla_x F(f, g; x_{t-1})) \\ \eta_T(z) &= \mathrm{sign}(z)(|z| - T) \end{aligned} \tag{12}$$

where $\eta_T(\cdot)$ denotes the shrinkage operator with threshold T, and μ_t represents the sparse action codes at the iteration t. These update equations provide an efficient algorithm for learning compact video representations while maintaining essential visual information captured by Transformers.

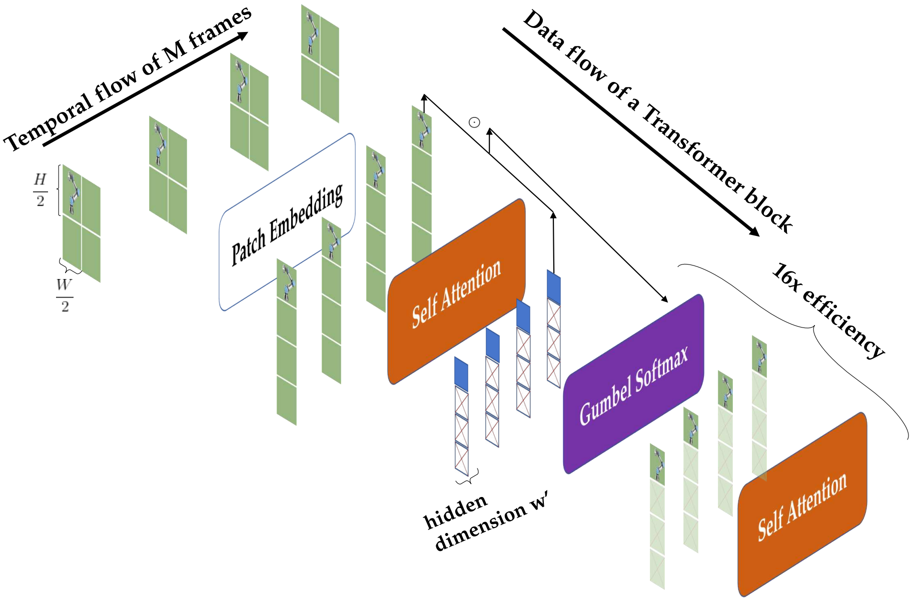


Figure 3. Bayesian token pruning within a ViT backbone where the Gumbel-Softmax module integrates video features and feature importance between the self-attention blocks for efficiency.

3.2. Bayesian Action Recognition

The Maximum A Posteriori (MAP) estimation of sparse action vectors x given observations μ is derived from the Bayes' theorem, where the posterior distribution $P(x \mid \mu)$ is proportional to the product of the likelihood $P(\mu \mid x)$ and prior $P(x)$, written as

$$x_{\mathrm{MAP}} = \arg\max_x \left[\frac{P(\mu|x) \cdot P(x)}{P(\mu)} \right]. \tag{13}$$

$$\propto \arg\max_x [P(\mu|x) \cdot P(x)] \tag{14}$$

$$P(\mu|x) = \prod_{i=1}^{N} G(\mu|x_i) \propto \sum_{i=1}^{N} f(\mu, x_i) \tag{15}$$

where the likelihood $P(\mu|x)$ follows Gaussian distribution if one uses ℓ_2 distance and Bernoulli distribution if one leverages entropy distance [64]. The marginal likelihood $P(\mu)$ is a norm. The prior $P(x)$ is linked to a sparse distribution reparameterized by Gumbel-Softmax [23] in the following section.

To balance the distance between multi-views, we replace the ℓ_2 distance in Equation (10) with InfoNCE distance [65]. This modified training objective increases multi-view comparison by bringing frame representations closer to their prototypes μ_i, repelling representations from prototypes of different action classes,

$$f_{\text{proto}}(\mu, x_i) = -\log \frac{exp\left(\frac{x_i^\top \mu_i - \alpha}{\tau}\right)}{exp\left(\frac{x_i^\top \mu_i - \alpha}{\tau}\right) + \sum_{j \neq i} exp\left(\frac{x_i^\top \mu_j}{\tau}\right)} \tag{16}$$

where the product similarity to the right camera views (prototypes) are encouraged to have a margin α over similarities to other views. With moving cameras, some views may not have samples. These view-specific missing prototypes are filled with the average of prototypes within that action class.

3.3. Bayesian Token Pruning

Intuitively, we want to find a threshold T to preserve semantically important tokens and to remove redundant tokens. However, the hyperparameter search will significantly increase the training cost of the sparse training. To address this problem, we can replace the hard thresholding with variational dropout using Gumbel-Softmax. That is, we can keep or remove the pth token according to the token importance x_{ip} measured in the ith iteration. Furthermore, to enable faster convergence, we can embed redundancy and semantic-awareness knowledge into the model as,

$$\begin{aligned}
a_{pq} &= \langle \tilde{x}_{Kp}, \tilde{x}_{Kq} \rangle = x_p^\top W_K^\top W_K x_q \\
a_{cp} &= \langle \tilde{x}_p, \tilde{\mu}_c \rangle = x_p^\top W_K^\top W_K \mu_c \\
s_{ip} &= (1 - \sum_{q \in Q} a_{pq} + \max_{c \in C} a_{cp}) \cdot x_{ip}.
\end{aligned} \tag{17}$$

where the similarity-based token importance is measured by the inner product between feature vectors. a_{cp} is the similarity between the prototype μ_c and the pth token in the attention space, and a_{pq} measures the redundancy of the pth token, measured by the similarity to the same position in previous frames. Q denotes the set of previous frames. Concretely, we compute the token importance scores with the first key projection head W_K following ToMe [59].

Finally, we introduce the sparse prior term $P(x_i)$ selecting important tokens by applying the Gumbel-Softmax reparameterization trick to the affinity scores. The Gumbel noise ϵ_i is sampled from a uniform distribution, imposing an equal prior assumption on tokens, written as,

$$\begin{aligned}
P(x_i) &\propto \text{gumbel\_softmax}(s_{ip}) \\
&= \frac{\exp\left(\left(\log(s_{ip}) + \epsilon_i\right)/\rho\right)}{\sum_{r=1}^{R} \exp\left(\left(\log(s_{ip}) + \epsilon_r\right)/\rho\right)} \\
&\overset{R=2}{=} \text{sigmoid}\left(\log(s_{ip} + \epsilon_i)/\rho\right),
\end{aligned} \tag{18}$$

where $\rho \in [0,1]$ is the temperature parameter that controls thresholding. The Gumbel-Softmax provides an eco-friendly pruning operation, reducing the burden of exhaustive threshold searching.

3.4. Summary of the Algorithm

The pseudo-code for our probabilistic token pruning method is given in Algorithm 1. To summarize, ProtoPrune improves the efficiency of video Transformers by dynamically pruning redundant video frames while preserving semantically important information. The algorithm begins by initializing prototypes μ using K-means clustering over frame-level features, allowing prototypes to capture typical sub-action patterns. During training, features are assigned to their nearest prototypes, which minimizes the distance between frame features and their cluster centers.

Algorithm 1 Prototype-Aware Token Pruning (ProtoPrune)

```
Input: Model pretrained for general video processing.
Step 1: Initialize prototypes $\mu$ with K-means clustering over features $x$;
for epoch in total_epochs do
    Step 2: Sample a mini-batch of clips $\{V_1, V_2, \cdots, V_N\}$;
    Apply clustering to clips from the same action class.
    for i in total_frames do
        Step 3: Compute feature embeddings $x_t$ of each patch;
        Step 4: Assign each frame to a cluster;
        Step 5: Compute frame importance scores based on the Equation (17);
        Step 6: Apply the Gumbel-Softmax to the scores to generate binary masks;
        Step 7: Generate the sparse feature $x_i$ with the mask;
    end for
    Step 8: Use the objective $F$ to finetune the backbone;
    Update model weights $W$ and prototypes $\mu$;
end for
Output: Efficient video Transformers and prototypes.
```

In Step 5, feature importance scores are computed based on the similarity to prototypes and temporal redundancy described in Equation (17). Then, in Step 6, this score goes into the Gumbel-Softmax operator and generates a binary mask for the frame, as shown in Step 7. After that, the model and prototypes are jointly updated with a prototypical contrastive loss. This loss encourages frames from the same sub-action to converge toward their sub-action prototypes, improving intra-cluster cohesion. Finally, after all training epochs elapse, the result outputs an efficient video Transformer where retained video tokens correspond to key information for action recognition.

4. Convergence Analysis

In this section, we provide a convergence analysis for the proposed sparse action recognition algorithm with the following assumption. Before we embark on the convergence analysis of ProtoPrune, it is essential to establish two key lemmas based on some assumptions.

Assumption 1. *Assume the predictor f can be approximated by its second-order Taylor expansion in its definition field. Then, the quadratic lower bound of $F(x)$ at the point x_t can be written as*

$$Q(x, x_t) = f(x_t) + \langle x - x_t, \nabla f(x_t) \rangle + \frac{L}{2} \|x - x_t\|^2 + g(x). \tag{19}$$

Assumption 2. *The predictor f and its first-order differential satisfy the Lipschitz smoothness condition in the field of its definition. Then, the first-order differential is upper bounded by the Lipschitz constant of function f,*

$$|\nabla f(x) - \nabla f(x_t)| \leqslant L(f)|x - x_t|. \tag{20}$$

Lemma 1. *(First-Order Optimal Condition, FOC) For the optimal deep feature x_t, let f be the distance function and L be the Lipschitz constant defined in Assumption 2. If $g(\cdot)$ is not differentiable at x and then there exists the subdifferential $\gamma(x_t) \in \partial g(\eta_T(x_t))$, such that*

$$\nabla f(x_t) + L(\eta_T(x_t) - x_t) + \gamma(x_t) = 0 \tag{21}$$

Proof. The proof is immediate from the optimality conditions of the strong convexity. □

Lemma 2. *Let x^* be the optimal deep feature, and $\eta_T(x_t)$ be the sparse representation after t iterations are shrunk with a threshold T; if $F(\eta_T(x_2)) \geqslant Q(\eta_T(x_t))$, then*

$$F(x^*) - F(\eta_T(x_t)) \geqslant \frac{L}{2}\|\eta_T(x_t) - x_t\|^2 + L\langle x_t - x^*, x_t - \eta_T(x_t)\rangle \tag{22}$$

Proof. From the quadratic lower bound, we have

$$F(x^*) \geqslant Q(x^*, \eta_T(x_t)) \tag{23}$$

From the convexity defined in Equations (10) and (11), we have

$$\begin{aligned} f(x^*) &\geqslant f(x_t) + \nabla f(x_t)(x^* - x_t) \\ g(x^*) &\geqslant g(\eta_T(x_t)) + \gamma(x_t)(x^* - \eta_T(x_t)) \end{aligned} \tag{24}$$

in which $\gamma(x_t)$ is defined in Lemma 1. Consequently, we can derive

$$\begin{gathered} F(x^*) \geq f(x_t) + g(\eta_T(x_t)) + \langle x^* - x_t, \nabla f(x_t)\rangle + \langle x^* - \eta_T(x_t), \gamma(x_t)\rangle + \frac{L}{2}\|x^* - x_t\|^2 \\ \|x^* - x_t\|^2 \geq \|\eta_T(x_t) - x_t\|^2 \end{gathered} \tag{25}$$

By substituting Equations (19) and (25) into Equation (23), yields

$$\begin{aligned} F(x^*) - F(\eta_T(x_t)) + Q(\eta_T(x_t)) &\geq -\frac{L}{2}\|\eta_T(x_t) - x_t\|^2 + \langle x^* - \eta_T(x_t), \nabla f(x_t) + \gamma(x_t)\rangle \\ &= -\frac{L}{2}\|\eta_T(x_t) - x_t\|^2 + L\langle x^* - \eta_T(x_t), x_t - \eta_T(x_t)\rangle \\ &= \frac{L}{2}\|\eta_T(x_t) - x_t\|^2 + L\langle x_t - x^*, \eta_T(x_t) - x_t\rangle. \end{aligned} \tag{26}$$

That completes the proof. □

Theorem 1. *Let $\{x_n^*\}$ be the sequence of deep features generated by our sparse encoding optimization. Then for any $n \geq 1$, the sparse feature x_t iteratively optimized in Algorithm 1 will converge to the neighborhood of an optimal compressed feature with $O(1/k)$,*

$$F(x_t) - F(x^*) \leq \frac{L(f)\|x_0 - x^*\|^2}{2k} \tag{27}$$

Proof. Invoking Lemma 2 and set $L = L_{n+1}$, we obtain

$$\begin{aligned} \frac{2}{L_{n+1}}(F(x^*) - F(x_t)) &\geq \|x_t - x_{t-1}\|^2 + 2\langle x_{t-1} - x^*, x_t - x_{t-1}\rangle \\ &= \|x^* - x_t\|^2 - \|x^* - x_{t-1}\|^2 \end{aligned} \tag{28}$$

Combining the upper bound and the optimal condition of $F(x^*) - F(x_t) \leq 0$, we can derive

$$\frac{2}{L(f)}(F(x^*) - F(x_t)) \geq \|x^* - x_t\|^2 - \|x^* - x_{t-1}\|^2. \tag{29}$$

Summing this inequality over $t = 0, \ldots, k-1$ gives

$$\frac{2}{L(f)}\left(kF(x^*) - \sum_{t=0}^{k-1} F(x_t)\right) \geq \|x^* - x_t\|^2 - \|x^* - x_0^*\|^2. \tag{30}$$

Invoking Lemma 2 again yields

$$\frac{2}{L_{n+1}}(F(x_{t-1}) - F(x_t)) \geq \|x_{t-1} - x_t\|^2 \tag{31}$$

Since the lower bound of the Lipschitz constant $L_{n+1} \geq L(f)$, it follows that

$$\frac{2}{L(f)}(F(x_{t-1}) - F(x_t)) \geq \|x_{t-1} - x_t\|^2. \tag{32}$$

Multiply the inequality (32) by n and sum over $t = 0, \dots, k-1$ to obtain

$$\frac{2}{L(f)} \sum_{n=0}^{k-1} (nF(x_{t-1}) - (n+1)F(x_t) + F(x_t)) \geq \sum_{n=0}^{k-1} n\|x_{t-1} - x_t\|^2 \tag{33}$$

which simplifies to

$$\frac{2}{L(f)} \left(-kF(x_t) + \sum_{n=0}^{k-1} F(x_t) \right) \geq \sum_{n=0}^{k-1} n\|x_{t-1} - x_t\|^2 \tag{34}$$

Adding Equations (30) and (34), we obtain

$$\frac{2k}{L(f)}(F(x^*) - F(x_t)) \geq \|x^* - x_t\|^2 + \sum_{n=0}^{k-1} n\|x_{t-1} - x_t\|^2 - \|x^* - x_0\|^2 \tag{35}$$

and hence it follows that

$$F(x_t) - F(x^*) \leq \frac{L(f)\|x^* - x_0\|^2}{2k} \tag{36}$$

That completes the proof. □

5. Experiments

5.1. Datasets and Metrics

We evaluated the accelerated inference performance of the proposed action recognition methods with state-of-the-art baselines on the Kinetics-400 [66] dataset, Something-Something V2 (SSV2) [67] dataset, and an HRC testbed [68]. The main features of these datasets are summarized in Table 3. The Kinetics-400 dataset focuses on general human activity recognition across diverse actions. In contrast, SSV2 focuses on fine-grained interactions from an egocentric (first-person view) perspective, while human–robot collaboration (HRC) datasets address action recognition with varying illumination.

Table 3. Main features of Kinetics-400 and SSV2 datasets.

Feature	Kinetics-400	SSV2	HRC
Classes	400	174	4
#Videos/Images	∼260,000	∼200,000	13,926
Duration	∼10 s	∼ 2–6 s	∼2 min
Frame Rate	25 fps	12 fps	30 fps
Focus	general	egocentric, FPV	illumination
Source	YouTube	crowd-sourced	laboratory

To assess how well the model can classify samples, we report Top-1 accuracy on the test set. The retention rate [69] is computed as the ratio between the performance of the pruned model and the original model. We report the performance retention ratio

(PRR) as the harmonic mean of the ratio between pruned and raw models. We weigh the computational complexity with giga floating-point operations (GFLOPs).

5.2. Implementation Details

Our experiments investigate human action recognition in a collaborative production environment, focusing on inference acceleration. The model is expected to understand actions by correctly predicting the category and estimating the importance of the tokens. An efficient action recognition algorithm should capture a range of visual cues pivotal for action categories, including inspect, pick, assemble, or place. These identified actions may be split into different shots, which requires detailed analysis. Our experiments employ two video Transformer baselines, UniFormer [70] and ActionCLIP [71]. The choice of using Uniformer and ActionCLIP as comparative baselines is based on their task alignment for action recognition and their rigorous supervised training procedures with established performance on standard benchmarks. We finetune ActionCLIP on Kinetics-400 and SSV2 in the experiments of ProtoPrune. To evaluate the importance of prototypes, we replace the prototypes with the [Score] token for the ablation study. In our experiments, we split 10% of the training set to create a validation dataset for hyperparameter tuning. Additionally, we leverage weights and biases [72] to facilitate an efficient hyperparameter search, with the key hyperparameters listed in Table 4. To handle dynamic environments, we incorporate widely used image augmentation techniques into the ProtoPrune, including random resized crop, horizontal flip, and color jittering, to improve the robustness of the action recognition algorithm against environmental factors. The hardware requirement is four Tesla 32G V100 GPUs.

Table 4. Key hyperparameters for Bayesian Prototypical Learning.

Hyperparameter	Value	Description
Number of Prototypes	4	#views that limit the granularity of actions
Temperature	0.5	Gumbel-Softmax's hyperparameter for sparsity
Learning Rate (LR)	1×10^{-4}	Base LR with cosine decay
Batch Size	1024	Number of samples in a mini-batch
Weight Decay	0.05	Weight regularization term
Warmup Epochs	5	Epochs to stabilize early training
Total Epochs	100/30	Epochs on Kinetics-400/SSV2

5.3. Results

Table 5 provides a clear comparison of ProtoPrune with baselines on Kinetics-400 on the test splits of Kinetics-400 and SSV2. Overall, ProtoPrune learns token pruning functions on video data distributions, providing inference acceleration with a high retention rate. For the ablation study, we mark the importance of prototypes by thresholding $f(x_i)$ with respect to the [Score] token of Uniformer, as shown in the "Ablation" group in the second last row of the table. By reducing redundant video content, the method shows fewer GFLOPs in two setups, three image crops per frame with four shots (32 × 3 × 4) and one image crop per frame with four shots (16 × 1 × 4). When applying ProtoPrune to UniFormer, the model achieves a high-performance retention rate of 94.8% while reducing 37.2% of GFLOPs. When adapting the model from Kinetics to SSV2, ProtoPrune classifies 70.2% of frames accurately in the top-1 prediction. The method outperforms CLIP and Action-CLIP by an obvious margin, demonstrating the gap between prototypical and textual supervision. The performance retention rate drops from 94.8% to 70.7%, demonstrating prototypical semantic-awareness is the major component of the algorithm.

Table 5. Comparison of ProtoPrune and baselines on Kinetics-400 and SSV2.

Method	Kinetics-400			SSV2			PRR
	#frame	GFLOPs↓	Top-1↑	#frame	GFLOPs↓	Top-1↑	
I3D [42]	16×1×10	108	72.1	16×2×8	167.8	62.8	-
SlowFast [73]	32×3×10	12,720	77.0	32×3×10	12,720	58.4	-
CLIP [74]	8×1×1	149.1	57.5	8×1×1	149.1	5.1	-
ActionCLIP [71]	8×1×1	149.1	52.6	8×1×1	149.1	69.6	-
Dropout [21]	8×1×1	$92.6_{\downarrow 37.9\%}$	18.3	8×1×1	$92.6_{\downarrow 37.9\%}$	10.3	17.9
ProtoPrune	8×1×1	$92.6_{\downarrow 37.9\%}$	74.1	8×1×1	$92.6_{\downarrow 37.9\%}$	62.7	91.0
UniFormer [70]	16×1×4	389	82.0	16×3×1	290	70.2	-
Dropout [21]	16×1×4	$244.3_{\downarrow 37.2\%}$	24.7	16×3×1	$182.7_{\downarrow 37\%}$	10.9	20.5
Ablation	16×1×4	$244.3_{\downarrow 37.2\%}$	49.2	16×3×1	$182.7_{\downarrow 37\%}$	47.3	63.5
ProtoPrune	16×1×4	$244.3_{\downarrow 37.2\%}$	**75.9**	16×3×1	$182.7_{\downarrow 37\%}$	**65.4**	**92.9**

Figure 4 visualizes the performance of ProtoPrune in human–robot collaboration. We select image patches to represent prototypes of typical actions in multi-view HRC environments, such as "command" and "handover", shown in the last row. With semantic awareness, ProtoPrune can capture salient features for action recognition, such as the chessboard, the moving body, and the robot arm. By contrast, more noisy background objects are included in the comparison group, with parts of the human body and the robot arm mistakenly pruned. Once pruned, information encapsulated in these tokens is lost in the baseline. By contrast, ProtoPrune is good at reweighing the importance of tokens with its viable solution.

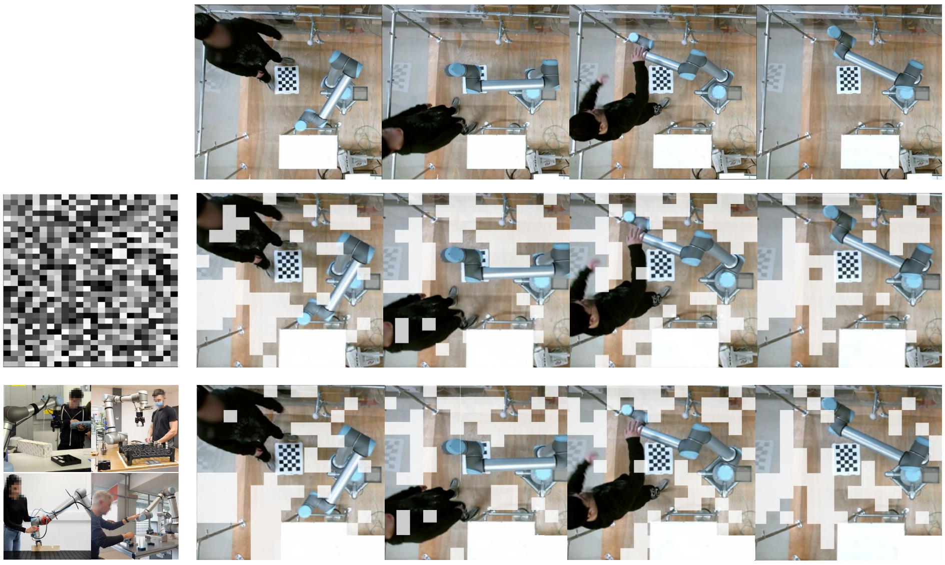


Figure 4. Visualization of pruned video tokens. The first row shows the input frames followed by tokens pruned with similarity to the [Score] token; the third row shows 4 prototypes of shared control.

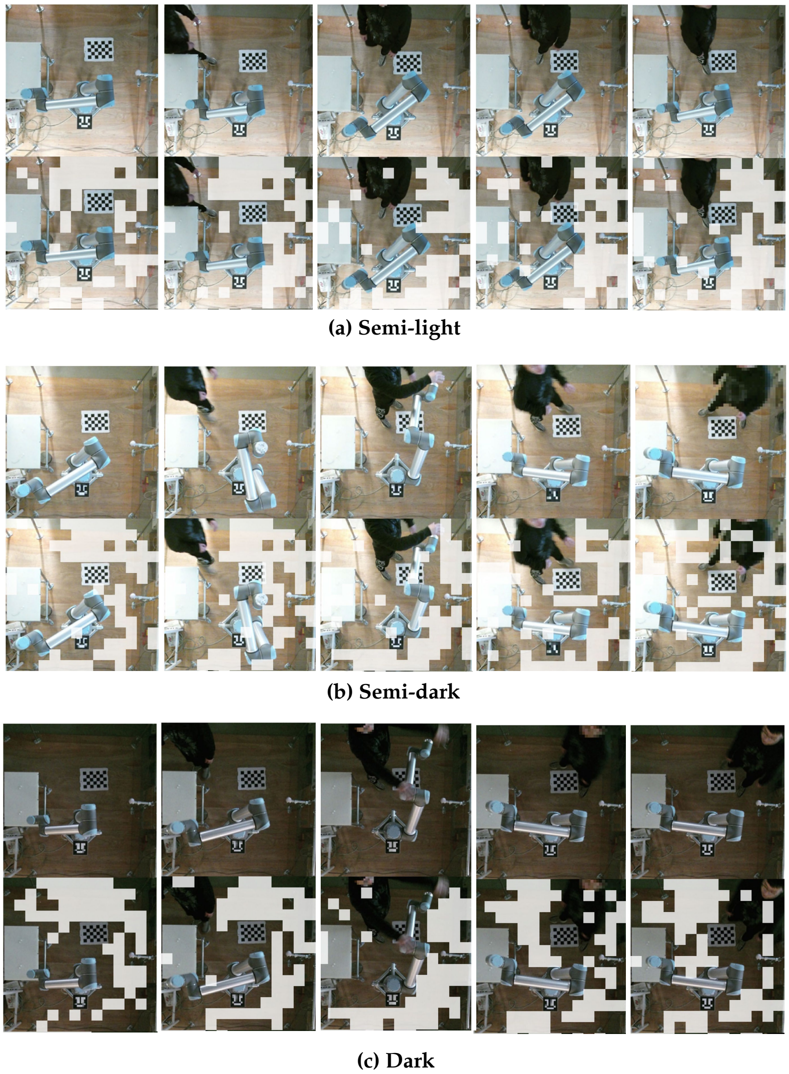
Table 6 compares the action recognition performance of ProtoPrune and baselines under four illumination conditions. We use the same hyperparameters, including the number of prototypes and Gumbel-Softmax temperatures, under four illumination conditions (light, semi-light, semi-dark, dark). Overall, the method maintains high accuracy (e.g., 96.4% PRR during illumination change) while having less computational complexity than ActionCLIP. In the Light condition, ActionCLIP achieves a human-level action recognition accuracy of 99.1%. In the Dark condition, ProtoPrune maintains a stable performance of 89.9% accuracy, demonstrating only a marginal drop compared to ActionCLIP (91.6%). While ActionCLIP achieves the best accuracy across four illumination conditions, it comes at a significant computational cost, requiring 149.1 GFLOPs. In stark contrast, ProtoPrune achieves competitive performance with only 92.6 GFLOPs, which is a 37.9% reduction in computational cost. The comparison demonstrates the proposed method's adaptability to low-light conditions, which is crucial for real-world HRC environments.

Table 6. Performance analysis in HRC environments under four illumination conditions.

Method	GFLOPs	Light	Semi-Light	Semi-Dark	Dark	PRR
ActionCLIP [71]	149.1	99.1	95.4	92.3	91.6	-
Dropout [21]	92.6	40.2	38.4	35.7	35.5	39.6
ProtoPrune	92.6	94.7	90.3	89.7	89.9	96.4

Figure 5 visualizes ProtoPrune under three different lighting groups. The images represent lighting conditions: semi-light, semi-dark, and dark. Each group of images is organized horizontally, presenting a sequential batch of sampled frames, with the second row in each group specifically illustrating the token pruning results, where pruned patches are filled in white. ProtoPrune effectively preserves essential visual elements, such as humans, calibration boards, and robotic components, within varying lighting conditions. In the idling mode (depicted in the leftmost images), the algorithm adeptly eliminates background noise, including the high-contrast white table, from the model's focus. Remarkably, even under dark conditions, key structural details are retained, allowing the system to sustain task continuity. This capability is particularly advantageous in industrial environments, where maintaining a safe workflow is important. As the illumination becomes dark, the mistake of pruning the operator's patch is observed. The resultant information loss regarding subtle visual cues, such as clothing patterns, poses challenges for critical tasks, including person identification, workload tracking, and safety monitoring. Notably, these errors are context-dependent and can be mitigated by overweighing these samples during training.

Figure 6 presents a detailed visualization of the learned action representations with t-distributed stochastic neighbor embedding (t-SNE) [75]. In this visualization, each small cluster composed of eight points corresponds to a distinct video clip, with each individual point representing a specific frame within that clip. In the left diagram, we observe clusters are marked by excessive compactness. This phenomenon suggests that the action representations generated under these conditions fail to adequately capture the diversity and complexity inherent in the underlying data. In stark contrast, the application of ProtoPrune introduces a regularization framework that significantly improves action representation learning through the integration of prototype similarity or sub-action consciousness. As a result, the fine-grained action representations reflect a deeper, more nuanced understanding of human behaviors, allowing for more precise differentiation between actions and a clearer delineation of their semantic overlaps. This enhanced representation of actions balances the intra-class and inter-class distance in video frames, contributing to better classification while facilitating video token pruning for better computational efficiency.



(a) Semi-light

(b) Semi-dark

(c) Dark

Figure 5. Visualization of ProtoPrune under semi-light, semi-dark, and dark illumination conditions. Each illumination group displays a temporal flow of five sequential frames (**top row**) and the pruned results (**bottom row**), where pruned patches are filled in white to highlight pruning effects.

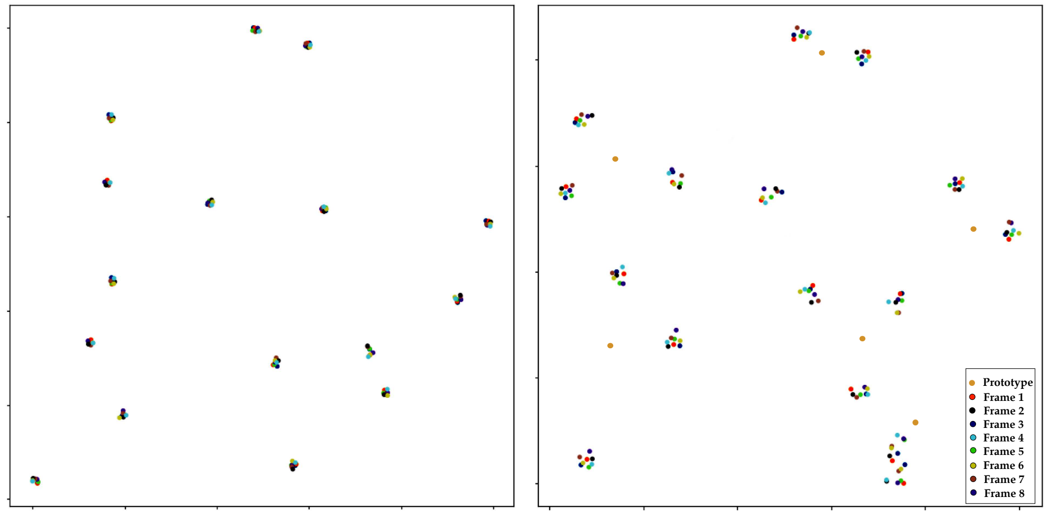


Figure 6. T-SNE plots without token pruning (**left**) and with ProtoPrune (**right**).

6. Conclusions

6.1. Main Contributions

In this paper, we propose a novel Bayesian Prototypical Pruning (ProtoPrune) method to address the critical challenge of computational efficiency in video understanding. By combining prototypical contrastive learning with attention-based Bayesian sampling, our method enables efficient and interpretable token pruning without the need for an exhaustive threshold search. We use theoretical and experimental analysis to show that ProtoPrune can optimize the computational efficiency of video Transformer models.

6.2. Main Results

Our experiments on Kinetics-400 demonstrate that ProtoPrune reduces GFLOPs by 37.2% while retaining 92.9% of the original performance. When transferred to the SSV2 dataset, ProtoPrune achieves 70.2% top-1 accuracy and outperforms both CLIP and Action-CLIP baselines. The ablation studies confirm that prototypical similarity-based task awareness is crucial for effectiveness, supported by the performance retention rate change from 92.9%

6.3. Limitation and Future Work

In this work, we observed that the temperature parameter in Gumbel-Softmax is sensitive to data distributions. An effective approach to address the corresponding overfitting problem is to select a good initial point. For video Transformer pruning, we typically set the initial temperature to 0.5 and allow the temperature to be adaptive and gradually change over time, which helps mitigate the overfitting problem. Due to time constraints, this study only investigates the use of a uniform distribution as the prior for Gumbel-Softmax. In fact, other prior distributions could also be considered. For example, employing a Dirichlet distribution allows control over weight sparsity by adjusting its parameters (the α vector) and offers a better interpretation of weight importance. Future work will explore other priors. Another direction of our future work will focus on environmental factors to facilitate safety monitoring, including occlusion and reflectiveness.

Author Contributions: Conceptualization, methodology, software, validation, B.P.; formal analysis, B.P. and B.C; writing—original draft preparation, B.P.; writing—review and editing, B.P. and B.C.; supervision and project administration, B.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: This paper leverages publicly available datasets with license agreements from the corresponding institutions with proper channels.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Orsag, L.; Koren, L. Towards a Safe Human–Robot Collaboration Using Information on Human Worker Activity. *Sensors* **2023**, *23*, 1283.
2. Peng, B.; Chen, B.; He, W.; Kadirkamanathan, V. Prototype-aware Feature Selection for Multi-view Action Prediction in Human-Robot Collaboration. In Proceedings of the 2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA), Padova, Italy, 10–13 September 2024; pp. 1–7. https://doi.org/10.1109/ETFA61755.2024.10710750.
3. Yan, Y.; Su, H.; Jia, Y. Modeling and analysis of human comfort in human–robot collaboration. *Biomimetics* **2023**, *8*, 464.
4. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.

5. Girdhar, R.; Carreira, J.; Doersch, C.; Zisserman, A. Video action transformer network. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 244–253.
6. Lou, S.; Hu, Z.; Zhang, Y.; Feng, Y.; Zhou, M.; Lv, C. Human-cyber-physical system for Industry 5.0: A review from a human-centric perspective. *IEEE Trans. Autom. Sci. Eng.* **2024**, *22*, 494–511.
7. Ramirez-Amaro, K.; Yang, Y.; Cheng, G. A survey on semantic-based methods for the understanding of human movements. *Robot. Auton. Syst.* **2019**, *119*, 31–50.
8. Wang, B.; Zhang, X.; Zhao, Y. Exploring sub-action granularity for weakly supervised temporal action localization. *IEEE Trans. Circuits Syst. Video Technol.* **2021**, *32*, 2186–2198.
9. Karpathy, A.; Toderici, G.; Shetty, S.; Leung, T.; Sukthankar, R.; Fei-Fei, L. Large-scale video classification with convolutional neural networks. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 28 2014; pp. 1725–1732.
10. Jahanmahin, R.; Masoud, S.; Rickli, J.; Djuric, A. Human–robot interactions in manufacturing: A survey of human behavior modeling. *Robot. Comput. Integr. Manuf.* **2022**, *78*, 102404.
11. Tian, Y.; Krishnan, D.; Isola, P. Contrastive Multiview Coding. In Proceedings of the European Conference on Computer Vision, Seoul, Republic of Korea, 27 October 27–2 November 2019.
12. Chen, Z.; Wan, Y.; Liu, Y.; Valera-Medina, A. A knowledge graph-supported information fusion approach for multi-faceted conceptual modelling. *Inf. Fusion* **2023**, *101*, 101985.
13. Richard, A.; Kuehne, H.; Gall, J. Weakly supervised action learning with rnn based fine-to-coarse modeling. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 754–763.
14. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv* **2020**, arXiv:2010.11929.
15. Zhang, P.; Tian, C.; Zhao, L.; Duan, Z. A multi-granularity CNN pruning framework via deformable soft mask with joint training. *Neurocomputing* **2024**, *572*, 127189.
16. Wang, H.; Zhang, Z.; Han, S. Spatten: Efficient sparse attention architecture with cascade token and head pruning. In Proceedings of the 2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA), Seoul, Republic of Korea, 27 February–3 March 2021; pp. 97–110.
17. Chitty-Venkata, K.T.; Mittal, S.; Emani, M.; Vishwanath, V.; Somani, A.K. A survey of techniques for optimizing transformer inference. *J. Syst. Archit.* **2023**, *144*, 102990.
18. Kim, S.; Shen, S.; Thorsley, D.; Gholami, A.; Kwon, W.; Hassoun, J.; Keutzer, K. Learned token pruning for transformers. In Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 14–18 August 2022; pp. 784–794.
19. Peng, B.; Islam, M.; Tu, M. Angular Gap: Reducing the Uncertainty of Image Difficulty through Model Calibration. In Proceedings of the 30th ACM International Conference on Multimedia, New York, NY, USA, 10–14 October 2022; MM ’22, pp. 979–987. https://doi.org/10.1145/3503161.3548289.
20. Hu, J.F.; Zheng, W.S.; Lai, J.; Zhang, J. Jointly learning heterogeneous features for RGB-D activity recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 12 2015; pp. 5344–5352.
21. Molchanov, D.; Ashukha, A.; Vetrov, D. Variational dropout sparsifies deep neural networks. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; pp. 2498–2507.
22. Liu, Y.; Dong, W.; Zhang, L.; Gong, D.; Shi, Q. Variational bayesian dropout with a hierarchical prior. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 7124–7133.
23. Jang, E.; Gu, S.; Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv*, **2017**, arXiv:1611.01144.
24. Patalas-Maliszewska, J.; Dudek, A.; Pajak, G.; Pajak, I. Working toward solving safety issues in human–robot collaboration: A case study for recognising collisions using machine learning algorithms. *Electronics* **2024**, *13*, 731.
25. Liu, Y.; Zhang, W.; Cheng, Q.; Ming, D. Efficient Reachable Workspace Division under Concurrent Task for Human-Robot Collaboration Systems. *Appl. Sci.* **2023**, *13*, 2547.
26. Tuli, T.B.; Kohl, L.; Chala, S.A.; Manns, M.; Ansari, F. Knowledge-Based Digital Twin for Predicting Interactions in Human-Robot Collaboration. In Proceedings of the 2021 26th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Västerås, Sweden, 7–10 September 2021; pp. 1–8.
27. Malik, A.A.; Masood, T.; Bilberg, A. Virtual reality in manufacturing immersive and collaborative artificial-reality in design of human–robot workspace. *Int. J. Comput. Integr. Manuf.* **2020**, *33*, 22–37.
28. Guerra-Zubiaga, D.A.; Kuts, V.; Mahmood, K.; Bondar, A.; Esfahani, N.N.; Otto, T. An approach to develop a digital twin for industry 4.0 systems: Manufacturing automation case studies. *Int. J. Comput. Integr. Manuf.* **2021**, *34*, 933–949.
29. Hanna, A.; Bengtsson, K.; Götvall, P.L.; Ekström, M. Towards safe human robot collaboration - Risk assessment of intelligent automation. In Proceedings of the 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Vienna, Austria, 8–11 September 2020; Volume 1, pp. 424–431. https://doi.org/10.1109/ETFA46521.2020.9212127.

30. Inaba, M.; Guo, H.J.; Nakao, K.; Abe, K. Adaptive control systems switched by control and robust performance criteria. In Proceedings of the 1996 IEEE Conference on Emerging Technologies and Factory Automation (ETFA '96), Kauai Marriott, HI, USA, 18–21 November 1996; Volume 2, pp. 690–696 https://doi.org/10.1109/ETFA.1996.573988.
31. Demir, K.A.; Döven, G.; Sezen, B. Industry 5.0 and Human-Robot Co-working. *Procedia Computer Science* **2019**, *158*, 688–695. In Proceedings of the 3rd World Conference on Technology, Innovation and Entrepreneurship" Industry 4.0 Focused Innovation, Technology, Entrepreneurship and Manufacture, Istanbul, Turkey, 21–23 June 2019. https://doi.org/10.1016/j.procs.2019.09.104.
32. Pini, F.; Ansaloni, M.; Leali, F. Evaluation of operator relief for an effective design of HRC workcells. In Proceedings of the 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA), Berlin, Germany, 6–9 September 2016; pp. 1–6. https://doi.org/10.1109/ETFA.2016.7733526.
33. Likitlersuang, J.; Sumitro, E.R.; Cao, T.; Visée, R.J.; Kalsi-Ryan, S.; Zariffa, J. Egocentric video: A new tool for capturing hand use of individuals with spinal cord injury at home. *J. Neuroeng. Rehabil.* **2019**, *16*, 1–11.
34. Zhuang, Y.; Rui, Y.; Huang, T.S.; Mehrotra, S. Adaptive key frame extraction using unsupervised clustering. In Proceedings of the 1998 International Conference on Image Processing. icip98 (cat. no. 98cb36269), Chicago, IL, USA, 7 October 1998; Volume 1, pp. 866–870.
35. De Avila, S.E.F.; Lopes, A.P.B.; da Luz, A.; de Albuquerque Araújo, A. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. *Elsevier PRL* **2011**, *32*, 56–68.
36. Kuanar, S.K.; Panda, R.; Chowdhury, A.S. Video key frame extraction through dynamic Delaunay clustering with a structural constraint. *J. Vis. Commun. Image Represent.* **2013**, *24*, 1212–1227.
37. Gharbi, H.; Bahroun, S.; Massaoudi, M.; Zagrouba, E. Key frames extraction using graph modularity clustering for efficient video summarization. In Proceedings of the ICASSP, New Orleans, LA, USA, 5–9 March 2017.
38. Wang, Q.; He, X.; Jiang, X.; Li, X. Robust bi-stochastic graph regularized matrix factorization for data clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *44*, 390–403.
39. Lucci, N.; Preziosa, G.F.; Zanchettin, A.M. Learning Human Actions Semantics in Virtual Reality for a Better Human-Robot Collaboration. In Proceedings of the 2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN), Napoli, Italy, 29 August–1 September 2022; pp. 785–791.
40. Guo, K.; Ishwar, P.; Konrad, J. Action recognition from video using feature covariance matrices. *IEEE Trans. Image Process.* **2013**, *22*, 2479–2494.
41. Ji, S.; Xu, W.; Yang, M.; Yu, K. 3D Convolutional Neural Networks for Human Action Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 221–231. https://doi.org/10.1109/TPAMI.2012.59.
42. Carreira, J.; Zisserman, A. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6299–6308.
43. Neimark, D.; Bar, O.; Zohar, M.; Asselmann, D. Video transformer network. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, BC, Canada, 11–17 October 2021; pp. 3163–3172.
44. Choi, J.; Gao, C.; Messou, J.C.E.; Huang, J.B. Why cannot I dance in a mall? Learning to mitigate scene bias in action recognition. In Proceedings of the 33rd International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 8–14 December 2019; pp. 77–89.
45. Korban, M.; Li, X. DDGCN: A Dynamic Directed Graph Convolutional Network for Action Recognition. In Proceedings of the Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, 23–28 August 2020; Part XX, Volume 16.
46. Hussain, T.; Muhammad, K.; Ding, W.; Lloret, J.; Baik, S.W.; De Albuquerque, V.H.C. A comprehensive survey of multi-view video summarization. *Pattern Recognit.* **2021**, *109*, 107567.
47. Wang, H.; Yang, J.; Yang, L.T.; Gao, Y.; Ding, J.; Zhou, X.; Liu, H. MvTuckER: Multi-view knowledge graphs representation learning based on tensor tucker model. *Inf. Fusion* **2024**, *106*, 102249.
48. Lucas, B.D.; Kanade, T. An Iterative Image Registration Technique with an Application to Stereo Vision. In Proceedings of the International Joint Conference on Artificial Intelligence, Vancouver, BC, Canada, 24–28 August 1981.
49. Sevilla-Lara, L.; Liao, Y.; Güney, F.; Jampani, V.; Geiger, A.; Black, M.J. On the Integration of Optical Flow and Action Recognition. In Proceedings of the German Conference on Pattern Recognition, Basel, Switzerland, 13–15 September 2017.
50. Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; Gool, L.V. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
51. Sun, Deqing and Roth, Stefan and Black, Michael J. Secrets of Optical Flow Estimation and Their Principles. In Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA, 13–18 June 2010.
52. Donoho, D.L. Compressed sensing. *IEEE Trans. Inf. Theory* **2006**, *52*, 1289–1306.
53. Candès, E.J.; Romberg, J.; Tao, T. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory* **2006**, *52*, 489–509.

54. Zheng, C.; Li, Z.; Zhang, K.; Yang, Z.; Tan, W.; Xiao, J.; Ren, Y.; Pu, S. SAViT: Structure-Aware Vision Transformer Pruning via Collaborative Optimization. In Proceedings of the Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022.
55. Frantar, E.; Alistarh, D. SparseGPT: Massive language models can be accurately pruned in one-shot. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023.
56. Liu, Z.; Wang, J.; Dao, T.; Zhou, T.; Yuan, B.; Song, Z.; Shrivastava, A.; Zhang, C.; Tian, Y.; Re, C.; et al. Deja vu: Contextual sparsity for efficient llms at inference time. In Proceedings of the International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; pp. 22137–22176.
57. Feng, Z.; Zhang, S. Efficient vision transformer via token merger. *IEEE Trans. Image Process.* **2023**, *32*, 4156–4169.
58. Park, S.H.; Tack, J.; Heo, B.; Ha, J.W.; Shin, J. K-centered patch sampling for efficient video recognition. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022, pp. 160–176.
59. Bolya, D.; Fu, C.Y.; Dai, X.; Zhang, P.; Feichtenhofer, C.; Hoffman, J. Token Merging: Your ViT but Faster. In Proceedings of the International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
60. Kim, M.; Gao, S.; Hsu, Y.C.; Shen, Y.; Jin, H. Token fusion: Bridging the gap between token pruning and token merging. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, Waikoloa, HI, USA, 3–8 January 2024; pp. 1383–1392.
61. Ishibashi, R.; Meng, L. Automatic pruning rate adjustment for dynamic token reduction in vision transformer. *Appl. Intell.* **2025**, *55*, 1–15.
62. Gao, S.; Tsang, I.W.H.; Chia, L.T. Sparse representation with kernels. *IEEE Trans. Image Process.* **2012**, *22*, 423–434.
63. Nitanda, A. Stochastic proximal gradient descent with acceleration techniques. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS'14), Montreal, QC, Canada, 8–13 December 2014; pp. 1574–1582.
64. Deisenroth, M.; Faisal, A.A.; Ong, C.S. *Mathematics for Machine Learning*; Cambridge University Press: Cambridge, UK, 2020.
65. Li, J.; Zhou, P.; Xiong, C.; Hoi, S.C. Prototypical contrastive learning of unsupervised representations. *arXiv* **2020**, arXiv:2005.04966.
66. Kay, W.; Carreira, J.; Simonyan, K.; Zhang, B.; Hillier, C.; Vijayanarasimhan, S.; Viola, F.; Green, T.; Back, T.; Natsev, P.; et al. The kinetics human action video dataset. *arXiv* **2017**, arXiv:1705.06950.
67. Goyal, R.; Ebrahimi Kahou, S.; Michalski, V.; Materzynska, J.; Westphal, S.; Kim, H.; Haenel, V.; Fruend, I.; Yianilos, P.; Mueller-Freitag, M.; et al. The "something something" video database for learning and evaluating visual common sense. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5842–5850.
68. Wang, S.; Zhang, J.; Wang, P.; Law, J.; Calinescu, R.; Mihaylova, L. A deep learning-enhanced Digital Twin framework for improving safety and reliability in human–robot collaborative manufacturing. *Robot. Comput. Integr. Manuf.* **2024**, *85*, 102608.
69. Han, S.; Pool, J.; Tran, J.; Dally, W. Learning both weights and connections for efficient neural network. In Proceedings of the 29th International Conference on Neural Information Processing Systems (NIPS'15), Montreal, QC, Canada, 7–12 December 2015; pp. 1135–1143.
70. Li, K.; Wang, Y.; Zhang, J.; Gao, P.; Song, G.; Liu, Y.; Li, H.; Qiao, Y. Uniformer: Unifying convolution and self-attention for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 12581–12600.
71. Wang, M.; Xing, J.; Liu, Y. ActionCLIP: A New Paradigm for Video Action Recognition. *arXiv* **2021**, arXiv:2109.08472.
72. Jeff R. Experiment Tracking with Weights & Biases. 2020. Available online: https://wandb.ai/site/ (accessed on 23 April 2025).
73. Feichtenhofer, C.; Fan, H.; Malik, J.; He, K. SlowFast Networks for Video Recognition. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6201–6210. https://doi.org/10.1109/ICCV.2019.00630.
74. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning Transferable Visual Models From Natural Language Supervision. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021.
75. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.