Dynamic Network Quantization for Efficient Video Inference

Ximeng Sun[1] Rameswar Panda[2] Chun-Fu (Richard) Chen[2]
Aude Oliva[2,3] Rogerio Feris[2] Kate Saenko[1,2]
[1]Boston University, [2]MIT-IBM Watson AI Lab, [3]MIT

Abstract

Deep convolutional networks have recently achieved great success in video recognition, yet their practical realization remains a challenge due to the large amount of computational resources required to achieve robust recognition. Motivated by the effectiveness of quantization for boosting efficiency, in this paper, we propose a dynamic network quantization framework, that selects optimal precision for each frame conditioned on the input for efficient video recognition. Specifically, given a video clip, we train a very lightweight network in parallel with the recognition network, to produce a dynamic policy indicating which numerical precision to be used per frame in recognizing videos. We train both networks effectively using standard backpropagation with a loss to achieve both competitive performance and resource efficiency required for video recognition. Extensive experiments on four challenging diverse benchmark datasets demonstrate that our proposed approach provides significant savings in computation and memory usage while outperforming the existing state-of-the-art methods. Project page: https://cs-people.bu.edu/sunxm/VideoIQ/project.html.

1. Introduction

With the availability of large-scale video datasets [5, 36], deep learning models based on 2D/3D convolutional neural networks (CNNs) [6, 52, 48, 28, 17] have dominated the field of video recognition. However, despite impressive performance on standard benchmarks, efficiency remains a great challenge for many resource constrained applications due to the heavy computational burden of deep CNN models.

Motivated by the need of efficiency, existing research efforts mainly focus on either designing compact models [41, 49, 11] or sampling of salient frames for efficient recognition [60, 57, 34]. While these methods have shown promising results, they all use 32-bit precision for processing all the frames in a given video, limiting their achievable efficiency. Specifically, orthogonal to the network design, the computational cost of a CNN is directly affected by the

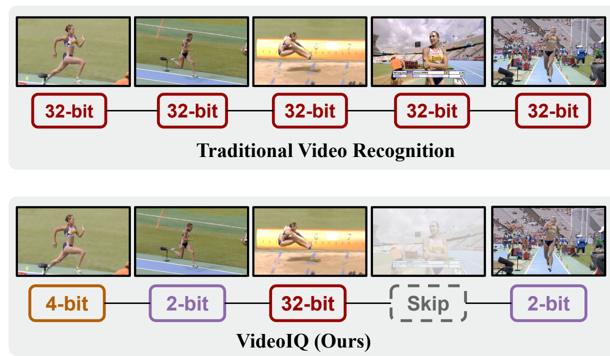

Figure 1: **A conceptual overview of our approach.** Instead of processing all the video frames with the same 32-bit precision, `VideoIQ` learns to dynamically select optimal quantization precision conditioned on input clips for efficient video recognition. It is computationally very efficient to process more informative frames with high precision and less informative ones with lower precision, without sacrificing accuracy. Best viewed in color.

bit-width of weights and activations [16, 68, 8], which surprisingly as another degree of freedom for efficient video inference, is almost overlooked in previous works. To illustrate this, let us consider the video in Figure 1, represented by five uniformly sampled frames. A quick glance on the video clearly shows that only the third frame can be processed using 32-bit precision as this is the most informative frame for recognizing the action "Long Jump", while the rest can be processed at very low precision or even skipped (i.e., precision set to zero) without sacrificing the accuracy (Bottom), resulting in large computational savings compared to processing all frames with same 32-bit precision, as generally done in mainstream video recognition methods (Top).

Inspired by this observation, we introduce **Video Instance-aware Quantization** (`VideoIQ`), which for the first time advocates a novel input-dependent dynamic network quantization strategy for efficient video recognition. While dynamic network quantization looks trivial and handy at the first glance, we need to address two challenges: (1) how to efficiently determine what quantization precision to use per target instance; and (2) given instance-specific precisions, how can we flexibly quantize the weights and activations of a single deep recognition network into various precision levels, without additional storage or computation cost.

To address the aforementioned challenges, we propose a simple end-to-end differentiable approach to learn a decision policy that selects optimal precision conditioned on the input, while taking both accuracy and efficiency into account in recognizing complex actions. We achieve this by sampling the policy from a discrete distribution parameterized by the output of a lightweight policy network, which decides on-the-fly what precision should be used on a per frame basis. Since these decision functions are discrete and non-differentiable, we train the policy network using standard back-propagation through Gumbel Softmax sampling [24], without resorting to complex reinforcement learning, as in [60, 9, 63]. Moreover, instead of storing separate precision-specific models, we train a single deep neural network for action recognition using joint training, which enables us to directly adjust the numerical precision by simply truncating the least significant bits, without performance degradation. Our proposed approach provides not only high computational efficiency but also significant savings in memory–a practical requirement of many real-world applications which has been largely ignored by prior works [34, 59, 35, 60].

We conduct extensive experiments on four standard video recognition datasets (ActivityNet-v1.3 [3], FCVID [25], Mini-Sports1M [28] and Mini-Kinetics [5]) to demonstrate the superiority of our proposed approach over state-of-the-art methods. Our results show that **`VideoIQ`** can yield significant savings in computation and memory (e.g., average 26.0% less GFLOPS and 55.8% less memory), while achieving better recognition performance, over the most competitive SOTA baseline [34]. We also discover that the decision policies learned using our method are transferable to unseen classes and videos across different datasets. Furthermore, qualitative results suggest that our learned policies correlate with the distinct visual patterns in video frames, i.e., our method utilizes 32-bit full precision only for relevant video frames and process non-informative frames at low precision or skip them for computation efficiency.

2. Related Work

Video Recognition. Much progress has been made in developing a variety of ways to recognize videos, by either applying 2D-CNNs [28, 52, 45, 46] or 3D-CNNs [48, 5, 17]. Despite promising results, there is a significant interest in developing more efficient models with reasonable performance [41, 49]. SlowFast network [12] employs two pathways for recognizing actions by processing a video at both slow and fast frame rates. Many works utilize 2D-CNNs for efficient recognition by modeling temporal causality using different aggregation modules [52, 67, 10, 32]. Expansion of 2D architectures across frame rate, spatial resolution, network width, is proposed in [11]. While these approaches bring reasonable efficiency improvements, all of them process the video frames using same 32-bit precision, regardless of information content in each input frame, which varies in most real-world long videos. In contrast, our approach dynamically selects bit-width per input, to strategically allocate computation at test time for efficient recognition.

Dynamic Computation. Dynamic computation to improve efficiency has been studied from multiple perspectives [1, 2, 50, 53, 15, 37, 13, 33]. Representative methods for image classification, dynamically adjust network depth [13, 33, 58, 21, 62], width [65, 7, 20], perform routing [26, 33] or switch resolutions [61]. Similar in spirit, dynamic methods for efficient video recognition adaptively select salient frames/clips [63, 60, 30, 9, 57, 23], utilize audio [14], reduce feature redundancy [38], or select frame resolutions [59, 34]. Recently, AdaFuse [35] proposes adaptive fusion of channels from current and past feature maps on a per instance basis, for recognizing video actions. Our approach is closely related yet orthogonal to these approaches as it focuses on network quantization to dynamically select the optimal bit-width conditioned on inputs, in pursuit of computational efficiency without sacrificing accuracy. Moreover, unlike existing works, our framework requires neither complex RL policy gradients [60, 57, 63] nor additional modalities such as audio [14, 30] to learn dynamic policies.

Network Quantization. Low-precision networks [16, 68, 8], have attracted intense attention in recent years. Early works such as [16, 31, 68] mainly focus on quantizing weights while using 32-bit activations. Recent approaches quantize both weights and activations through using uniform quantization that uses identical bit-width for all layers [66, 8, 39], or mixed precision quantization that uses different bit-widths for different layers or even channels [51, 4, 56]. Binary networks [22, 42] constrain both weights and activations to binary values, which brings great benefits to specialized hardware devices. Designing efficient strategies for training low-precision [70, 29, 69] or any-precision networks [27, 64] that can flexibly adjust the precision during inference is also another recent trend in quantization. Despite recent progress, the problem of quantization for video recognition models is rarely explored. Moreover, existing methods perform quantization in a static manner with a fixed computational cost, leaving adaptive quantization conditioned on inputs an open problem.

3. Proposed Method

Given T sampled frames from a video $V = \{x_1, x_2, \cdots, x_T\}$ with the action label y and a set of n candidate bit-widths (precisions) $\mathcal{B} = \{b_1, b_2, \cdots, b_n\}$ (assuming $b_1 > b_2 > \cdots > b_n$), our goal is to seek (1) a policy function $g : V \rightarrow \mathcal{B}^T$ that automatically decides the optimal

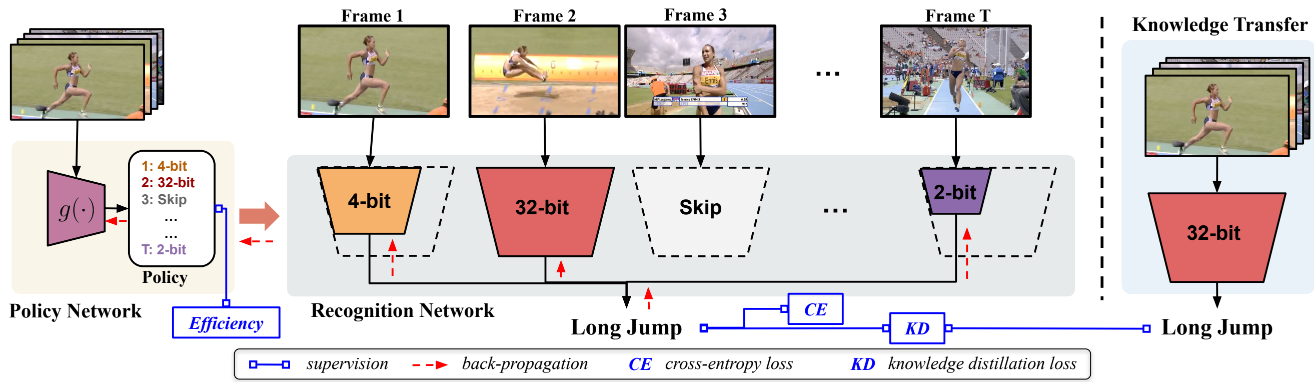

Figure 2: **Illustration of our proposed approach**. **`VideoIQ`** consists of a very lightweight policy network and a single backbone network for recognition which can be simply quantized to lower precisions by truncating the least significant bits. The policy network decides what quantization precision to use on a per frame basis, in pursuit of a reduced overall computational cost without sacrificing recognition accuracy. We train both networks using back-propagation with a combined loss of standard cross-entropy and efficiency for video recognition. We additionally distill knowledge from a pre-trained full-precision model to guide the training of lower precisions. During inference, each frame is sequentially fed into the policy network to select optimal precision for processing the current frame through the recognition network and then the network averages all the frame-level predictions to obtain the video-level prediction. Best viewed in color.

bit-width b for the frame x_i for processing in the recognition network, (2) a single recognition network $f : V \rightarrow y$ which can be quantized to different precisions in $\mathcal{B}$ without additional storage or computation cost. With the desired policy network g and recognition network f, our main objective is to improve accuracy, while taking the resource efficiency into account for video action recognition. Note that given the optimal bit-width b for the frame x_i, we quantize all the network weights and activations to the same bit-width b, which is well supported by existing hardwares.

3.1. Preliminaries

We denote the full-precision network weights by $\mathbf{W}$ and activations by $\mathbf{A}$. Given a certain precision with bit-width b and a quantization function Q, we denote the quantization of $\mathbf{W}$ and $\mathbf{A}$ as $Q(\mathbf{W}, b) = \widehat{W}_b$ and $Q(\mathbf{A}, b) = \widehat{A}_b$. In this paper, we use DoReFa [68] for weight quantization and PACT [8] for activation quantization.

Weight Quantization. DoReFa [68] normalizes $\mathbf{W}$ into $[-1, 1]$ and then rounds it to the nearest quantization levels:

$$\widehat{W}_b = 2 \times \text{quantize}_b\left(\frac{tanh(\mathbf{W})}{2\max tanh(\mathbf{W})} + \frac{1}{2}\right) - 1, \quad (1)$$

$$\text{quantize}_b(x) = \frac{1}{2^b - 1} \times \lfloor (2^b - 1)x \rceil, \quad (2)$$

where $\lfloor . \rceil$ is the rounding operation.

Activation Quantization. PACT [8] introduces a learnable clipping value α for activations in each layer. More specifically, the activation $\mathbf{A}$ is first clipped into $[0, \alpha]$ and then rounded to the nearest quantization levels:

$$\widehat{A}_b = \alpha \times \text{quantize}_b(\text{clip}(A, 0, \alpha)/\alpha).$$

3.2. Approach Overview

Figure 2 shows an overview of our approach. In general, we learn a instance-specific policy a_i that decides on-the-fly which precision to use (or even skip) for processing the current frame x_i, and a video classifier f which can be flexibly quantized to the desired precision of the current frame by simply truncating the least significant bits without any extra computation or memory cost. To this end, **`VideoIQ`** consists of a lightweight policy network g and a video recognition network f. The policy network g contains a feature extractor and an LSTM module to learn the discrete decisions of which precision to use, per input frame (see Section 3.3). Moreover, it is often unnecessary and inefficient to process every frame in a video due to large redundancy resulting from static scenes or frame quality being very low. Thus, we skip frames (i.e., precision set to zero) in addition to dynamic selection of precisions in an unified framework to improve efficiency in video recognition. To further enable flexible and scalable quantization, we learn the video classifier as an any-precision network and design a simple yet effective optimization scheme to ensure that the single set of network weights get executed with multiple precisions without additional storage and computation cost (see Section 3.4).

During the training, we first learn the any-precision recognition network and then optimize the policy network with Gumbel-Softmax Sampling [24] through standard back-propagation. We design the loss to achieve both competitive performance and computational efficiency (measured by FLOPS [54]) required for video recognition. We additionally distill knowledge from a pre-trained full-precision model to guide training of the lower precisions. During the inference, each video frame is sequentially fed into the policy network whose output decides the right precision to use for the given frame and then the frame is processed through the recognition network with the predicted precision to generate a frame-level prediction. Finally, the network averages predictions of all the frames as the final video-level prediction. It is worth noting that the policy network is designed to be very lightweight so that its computational overhead is negligible (e.g., MobileNetv2 [43] in our work).

3.3. Learning Dynamic Quantization Policy

`VideoIQ` learns the frame-wise policy a_i to decide which precision to process the frame x_i or directly skip it where skipping can be viewed as processing the frame with 0-bit. So our entire action space is $\Omega = \mathcal{B} \cup \{0\}$. We generate decision $a_i \in \Omega, \forall i \in [1, T]$ from the policy network g sequentially. We compose the policy network with a feature extractor ϕ followed by an LSTM module:

$$h_i, o_i = \text{LSTM}(\phi(x_i), h_{i-1}, o_{i-1}), \tag{3}$$

where h_i and o_i are hidden state and outputs of LSTM at the time step i. We further compute the distribution $\pi_i \in \mathbb{R}^{|\Omega|}$ over our action space Ω from h_i:

$$\pi_i = \text{Softmax}(fc(h_i)). \tag{4}$$

However, sampling policy a_i from the discrete distribution π_i is non-differentiable which makes direct optimization difficult. One way to solve this is to model the optimization problem as a reinforcement learning problem and then derive the optimal parameters of the policy network using policy gradient methods [55]. However, policy gradient is often complex, unwieldy to train and requires techniques to reduce variance during training as well as carefully selected reward functions. In contrast, we use Gumbel-Softmax Sampling [24] to circumvent this non-differentiability and make our framework fully differentiable, as in [59, 47].

Gumbel-Softmax Sampling. The Gumbel Softmax trick [24] substitutes the original non-differentiable sample from a discrete distribution with a differentiable sample from a corresponding Gumbel-Softmax distribution.

Specifically, instead of directly sampling a_i from its distribution π_i, we generate it as,

$$a_i = \arg\max_{j \in \Omega} \big(\log \pi_i(j) + G_i(j)\big), \tag{5}$$

where $G_i = -\log(-\log U_i)$ is a standard Gumbel distribution with U_i sampled from a uniform distribution $\text{Unif}(0, 1)$. To remove the non-differentiable argmax operation in Eq. 5, the Gumbel Softmax trick relaxes one-hot$(a_i) \in \{0, 1\}^{|\Omega|}$ (the one-hot encoding of a_i) to $p_i \in \mathbb{R}^{|\Omega|}$ with the reparameterization trick [24]:

$$p_i(j) = \frac{\exp\big((\log \pi_i(j) + G_i(j))/\tau\big)}{\sum_{k \in \Omega} \exp\big((\log \pi_i(k) + G_i(k))/\tau\big)}, \tag{6}$$

where $j \in \Omega$ and τ is the temperature of the softmax. Clearly, when $\tau > 0$, the Gumbel-Softmax distribution p_i is smooth so π_i can be directly optimized by gradient descent, and when τ approaches 0, the soft decision p_i becomes the same as one-hot(a_i). Following [15, 47], we set $\tau = 5$ as the initial value and gradually anneal it down to 0 during training.

3.4. Any-Precision Video Recognition

Given frame-specific precisions, quantizing weights and activations of a single network while recognizing videos is a major challenge. A naive strategy is to manually train different models tailored for the different precision and then route frames to the corresponding models to generate predictions. However such a strategy requires time-consuming training for each of the models and also increases the memory storage cost, making it inefficient for many real-time applications. To tackle this problem, we adopt any-precision recognition [27, 64] that makes a single model be flexible to any numerical precision during the inference. Specifically, we first modify the weight quantizer to enable the network parameters to get quantized to lower precision with low computation cost after the training. Then, we propose a simple and effective learning scheme for training of the any-precision video recognition network.

With the original DoReFa quantization [68] (Eq. 1 and 2), all numerical precisions need to be quantized down from the full-precision value. Thus, the repeated weight quantizations cause redundant computation when the recognition network frequently switches across different precisions. To reduce computational cost of switching operation, we quantize full precision weight W to the largest bit-width b_1 and then truncate least significant $b_1 - b$ bits to get quantized weight $\widehat{W}_b$. We save the quantized b_1-bit network weights after the training. Benefiting from this modified quantization, we only need to discard the extra bits to switch to lower precisions during inference. Furthermore, we align $\mathbb{E}[\widehat{W}_b]$ with $\mathbb{E}[\widehat{W}_{b_1}]$ to minimize the mean discrepancy caused by discarded bits.

Inspired by [65, 27], we jointly train a single network under different bit-widths with shared weights for any-precision video recognition. Specifically, we gather losses of all precisions with same input batch and then update the network. To get the loss of a precision with bit-width b, we feed the input video and quantize network weights and activations to b-bit for every frame. To resolve mismatch in statistics of activations with different precisions, we use a separate set of Batch Normalization layers and clipping level parameters for different precisions [65]. Moreover, following the success of knowledge distillation [19], we transfer knowledge from a pretrained full-precision recognition network to guide training of lower precisions because the full-precision weights is expected to give confident predictions, and provide valuable knowledge in its soft logits, while the low-precision student gains the knowledge by mimicking the teacher.

3.5. Losses

For video action recognition, we minimize standard cross-entropy loss between predicted label and ground truth action:

$$\mathcal{L}_{ce}(V|A) = \mathbb{E}[-y \log(f(V|A))], \tag{7}$$

where $A = a_1, a_2, \cdots, a_T$ represents precisions to use for the sampled T frames, which can be either predicted by the lightweight policy network ($A = g(V)$) or set manually.

To better guide the optimization of the model with lower capacity, *e.g.* the recognition network with lower precision, we utilize a distillation loss $\mathcal{L}_{kd}$ to transfer knowledge from a pretrained full-precision video recognition network (teacher) by taking Kullback–Leibler (KL) divergence between soft-logits of our model y_A and of the teacher network y_t as

$$\mathcal{L}_{kd}(V|A) = \mathrm{KL}(y_t||y_A) = \sum_{i=1}^{m}(y_t)_i \log\frac{(y_t)_i}{(y_A)_i}, \tag{8}$$

where m is the number of video categories and $(\cdot)_i$ denotes the i-th element of the vector. Thus, given the input video V, the overall loss $\mathcal{L}_f$ to optimize the any-precision video recognition network f is defined as

$$\mathcal{L}_f(V) = \sum_{A=b_1^T,\cdots b_n^T} \mathcal{L}_{ce}(V|A) + \mathcal{L}_{kd}(V|A). \tag{9}$$

To address computational efficiency, we pre-compute FLOPs [54] needed for one frame to get processed in the recognition network with different candidate precisions in B. We directly minimize FLOPs usage per video with the generated policy A, to reduce the computational cost as

$$\mathcal{L}_e(A) = \sum_{i=1}^{T}(\mathrm{FLOP}(a_i)). \tag{10}$$

Furthermore, we introduce two additional regularizers to better optimize the policy network. First, we enforce a balanced policy usage over the entire action space to avoid the policy network learning some sub-optimal solutions where some actions are totally ignored. More formally, we define the balanced policy usage loss $\mathcal{L}_b$ as

$$\mathcal{L}_b(A) = \sum_{k\in\Omega}(\mathbb{E}[\frac{1}{T}\sum_{i=1}^{T}\mathbb{1}(a_i = k)] - \frac{1}{|\Omega|}). \tag{11}$$

Second, we minimize the entropy of the learned probability distribution over the action space Ω of each frame. It forces the policy network to avoid randomness during the inference by generating deterministic prediction for the precision to use for each video frame:

$$\mathcal{L}_d(\pi) = \sum_{i=1}^{T} H(\pi_i), \tag{12}$$

where $H(\cdot)$ is the entropy function. Finally, the overall loss $\mathcal{L}_g$ to optimize the policy network g is defined as

$$\begin{aligned}\mathcal{L}_g(V) = {} & \mathcal{L}_{ce}(V|A) + \mathcal{L}_{kd}(V|A) \\ & + w_1\mathcal{L}_e(A) + w_2\mathcal{L}_b(A) + w_3\mathcal{L}_d(\pi),\end{aligned} \tag{13}$$

where $A = g(V)$, and w_1, w_2 and w_3 are hyperparameters to balance loss terms. In summary, we first jointly train the any-precision recognition network f with all precisions in $\mathcal{B}$ (using Eq. 9), and then train policy network g (using Eq. 13) to generate policy over the action space Ω per input frame.

4. Experiments

4.1. Experimental Setup

Datasets. We evaluate our approach using four datasets, namely ActivityNet-v1.3 [3], FCVID [25], Mini-Sports1M [28] and Mini-Kinetics [5]. ActivityNet contains $10,024$ videos for training and $4,926$ videos for validation across 200 categories. FCVID consists of $45,611$ videos for training and $45,612$ videos for testing across 239 classes. Mini-Sports1M [14] is a subset of full Sports1M dataset [28] containing 30 videos per class in training and 10 videos per class in testing over 487 classes. Mini-Kinetics [6] is a subset of full Kinetics400 [5] dataset containing $121,215$ videos for training and $9,867$ videos for testing across 200 classes.

Implementation Details. We adopt temporal segment network (TSN) [52] to aggregate the predictions over $T = 16$ uniformly sampled frames from each video. We use ResNet-18 and ResNet-50 [18] for the recognition network while MobileNetv2 [43] combined with a single-layer LSTM (with 512 hidden units) to serve as policy network in all our experiments. To save computation, we use lower resolution images (84×84) in policy network. We set the action space $\Omega = \{32, 4, 2, 0\}$ in all experiments, i.e., the policy network can choose either one out of $\{32, 4, 2\}$ precision or skip frame for efficient recognition. We first train the any-precision recognition network (pretrained from ImageNet weights) for 100 epochs to provide a good starting point for policy learning and then train the policy network for 50 epochs on all datasets. We use separate sets of learning parameters (learning rate, weight decay) for clipping values of each precision. Following [68, 8], we do not quantize input, first layer and last layer of the network. More implementation details are included in the supplementary material.

Baselines. We compare our approach with the following baselines and existing approaches. First, we consider a 2D-CNN based "Uniform" baseline that uses 32-bit precision to process all the sampled frames and then averages the frame-level results as the video-level prediction. We also compare with two more variants of uniform baseline that uses lower precisions such as 4-bit and 2-bit respectively to process the video frames. Second, we compare with "Ensemble" baseline that gathers all the frame-level predictions by processing

Model	ActivityNet		FCVID		Mem.
	mAP (%)	GFLOPs	mAP (%)	GFLOPs	(MB)
			ResNet-18		
Uniform (32-bit)	69.7	29.1	77.6	29.1	43.1
Uniform (4-bit)	68.0	7.3	76.5	7.3	5.4
Uniform (2-bit)	65.2	1.8	74.3	1.8	2.7
Ensemble	70.7	38.2	78.8	38.2	51.2
VideoIQ	70.9	9.5	79.1	9.4	50.2
			ResNet-50		
Uniform (32-bit)	72.5	65.8	81.0	65.8	91.4
Uniform (4-bit)	71.7	16.5	79.3	16.5	11.4
Uniform (2-bit)	69.3	4.1	78.5	4.1	5.7
Ensemble	74.7	86.4	83.0	86.4	108.5
VideoIQ	74.8	28.1	82.7	27.0	98.6

Table 1: **Video recognition results on ActivityNet and FCVID**. Our approach **VideoIQ** outperforms all the simple baselines.

them at different precision (instead of selecting an optimal precision per frame). This serves as a very strong baseline for classification, at the cost of heavy computation. Finally, we compare our method with existing efficient video recognition approaches, including LiteEval [59] (NeurIPS'19), SCSampler [30] (ICCV'19), AR-Net [34] (ECCV'20), and AdaFuse [35] (ICLR'21). We directly quote the numbers reported in the published papers when possible or use authors provided source codes [59, 35] using the same backbone and experimental settings for a fair comparison.

Metrics. We compute either mAP (mean average precision) or Top-1 accuracy depending on datasets to measure performance of different methods. We follow [54, 40, 44] and measure computational cost with giga floating-point operations (GFLOPs), which is a hardware independent metric. Specifically, given FLOPs of a full-precision layer by $\mathbf{a}$, the FLOPs of m-bit weight and n-bit activation quantized layer is $\frac{mn}{64} \times \mathbf{a}$. We also measure memory usage (MB) represented by the storage for parameters of the network, as in [54].

4.2. Results and Analysis

Comparison with Traditional Uniform Baselines. We first compare **VideoIQ** using different backbones (ResNet-18 and ResNet-50) to show how much performance our dynamic approach **VideoIQ** can achieve compared to simple 2D-CNN based baselines on both ActivityNet and FCVID datasets. As shown in Table 1, our approach consistently outperforms the full-precision uniform baseline (32-bit) in both mAP and GFLOPS, with minimal increase in memory on both datasets. Using ResNet-18 as the backbone, **VideoIQ** obtains an mAP of 70.9% and 79.1%, requiring 9.5 and 9.4 GFLOPS on ActivityNet and FCVID respectively. Uniform quantization with low bit-widths leads to a significant reduction in computation and memory but they suffer from a noticeable degradation in recognition performance, e.g., the 2-bit performance is 4.5% and 3.3% lower than the 32-bit counterpart on ActivityNet and FCVID respectively.

Similarly, with ResNet-50, **VideoIQ** offers 56.7% (65.8 vs 28.1) and 58.9% (65.8 vs 27.0) savings in GFLOPS while outperforming the Uniform (32-bit) baseline by 2.1% and 2.7% in mAP on ActivityNet and FCVID, respectively. We further compare with 8-bit Uniform Baseline that uses same percentage of random skipping as **VideoIQ** (i.e. 8% random skipping on ActivityNet). With ResNet-50, our approach outperforms this baseline by 2.7% (72.1% vs 74.8%), showing effectiveness of learned policy in selecting optimal quantization precision per frame while recognizing videos.

As shown in Table 1, Ensemble achieves comparable recognition performance because it is a very strong baseline that gathers all the predictions by processing frames through multiple backbones. However, **VideoIQ** provides 67.4% and 68.7% computational savings including a 10% savings in memory over the Ensemble baseline on ActivityNet and FCVID respectively, showing the importance of instance-aware dynamic quantization for efficient video recognition. Moreover, we also compare with a Weighted Ensemble baseline, where weights are assigned based on entropy of softmax scores to reflect prediction confidence of different predictions. We observe that it only achieves 0.3% higher mAP while requiring 67.4% more computation than our method on ActivityNet (75.1% vs 74.8%). Note that **VideoIQ** requires less computation on average on FCVID than ActivityNet as FCVID contains more static videos with high redundancy compared to ActivityNet that consists of action-centric videos with rich temporal information.

Comparison with State-of-the-Art Methods. Tables 2-3 summarize the results and comparisons with existing dynamic inference methods on all four datasets. Our approach is clearly better than all the compared methods in terms of both accuracy and resource efficiency (computation and memory), making it suitable for efficient video recognition.

Model	ActivityNet		FCVID		Mem.
	mAP (%)	GFLOPs	mAP (%)	GFLOPs	(MB)
LiteEval	72.7	95.1	80.0	94.3	177.2
SCSampler	72.9	42.0	81.0	42.0	98.6
AR-Net	73.8	33.5	81.3	35.1	223.4
AdaFuse	73.1	61.4	81.6	45.0	151.2
VideoIQ	**74.8**	**28.1**	**82.7**	**27.0**	**98.6**

Table 2: **Comparison with state-of-the-art methods on ActivityNet and FCVID**. **VideoIQ** achieves the best mAP while offering significant savings in both GFLOPS and Memory (MB).

Model	Mini-Sports1M		Mini-Kinetics		Mem.
	mAP (%)	GFLOPs	Tops-1 (%)	GFLOPs	(MB)
LiteEval	44.7	66.2	61.0	99.0	177.2
SCSampler	44.3	42.0	70.8	42.0	98.6
AR-Net	45.0	37.6	71.7	32.0	223.4
AdaFuse	44.1	60.3	72.3	23.0	151.2
VideoIQ	**46.4**	**26.8**	**72.3**	**20.4**	**98.6**

Table 3: **Comparison with state-of-the-art methods on Mini-Sports1M and Mini-Kinetics**. Our approach **VideoIQ** (w/ ResNet-50) obtains the best performance with great savings in computation (GFLOPS) and memory (MB).

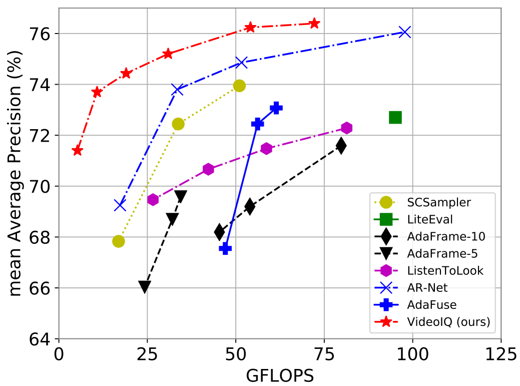

Figure 3: **Computational cost (GFLOPS) vs mean Average Precision (%) on ActivityNet dataset**. **VideoIQ** (red points) achieves the best trade-off when compared to existing methods.

VideoIQ obtains an mAP (accuracy for Mini-Kinetics) of 74.8%, 82.7%, 46.4% and 72.3%, while requiring 28.1, 27.0, 26.8 and 20.4 GFLOPs on ActivityNet, FCVID, Mini-Sports1M and Mini-Kinetics, respectively. Note that while most of the compared methods reduce computation at the cost of significant increase in memory, our approach improves computational efficiency by using a model whose memory size is just slightly larger than the 32-bit model.

Among the compared methods, AR-Net is the most competitive in terms of computational efficiency. However, **VideoIQ** consistently outperforms AR-Net in recognition performance while providing 26.0% savings on average in computation and 55.8% savings in memory. This is because of our two introduced components working in concert: dynamic quantization for computational efficiency and use of a single any-precision recognition network instead of separate models for memory efficiency. Likewise when compared with the recent method AdaFuse, our approach offers an average 41.1% and 34.7% reduction in computation and storage memory while improving the recognition performance (maximum 2.3% on Mini-Sports1M) across all the datasets. AdaFuse obtains the best performance compared to other existing methods on Mini-Kinetics but it fails to achieve similar performance on untrimmed video datasets. We suspect that being a method that relies on efficient reuse of history feature maps, it fails to aggregate the information of all time stamps when the video gets very long, as in untrimmed datasets. In summary, **VideoIQ** establishes new state-of-the-art for the task of efficient video recognition on four datasets, improving previous best result in terms of accuracy, computational efficiency and memory efficiency.

Figure 3 compares our approach to the existing methods by varying computational budgets on ActivityNet. Our method consistently outperforms all the compared methods and achieves the best trade-off between computational cost and accuracy, which once again shows that **VideoIQ** is an effective and efficient design for video recognition.

Train \ Test	ActivityNet	FCVID	Mini-Sports1M	Mini-Kinetics
ActivityNet	**74.8**	82.7	46.3	71.6
FCVID	74.4	**82.8**	45.8	72.1
Mini-Sports1M	74.6	82.6	**46.4**	72.2
Mini-Kinetics	74.7	82.7	46.3	**72.3**

Table 4: **Transferring learned policies**. Diagonal numbers refer to training and testing the quantization policy on the same dataset while non-diagonal numbers refer to learning the policy on one dataset (rows) and testing on others (columns).

Transferring Learned Policies. We analyze transferability of our learned policy by performing cross-dataset experiments, i.e., learning policy on one dataset while testing on the other. Specifically, we take the policy network trained on one dataset and utilize it directly for testing along with a trained any-precision recognition network on another dataset. Table 4 summarizes the results. As expected, training and testing on the same dataset provides the best performance on all cases (marked in blue). However, the negligible difference among the values across each column clearly shows that policies learned using our method are transferable to unseen classes and videos across different datasets.

Qualitative Analysis. To better understand the learned policy, we visualize selected precision per input frame in Figure 4. Videos are uniformly sampled in 8 frames. Overall, our approach **VideoIQ** focuses on the right quantization precision to use per frame for correctly classifying videos while taking efficiency into account. **VideoIQ** processes the most indicative frames in 32-bit precision while it uses lower precision (or skips) for frames that irrelevant to the action (e.g., “Playing saxophone” and “Snow Tubing”). Similarly in the case of “Playing violin” and “Mixing drinks”, after being confident about the prediction, it interestingly avoids using the 32-bit precision even if informative content appear later in the video. More qualitative examples are included in the supplementary material.

Figure 5 shows the overall policy distribution on different datasets. Our approach leads to distinctive policy patterns representing different characteristics of datasets. For example, while only few frames on ActivityNet use 2-bit precision, about 30% of the frames on the other datasets can be processed using 2-bit precision, leading to different amount of computational savings across datasets. **VideoIQ** skips very few frames on Mini-Kinetics (2%), which is because Mini-Kinetics dataset contains short trimmed videos ($6-10$ seconds) while the remaining datasets consists of long untrimmed videos, lasting up to 5 minutes.

4.3. Ablation Studies

We present the following ablation experiments using ResNet-50 on ActivityNet dataset to show the effectiveness of different components in our proposed method.

Effect of Different Losses. Table 5 summarizes the effect

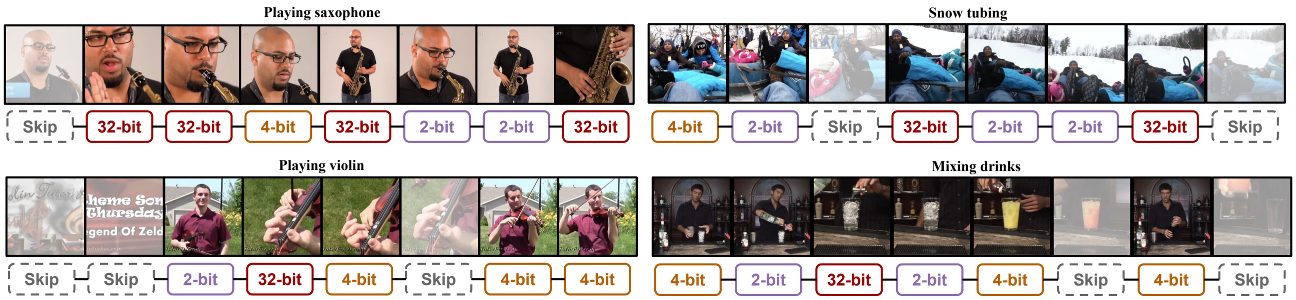

Figure 4: **Qualitative examples from ActivityNet dataset.** Our approach `VideoIQ` processes more informative frames with high precision and less informative ones with lower precision or skip them when irrelevant, for efficient video recognition. Best viewed in color.

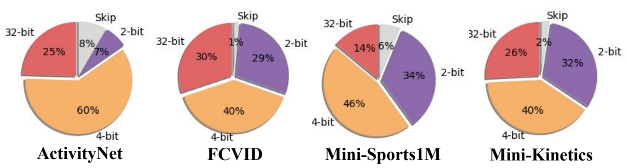

Figure 5: **Dataset-specific policy distribution**.

L_{ce}	L_{kd}	L_e	L_b	L_d	mAP (%)	GFLOPs
✓		✓	✓	✓	73.5	29.0
✓	✓				**75.1**	56.4
✓	✓	✓			74.5	34.6
✓	✓	✓	✓		74.3	32.0
✓	✓	✓	✓	✓	74.8	**28.1**

Table 5: **Effect of different losses on ActivityNet**.

of different losses on ActivityNet. Training without knowledge transfer from the 32-bit model (top row: by turning off $\mathcal{L}_{kd}$) only obtains a mAP of 73.5% with similar GFLOPS as ours, which shows that it is important to utilize soft targets of the full-precision model as the teacher to guide lower precisions in learning. As expected, training by setting $\mathcal{L}_e$ to 0 achieves the highest mAP of 75.1% while requiring 38.5% more GFLOPS compared to the one that uses efficient loss in training (2^{nd} vs 3^{rd} row). Finally, adding both regularizations ($\mathcal{L}_b$ and $\mathcal{L}_d$) during the policy learning leads to the best performance with least computation showing the effectiveness of different losses in our framework.

Effect of Decision Space. We investigate the effect of decision space Ω by using different combinations of precision and skipping. As shown in Table 6, only skipping frames (i.e., $\Omega = \{32, 0\}$) leads to an mAP of 72.9% while setting the decision space to choose only precisions (i.e., $\Omega = \{32, 4, 2\}$) leads to an mAP of 74.5% on ActivityNet. Compared to all the alternatives, the best strategy is to combine the set of precisions with skipping by setting $\Omega = \{32, 4, 2, 0\}$ for achieving top performance of 74.8% in mAP with 28.1 GFLOPS on ActivityNet dataset.

Comparison with Random Policy. We compare with random policy that uses the same backbone framework but randomly samples policy actions from uniform distribution

Decision Space Ω	mAP (%)	GFLOPs
{32, 0}	72.9	31.6
{32, 4, 2}	74.5	31.4
{32, 4, 0}	74.7	32.8
{32, 2, 0}	74.0	31.2
{32, 4, 2, 0}	**74.8**	**28.1**

Table 6: **Effect of different decision space on ActivityNet**. Note that 0 indicates skipping the frame for processing by the classifier.

and observe that our approach outperforms it by 2% in mAP (72.8% vs 74.8%) on ActivityNet, which demonstrates effectiveness of learned policy in selecting optimal quantization precision per frame while recognizing videos. We also observe similar improvements ($\sim 2\% - 3\%$) on other datasets.

Effectiveness of Any-Precision Recognition Network. We use three separate precision specific quantized models as part of the classifier and route frames to the corresponding models based on the policy to generate predictions. Our approach using separate models on ActivityNet (with ResNet-50) achieves an mAP of 74.9% (an improvement of only 0.1%) while requiring 34.0 GFLOPS and 115.6MB of memory, in contrast to 28.1 GFLOPS and 50.2MB of memory with a single any-precision network. Similarly, use of separate models on Mini-Sports1M yields only 0.1% improvement in mAP with 7.1% more computation and 56.5% of additional memory, compared to an any-precision network. This clearly shows the effectiveness of our any-precision network over individual quantized models in obtaining very competitive performance with less computation and memory.

5. Conclusion

In this paper, we introduce video instance-aware quantization that decides what precision should be used on a per frame basis for efficient video recognition. Specifically, we utilize a lightweight policy network to predict these decisions and train it in parallel with an any-precision recognition network with the goal of achieving both competitive accuracy and resource efficiency. Comprehensive experiments on four challenging and diverse datasets demonstrate the superiority of our approach over existing state-of-the-art methods.

References

[1] Emmanuel Bengio, Pierre-Luc Bacon, Joelle Pineau, and Doina Precup. Conditional computation in neural networks for faster models. *arXiv preprint arXiv:1511.06297*, 2015. 2

[2] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 2

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 961–970, 2015. 2, 5

[4] Zhaowei Cai and Nuno Vasconcelos. Rethinking differentiable search for mixed-precision neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2349–2358, 2020. 2

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 1, 2, 5

[6] Chun-Fu Chen, Rameswar Panda, Kandan Ramakrishnan, Rogerio Feris, John Cohn, Aude Oliva, and Quanfu Fan. Deep analysis of cnn-based spatio-temporal representations for action recognition. *arXiv preprint arXiv:2010.11757*, 2020. 1, 5

[7] Zhourong Chen, Yang Li, Samy Bengio, and Si Si. You look twice: Gaternet for dynamic filter selection in cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9172–9180, 2019. 2

[8] Jungwook Choi, Zhuo Wang, Swagath Venkataramani, Pierce I-Jen Chuang, Vijayalakshmi Srinivasan, and Kailash Gopalakrishnan. Pact: Parameterized clipping activation for quantized neural networks. *arXiv preprint arXiv:1805.06085*, 2018. 1, 2, 3, 5

[9] Hehe Fan, Zhongwen Xu, Linchao Zhu, Chenggang Yan, Jianjun Ge, and Yi Yang. Watching a small portion could be as good as watching all: Towards efficient video classification. In *IJCAI International Joint Conference on Artificial Intelligence*, 2018. 2

[10] Quanfu Fan, Chun-Fu Richard Chen, Hilde Kuehne, Marco Pistoia, and David Cox. More is less: Learning efficient video representations by big-little network and depthwise temporal aggregation. In *Advances in Neural Information Processing Systems*, pages 2261–2270, 2019. 2

[11] Christoph Feichtenhofer. X3d: Expanding architectures for efficient video recognition. *arXiv preprint arXiv:2004.04730*, 2020. 1, 2

[12] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019. 2

[13] Michael Figurnov, Maxwell D Collins, Yukun Zhu, Li Zhang, Jonathan Huang, Dmitry Vetrov, and Ruslan Salakhutdinov. Spatially adaptive computation time for residual networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1039–1048, 2017. 2

[14] Gao, Ruohan and Oh, Tae-Hyun, and Grauman, Kristen and Torresani, Lorenzo. Listen to look: Action recognition by previewing audio. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 5

[15] Yunhui Guo, Honghui Shi, Abhishek Kumar, Kristen Grauman, Tajana Rosing, and Rogerio Feris. Spottune: transfer learning through adaptive fine-tuning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4805–4814, 2019. 2, 4

[16] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. 1, 2

[17] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 6546–6555, 2018. 1, 2

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 4

[20] Weizhe Hua, Yuan Zhou, Christopher M De Sa, Zhiru Zhang, and G Edward Suh. Channel gating neural networks. In *Advances in Neural Information Processing Systems*, pages 1884–1894, 2019. 2

[21] Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens van der Maaten, and Kilian Q Weinberger. Multi-scale dense networks for resource efficient image classification. *arXiv preprint arXiv:1703.09844*, 2017. 2

[22] Itay Hubara, Matthieu Courbariaux, Daniel Soudry, Ran El-Yaniv, and Yoshua Bengio. Binarized neural networks. In *Advances in neural information processing systems*, pages 4107–4115, 2016. 2

[23] Noureldien Hussein, Mihir Jain, and Babak Ehteshami Bejnordi. Timegate: Conditional gating of segments in long-range activities. *arXiv preprint arXiv:2004.01808*, 2020. 2

[24] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*, 2017. 2, 3, 4

[25] Yu-Gang Jiang, Zuxuan Wu, Jun Wang, Xiangyang Xue, and Shih-Fu Chang. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *IEEE transactions on pattern analysis and machine intelligence*, 40(2):352–364, 2017. 2, 5

[26] Zequn Jie, Peng Sun, Xin Li, Jiashi Feng, and Wei Liu. Anytime recognition with routing convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 2

[27] Qing Jin, Linjie Yang, and Zhenyu Liao. Adabits: Neural network quantization with adaptive bit-widths. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2146–2156, 2020. 2, 4

[28] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video

classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1725–1732, 2014. 1, 2, 5

[29] Jangho Kim, Yash Bhalgat, Jinwon Lee, Chirag Patel, and Nojun Kwak. Qkd: Quantization-aware knowledge distillation. *arXiv preprint arXiv:1911.12491*, 2019. 2

[30] Bruno Korbar, Du Tran, and Lorenzo Torresani. Scsampler: Sampling salient clips from video for efficient action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6232–6242, 2019. 2, 6

[31] Cong Leng, Zesheng Dou, Hao Li, Shenghuo Zhu, and Rong Jin. Extremely low bit neural network: Squeeze the last bit out with admm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[32] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7083–7093, 2019. 2

[33] Mason McGill and Pietro Perona. Deciding how to decide: Dynamic routing in artificial neural networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2363–2372, 2017. 2

[34] Yue Meng, Chung-Ching Lin, Rameswar Panda, Prasanna Sattigeri, Leonid Karlinsky, Aude Oliva, Kate Saenko, and Rogerio Feris. Ar-net: Adaptive frame resolution for efficient action recognition. In *ECCV*, 2020. 1, 2, 6

[35] Yue Meng, Rameswar Panda, Chung-Ching Lin, Prasanna Sattigeri, Leonid Karlinsky, Kate Saenko, Aude Oliva, and Rogerio Feris. Adafuse: Adaptive temporal fusion network for efficient action recognition. In *International Conference on Learning Representations*, 2021. 2, 6

[36] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Tom Yan, Lisa Brown, Quanfu Fan, Dan Gutfruend, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *arXiv preprint arXiv:1801.03150*, 2018. 1

[37] Mahyar Najibi, Bharat Singh, and Larry S Davis. Autofocus: Efficient multi-scale inference. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9745–9755, 2019. 2

[38] Bowen Pan, Rameswar Panda, Camilo Luciano Fosco, Chung-Ching Lin, Alex J Andonian, Yue Meng, Kate Saenko, Aude Oliva, and Rogerio Feris. Va-red^2: Video adaptive redundancy reduction. In *International Conference on Learning Representations*, 2021. 2

[39] Eunhyeok Park, Junwhan Ahn, and Sungjoo Yoo. Weighted-entropy-based quantization for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5456–5464, 2017. 2

[40] Hai Phan, Zechun Liu, Dang Huynh, Marios Savvides, Kwang-Ting Cheng, and Zhiqiang Shen. Binarizing mobilenet via evolution-based searching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13420–13429, 2020. 6

[41] AJ Piergiovanni, Anelia Angelova, and Michael S Ryoo. Tiny video networks. *arXiv preprint arXiv:1910.06961*, 2019. 1, 2

[42] Mohammad Rastegari, Vicente Ordonez, Joseph Redmon, and Ali Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, pages 525–542. Springer, 2016. 2

[43] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018. 4, 5

[44] Mingzhu Shen, Feng Liang, Chuming Li, Chen Lin, Ming Sun, Junjie Yan, and Wanli Ouyang. Once quantized for all: Progressively searching for quantized efficient models. *arXiv preprint arXiv:2010.04354*, 2020. 6

[45] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Neural Information Processing System (NIPS)*, 2014. 2

[46] Swathikiran Sudhakaran, Sergio Escalera, and Oswald Lanz. Gate-shift networks for video action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1102–1111, 2020. 2

[47] Ximeng Sun, Rameswar Panda, Rogerio Feris, and Kate Saenko. Adashare: Learning what to share for efficient deep multi-task learning. *Advances in Neural Information Processing Systems*, 33, 2020. 4

[48] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015. 1, 2

[49] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5552–5561, 2019. 1, 2

[50] Andreas Veit and Serge Belongie. Convolutional networks with adaptive inference graphs. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–18, 2018. 2

[51] Kuan Wang, Zhijian Liu, Yujun Lin, Ji Lin, and Song Han. Haq: Hardware-aware automated quantization with mixed precision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8612–8620, 2019. 2

[52] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016. 1, 2, 5

[53] Xin Wang, Fisher Yu, Zi-Yi Dou, Trevor Darrell, and Joseph E Gonzalez. Skipnet: Learning dynamic routing in convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 409–424, 2018. 2

[54] Ziwei Wang, Jiwen Lu, Chenxin Tao, Jie Zhou, and Qi Tian. Learning channel-wise interactions for binary convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 568–577, 2019. 3, 5, 6

[55] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992. 4

[56] Bichen Wu, Yanghan Wang, Peizhao Zhang, Yuandong Tian, Peter Vajda, and Kurt Keutzer. Mixed precision quantization

of convnets via differentiable neural architecture search. *arXiv preprint arXiv:1812.00090*, 2018. 2

[57] Wenhao Wu, Dongliang He, Xiao Tan, Shifeng Chen, and Shilei Wen. Multi-agent reinforcement learning based frame sampling for effective untrimmed video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6222–6231, 2019. 1, 2

[58] Zuxuan Wu, Tushar Nagarajan, Abhishek Kumar, Steven Rennie, Larry S Davis, Kristen Grauman, and Rogerio Feris. Blockdrop: Dynamic inference paths in residual networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8817–8826, 2018. 2

[59] Zuxuan Wu, Caiming Xiong, Yu-Gang Jiang, and Larry S Davis. Liteeval: A coarse-to-fine framework for resource efficient video recognition. In *Advances in Neural Information Processing Systems*, pages 7778–7787, 2019. 2, 4, 6

[60] Zuxuan Wu, Caiming Xiong, Chih-Yao Ma, Richard Socher, and Larry S Davis. Adaframe: Adaptive frame selection for fast video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1278–1287, 2019. 1, 2

[61] Le Yang, Yizeng Han, Xi Chen, Shiji Song, Jifeng Dai, and Gao Huang. Resolution adaptive networks for efficient inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2369–2378, 2020. 2

[62] Taojiannan Yang, Sijie Zhu, Chen Chen, Shen Yan, Mi Zhang, and Andrew Willis. Mutualnet: Adaptive convnet via mutual learning from network width and resolution. In *European Conference on Computer Vision (ECCV)*, 2020. 2

[63] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2678–2687, 2016. 2

[64] Haichao Yu, Haoxiang Li, Honghui Shi, Thomas S Huang, and Gang Hua. Any-precision deep neural networks. *arXiv preprint arXiv:1911.07346*, 2019. 2, 4

[65] Jiahui Yu, Linjie Yang, Ning Xu, Jianchao Yang, and Thomas Huang. Slimmable neural networks. *arXiv preprint arXiv:1812.08928*, 2018. 2, 4

[66] Dongqing Zhang, Jiaolong Yang, Dongqiangzi Ye, and Gang Hua. Lq-nets: Learned quantization for highly accurate and compact deep neural networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 365–382, 2018. 2

[67] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018. 2

[68] Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. Dorefa-net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016. 1, 2, 3, 4, 5

[69] Bohan Zhuang, Lingqiao Liu, Mingkui Tan, Chunhua Shen, and Ian Reid. Training quantized neural networks with a full-precision auxiliary module. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1488–1497, 2020. 2

[70] Bohan Zhuang, Chunhua Shen, Mingkui Tan, Lingqiao Liu, and Ian Reid. Towards effective low-bitwidth convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7920–7928, 2018. 2