Adaptive Resolution for Efficient Action Recognition (Supplementary)

Table 1. Supplementary Material Overview

Section	Content
A	Details on Mini-Kinetics Dataset
B	GFLOPS Estimation
C	Policy Distributions
D	Qualitative Analysis
E	RL vs Gumbel Softmax in Policy Learning

A Mini-Kinetics

Kinetics is a large dataset containing 400 action classes and 240K training videos that are collected from YouTube. Since the full Kinetics dataset is quite large, we have created a smaller dataset that we call Mini-Kinetics by randomly selecting half of the categories of Kinetics-400 dataset. The mini-Kinetics dataset contains 121K videos for training and 10K videos for testing, with each video lasting 6-10 seconds. We will make the splits publicly available to enable future comparisons.

B GFLOPS Estimation

Table 2. GFLOPS for different backbones and resolutions

Network	Resolution	GFLOPS	Feature Dim
MobileNet−v2	84×84	0.0529	1280
ResNet−18	112×112	0.4683	512
ResNet−34	168×168	2.2490	512
ResNet−50	224×224	4.1103	2048
EfficientNet−b0	112×112	0.0975	1280
EfficientNet−b1	168×168	0.3937	1280
EfficientNet−b3	224×224	1.8000	1536

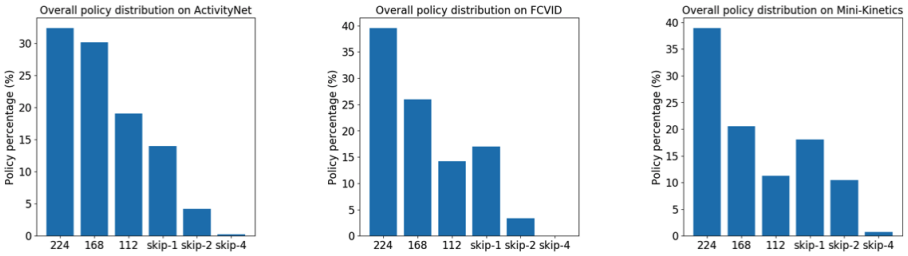
To estimate the overall GFLOPS for our framework, we compute a weighted sum based on online policy distribution and an offline GFLOPS look up table. The method to compute online policy distribution is summarized in Equation 11. To generate the look up table for GFLOPS with respect to different modules and resolutions, we first instantiate the specific network and then use THOP (https://pypi.org/project/thop/) to measure the GFLOPS. The example code snippet for computing the FLOPS for ResNet−50 at 224×224 frame resolution is given below.

```
import torch, torchvision, thop
model = getattr(torchvision.models, "resnet50")(True)
data =(torch.randn(1, 3, 224, 224),)
flops, _ = thop.profile(model, inputs=data)
```

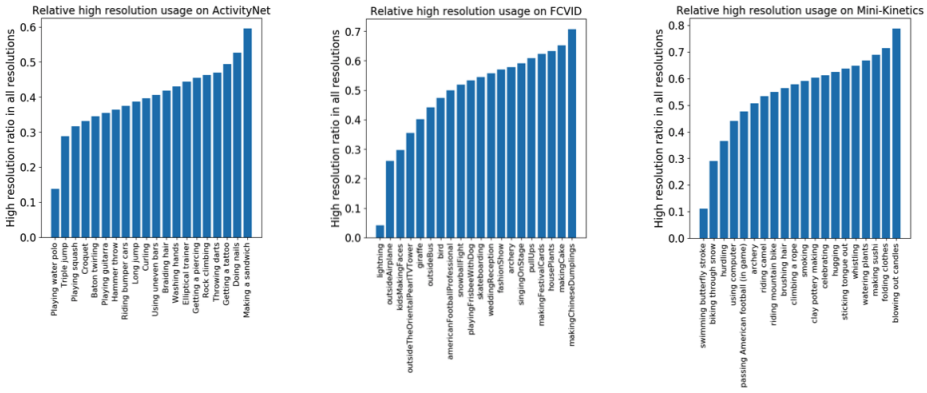
Table 2 presents all the results we need for computing GFLOPS. The GFLOPS for LSTM is approximated by "square of the input feature dimension", since we only need matrix-vector multiplications. Note that when feature dimension is around $1000 \sim 2000$, this value is normally smaller than 0.01, and hence negligible to other operations.

C Distributions

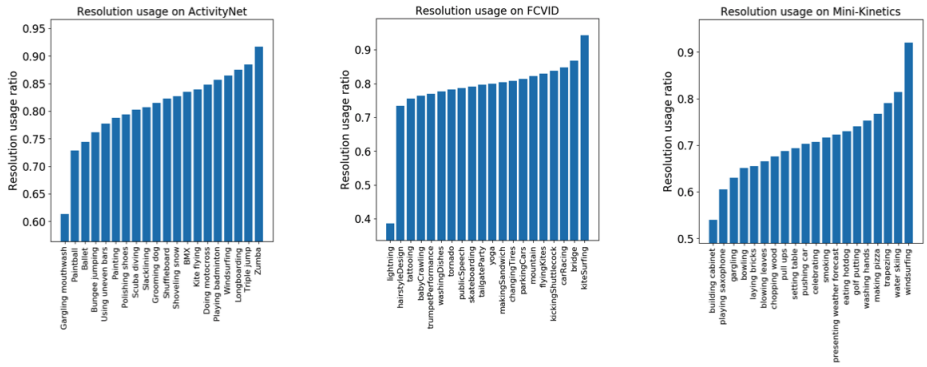
Figure 1 shows the dataset-specific and category-specific policy usages for "AR-Net(ResNet)". Videos are uniformly sampled in 8 frames. We present policy distribution (choosing 224×224/168×168/112×112 resolution or skipping 1/2/4 frames) in Figure 1(a), present a subset of classes sorted in relative high resolution usage (ratio of "choosing 224×224" over "choosing 224×224/168×168/112×112") in Figure 1(b) and list a subset of classes sorted in resolution usage ratio (ratio of "choosing 224×224/168×168/112×112" over all policies) in Figure 1(c). Only less than 1% of frames are used in 84×84 resolution in our experiments, so we omit "resolution 84" in Figure 1(a). On dataset level, we observe that AR-Net skips relatively more frames on Mini-Kinetics compared to ActivityNet and FCVID, indicating that videos in Mini-Kinetics are less motion-informative. Moreover, on the class level, samples with complex procedures (e.g. "making a sandwich" from ActivityNet in Figure 1(b)) are using more frames with high resolution, compared to the samples with static objects, scenes ("lightning" from FCVID in Figure 1(b) and (c)) or scene-related actions ("ballet" or "building cabinet" in Figure 1(c)), indicating that our learned decision policy often corresponds to the difficulty in making predictions (i.e., difficult samples require more frames with high resolution).



(a) Overall policy distribution on ActivityNet, FCVID and Mini-Kinetics



(b) Relative high resolution usage on ActivityNet, FCVID and Mini-Kinetics



(c) Resolution usage on ActivityNet, FCVID and Mini-Kinetics

Figure 1. Dataset-specific and category-specific policy usage for AR-Net(ResNet).

D Additional Qualitative Analysis

Figure 2 ∼ 4 show more qualitative results that AR-Net predicts on ActivityNet, FCVID and Mini-Kinetics. We define "difficulties" (Easy, Medium and Hard) based on their computation budgets. In general, AR-Net saves the computation greatly for examples that contain clear appearance or actions with less motion.

Figure 2. Qualitative results on ActivityNet dataset. Videos are uniformly sampled in 8 frames. The first row in each example is the original video input, and the second row represents the resolutions or skipping decisions that AR-NET chooses. We shows ground truth labels and define "difficulties" (Easy, Medium and Hard) based on their computation budgets. AR-Net can save the computation greatly for examples which contain clear appearance or actions with less motion. Best viewed in color.

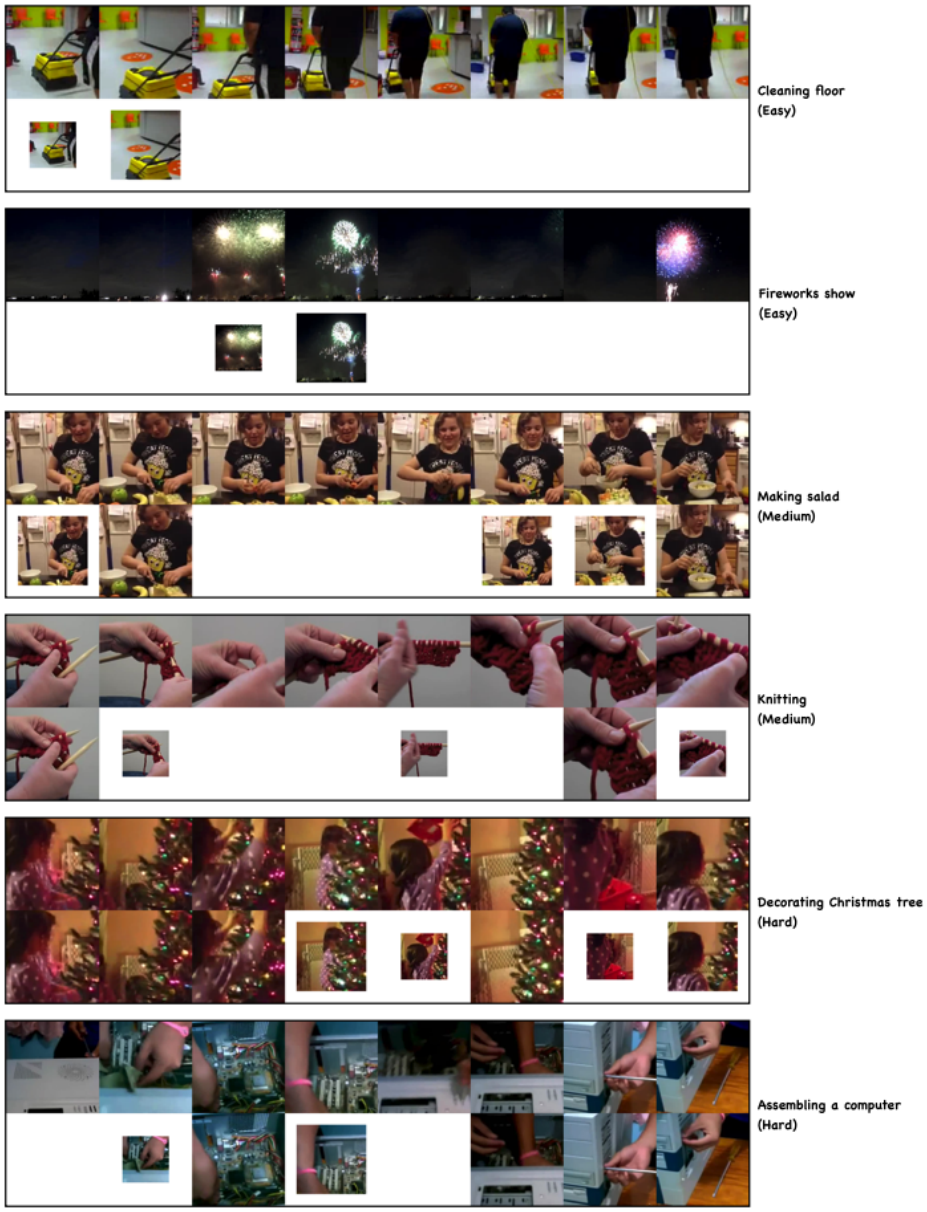


Figure 3. Qualitative results on FCVID dataset. Videos are uniformly sampled in 8 frames. The first row in each example is the original video input, and the second row represents the resolutions or skipping decisions that AR-NET chooses. We shows ground truth labels and define "difficulties" (Easy, Medium and Hard) based on their computation budgets. AR-Net can save the computation greatly for examples which contain clear appearance or actions with less motion. Best viewed in color.

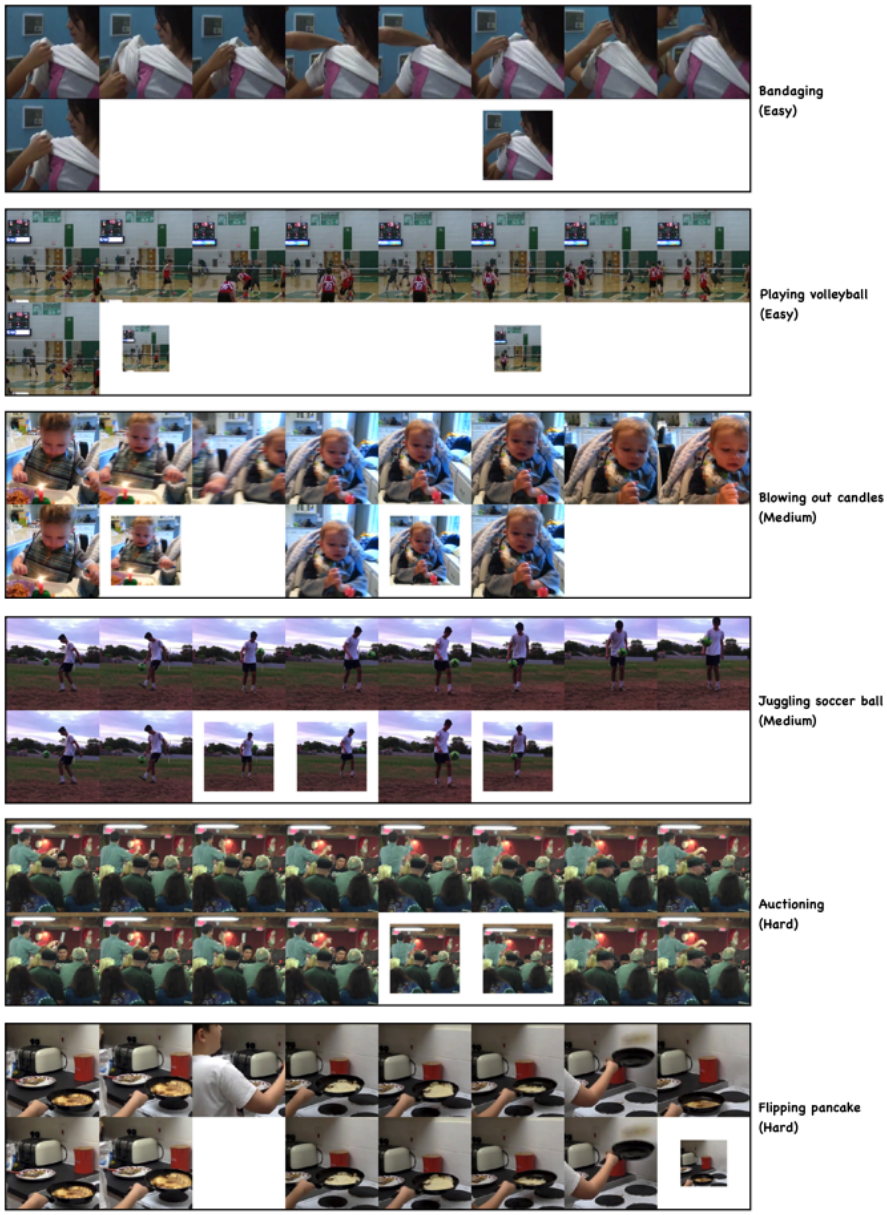


Figure 4. Qualitative results on Mini-Kinetics dataset. Videos are uniformly sampled in 8 frames. The first row in each example is the original video input, and the second row represents the resolutions or skipping decisions that AR-NET chooses. We shows ground truth labels and define "difficulties" (Easy, Medium and Hard) based on their computation budgets. AR-Net can save the computation greatly for examples which contain clear appearance or actions with less motion. Best viewed in color.

E RL vs Gumbel Softmax in Policy Learning

We conduct an experiment to compare different policy learning approaches. For the Reinforcement Learning method, we adopt the policy gradient approach and follow the same training procedures and number of epochs used in the Gumbel Softmax experiment. Based on the hyperparameters provided from the previous experiment, we further tune learning rates ($0.001 \rightarrow 0.002$ in joint-training stage; $0.0005 \rightarrow 0.001$ in finetuning) to get the best performance for the RL-based method. As shown in Table 3, Gumbel Softmax approach can achieve a better trade-off in recognition performance (less GFLOPS usage with higher mAP), showing its effectiveness over the RL-based approach.

Table 3. Performances for different learning approaches on ActivityNet-v1.3

Approach	mAP	GFLOPS/f	GFLOPS/v
Policy Gradient	72.4	3.17	50.69
Gumbel Softmax	**73.8**	**2.09**	**33.47**

Acknowledgement

This work is supported by IARPA via DOI/IBC contract number D17PC00341. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. This work is partly supported by the MIT-IBM Watson AI Lab.

Disclaimer: The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA, DOI/IBC, or the U.S. Government.