# Beyond Short Clips: End-to-End Video-level Learning with Collaborative Memories

Xitong Yang[1], Haoqi Fan[2], Lorenzo Torresani[2,3], Larry Davis[1], Heng Wang[2]

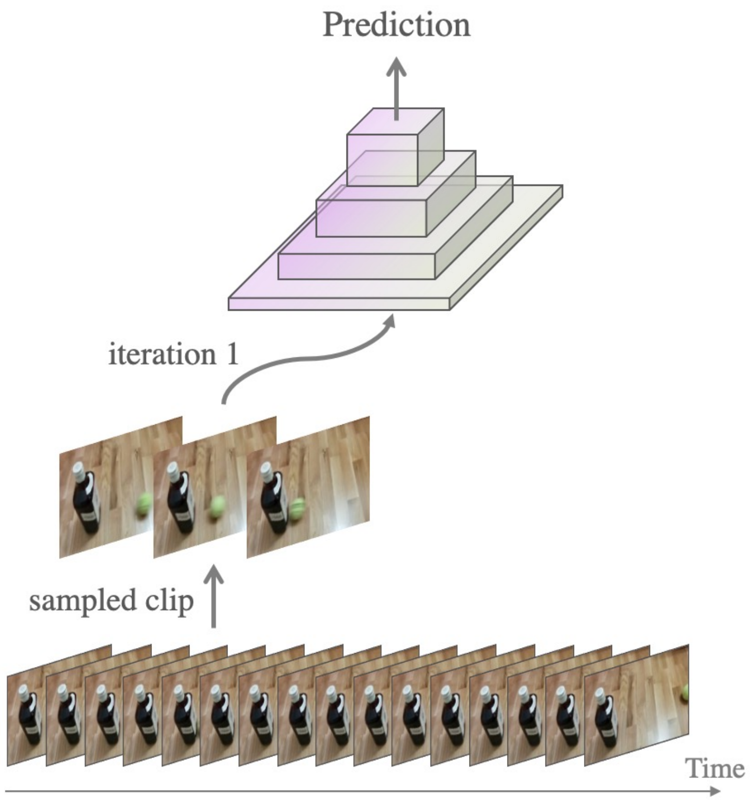
[1]University of Maryland, College Park [2]Facebook AI [3]Dartmouth

# Motivation

- End-to-end learning of 3D CNNs has emerged as the prominent paradigm for video classification
- However, modeling a long video as a whole is not feasible due to the high computational cost and large memory requirements

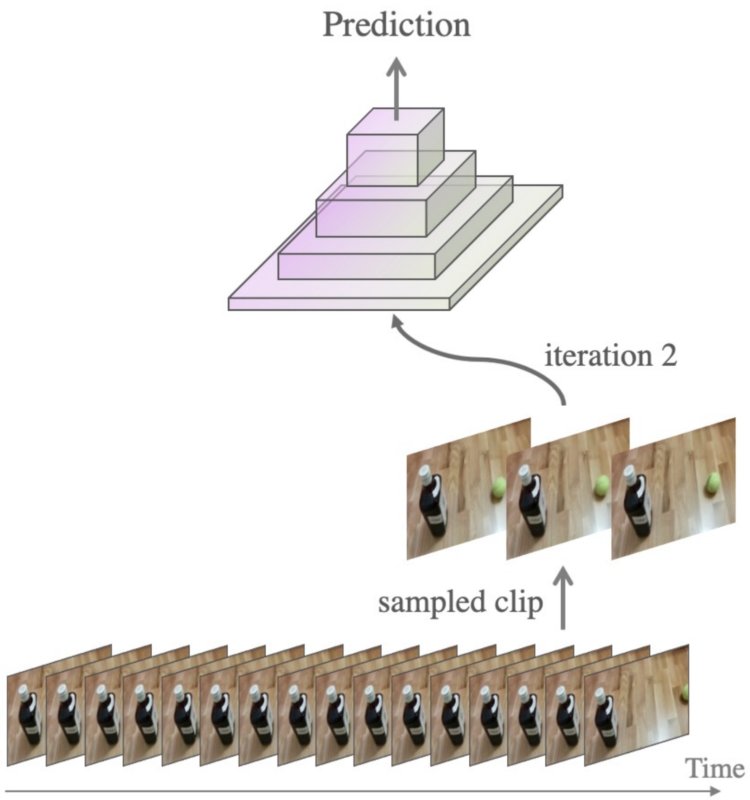
# Motivation

- End-to-end learning of 3D CNNs has emerged as the prominent paradigm for video classification
- However, modeling a long video as a whole is not feasible due to the high computational cost and large memory requirements
- The standard way of optimizing 3D video models is ***clip-level training***
  - A single short clip is sampled from the full-length video at each iteration
  - The clip-based prediction is optimized w.r.t. the video-level action label



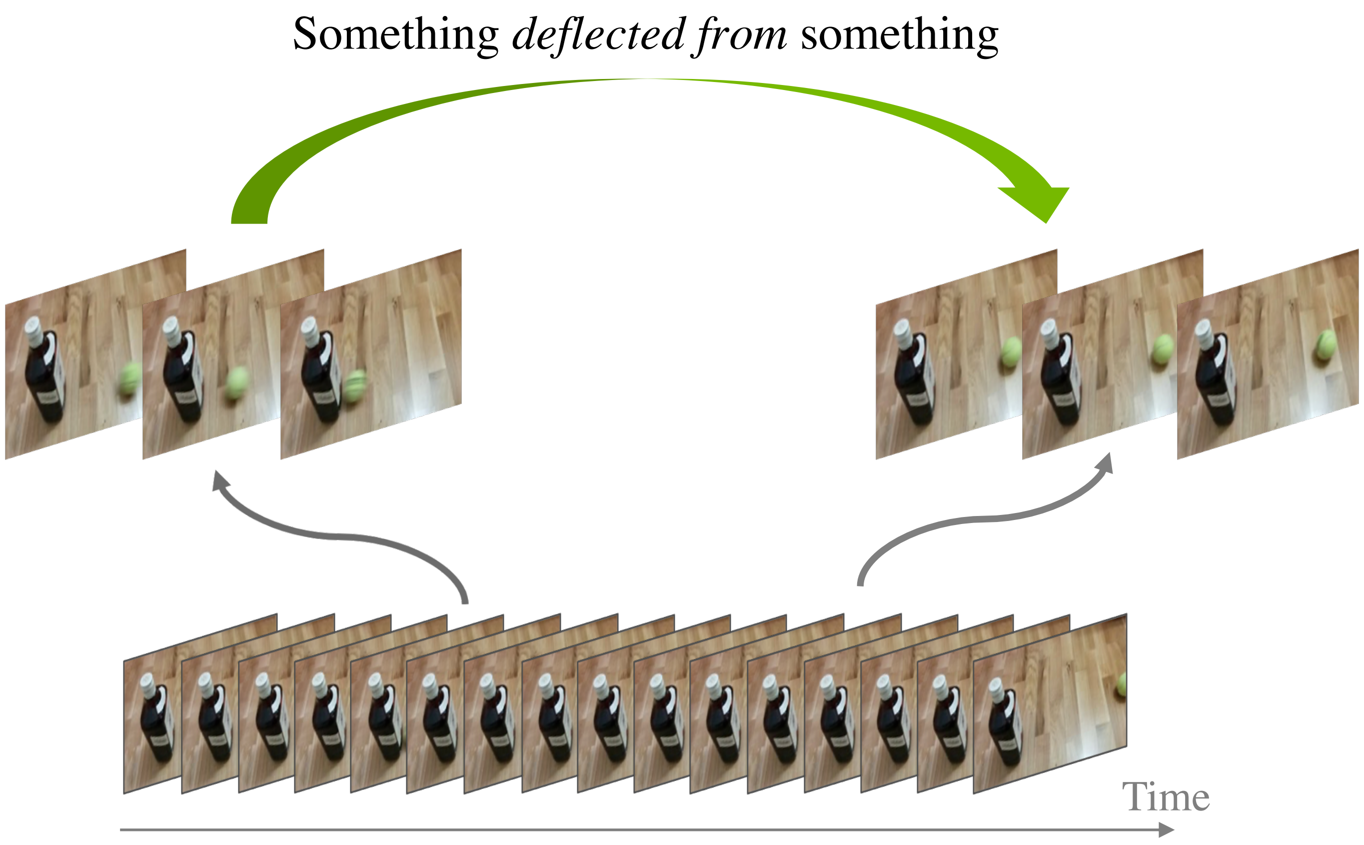
# Motivation

- End-to-end learning of 3D CNNs has emerged as the prominent paradigm for video classification
- However, modeling a long video as a whole is not feasible due to the high computational cost and large memory requirements
- The standard way of optimizing 3D video models is ***clip-level training***
  - A single short clip is sampled from the full-length video at each iteration
  - The clip-based prediction is optimized w.r.t. the video-level action label



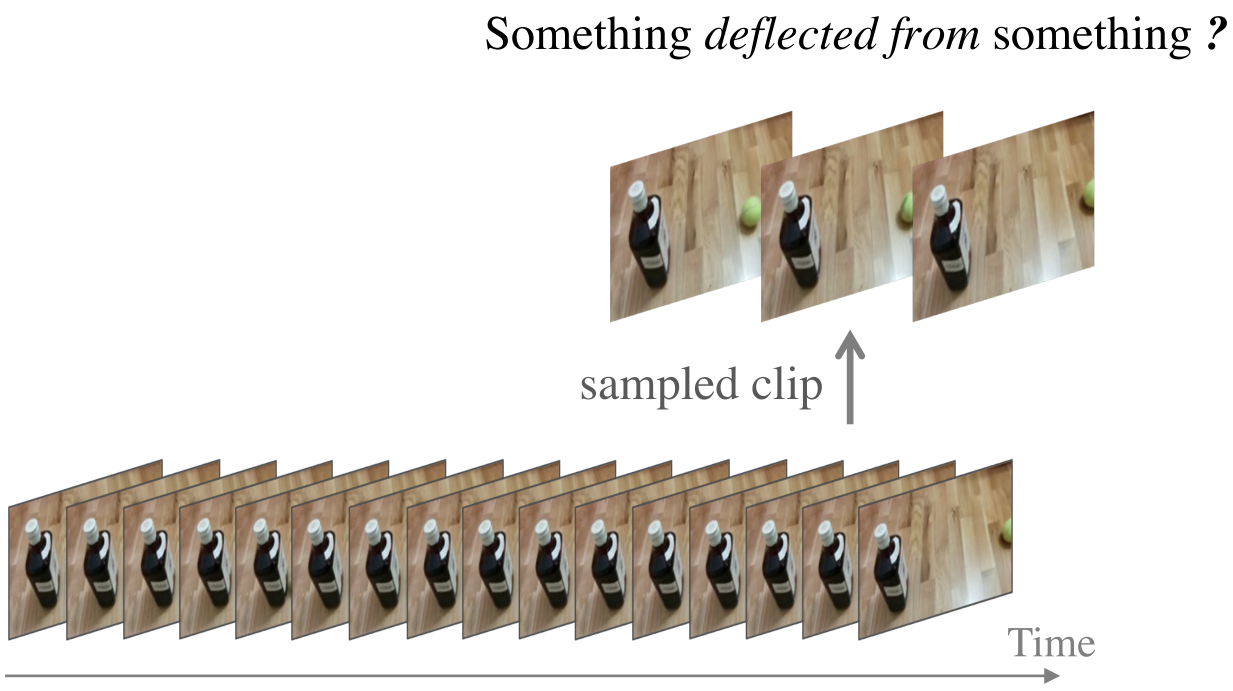
# Motivation

- Limitation of clip-level training
  - Not possible to capture long-range temporal dependencies beyond short clips



# Motivation

- Limitation of clip-level training
  - Not possible to capture long-range temporal dependencies beyond short clips
  - Video-level label may not be well represented in a brief clip

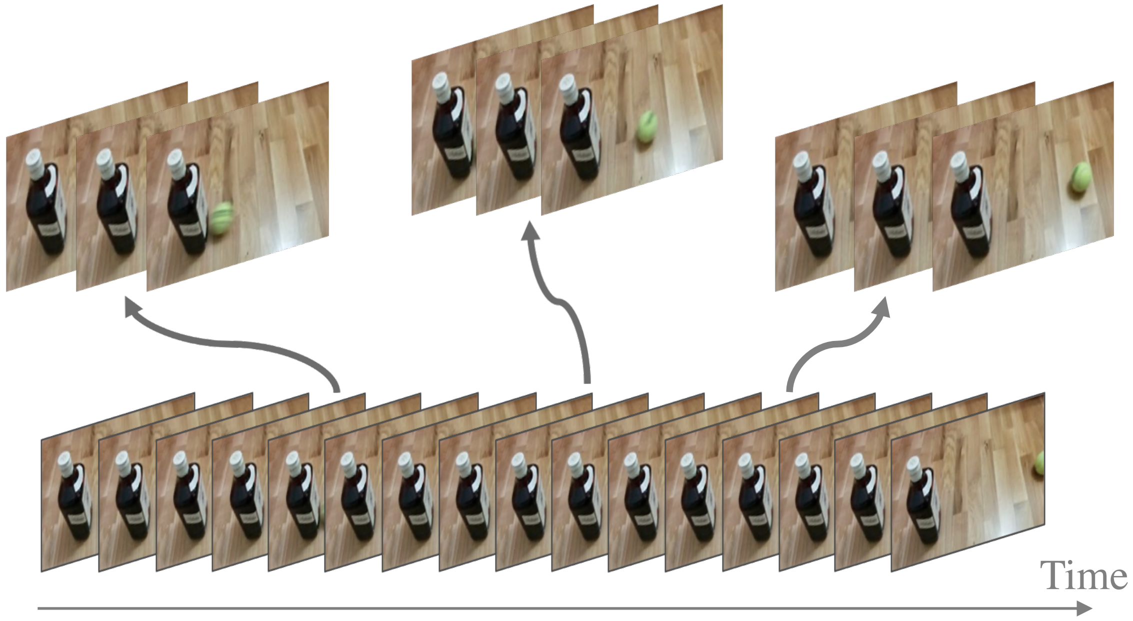


# Our Approach

- End-to-end video-level learning with Collaborative Memory (CM)
  - Optimize the *clip-based* model using *video-level* information collected from the whole video

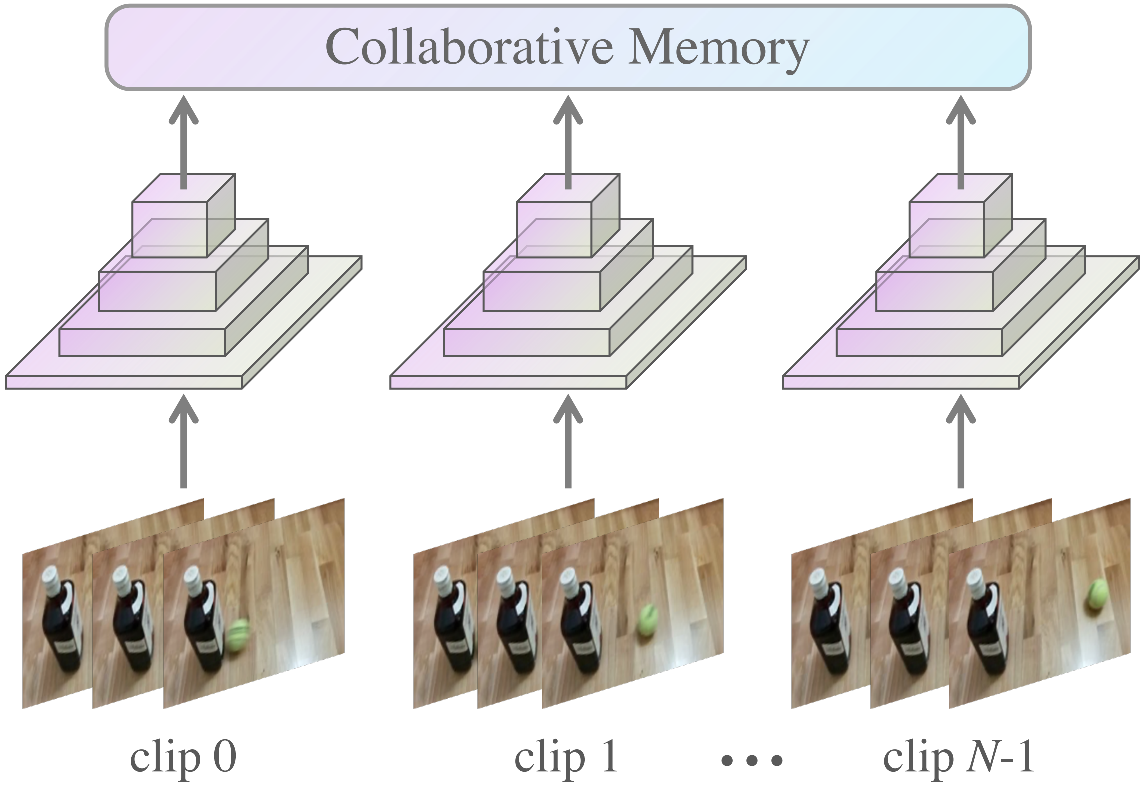
# Our Approach

- End-to-end video-level learning with Collaborative Memory (CM)
  - Optimize the *clip-based* model using *video-level* information collected from the whole video
- Multi-clip sampling at each iteration
  - Ensure sufficient temporal coverage of the video



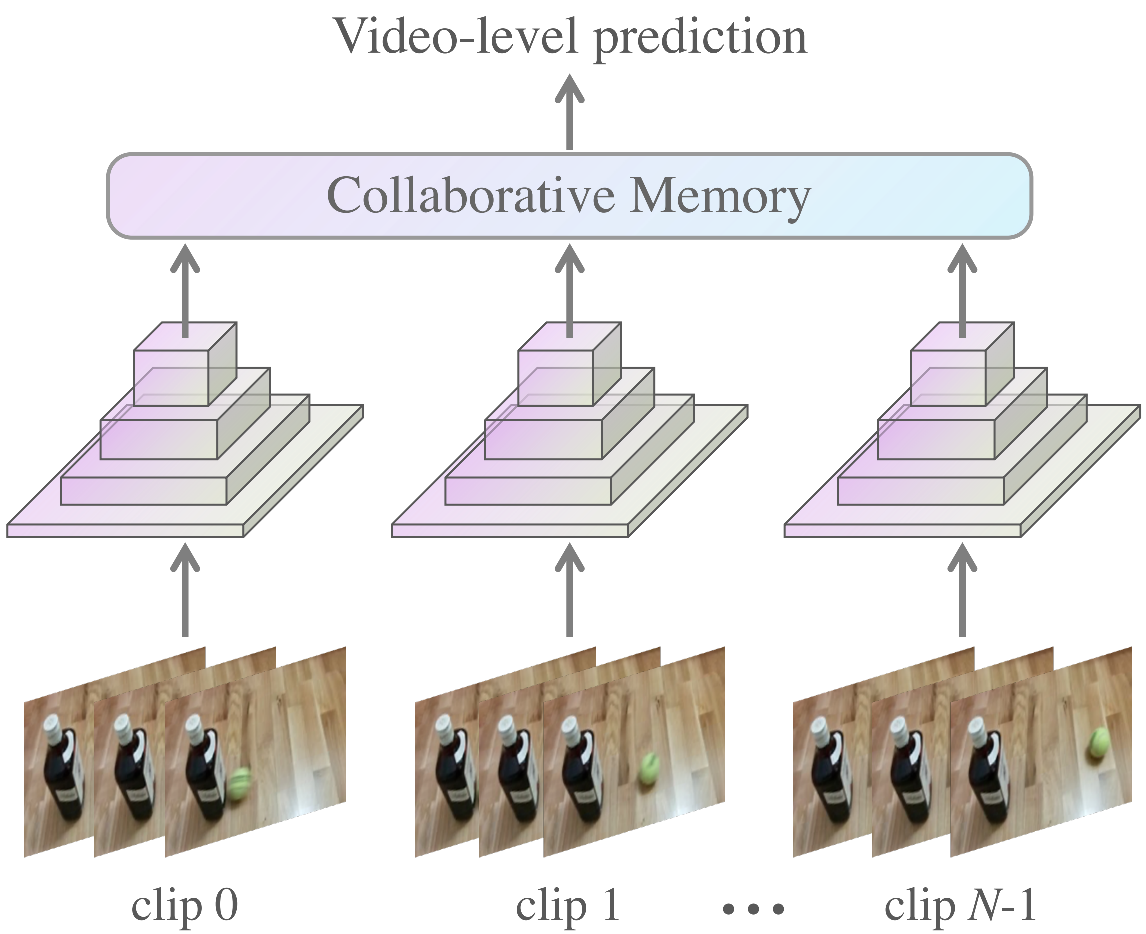
# Our Approach

- End-to-end video-level learning with Collaborative Memory (CM)
  - Optimize the *clip-based* model using *video-level* information collected from the whole video
- Multi-clip sampling at each iteration
  - Ensure sufficient temporal coverage of the video
- Collaborative memory
  - Accumulate information from multiple clips
  - Share the video-level context back with individual clips



# Our Approach

- End-to-end video-level learning with Collaborative Memory (CM)
  - Optimize the *clip-based* model using *video-level* information collected from the whole video
- Multi-clip sampling at each iteration
  - Ensure sufficient temporal coverage of the video
- Collaborative memory
  - Accumulate information from multiple clips
  - Share the video-level context back with individual clips
- Video-level supervision
  - Joint optimization of multiple clips with a video-level loss



# Collaborative Memory

- Memory interaction
  - Memory push: accumulate information from multiple clips to build a global memory
  - Memory pop: retrieve clip-specific, video-level context from the global memory
- Context infusion
  - Infuse the individual clip-based representations with video-level context

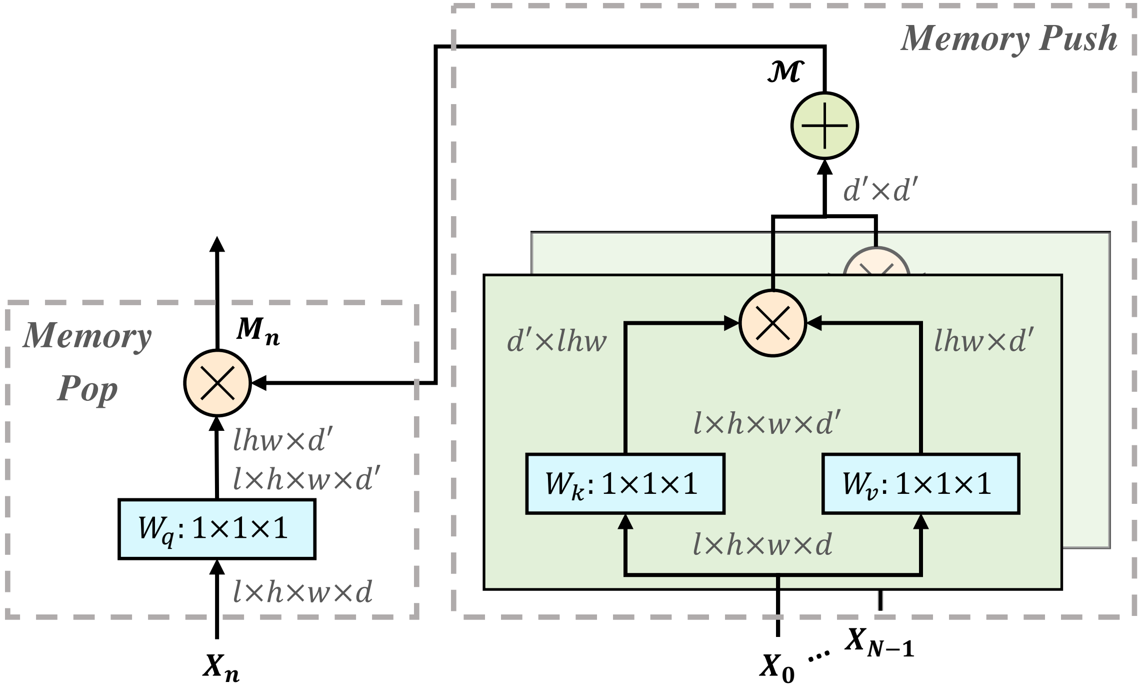
- The idea of collaborative memory is generic and can be implemented in various ways
  - The memory footprint for storing the global memory should be manageable
  - Interactions with the memory should be computationally efficient
  - Individual clip-based representations should not be dominated by the video-level context

# Collaborative Memory

- Memory interaction
  - Associate memory

$$\mathcal{M} = Push(\{X_n\}_{n=0}^{N-1}) = \frac{1}{N}\sum_{n=0}^{N-1}(X_n W_k)^T (X_n W_v)$$

$$M_n = Pop(\mathcal{M}, X_n) = (X_n W_q)\mathcal{M}$$



# Collaborative Memory

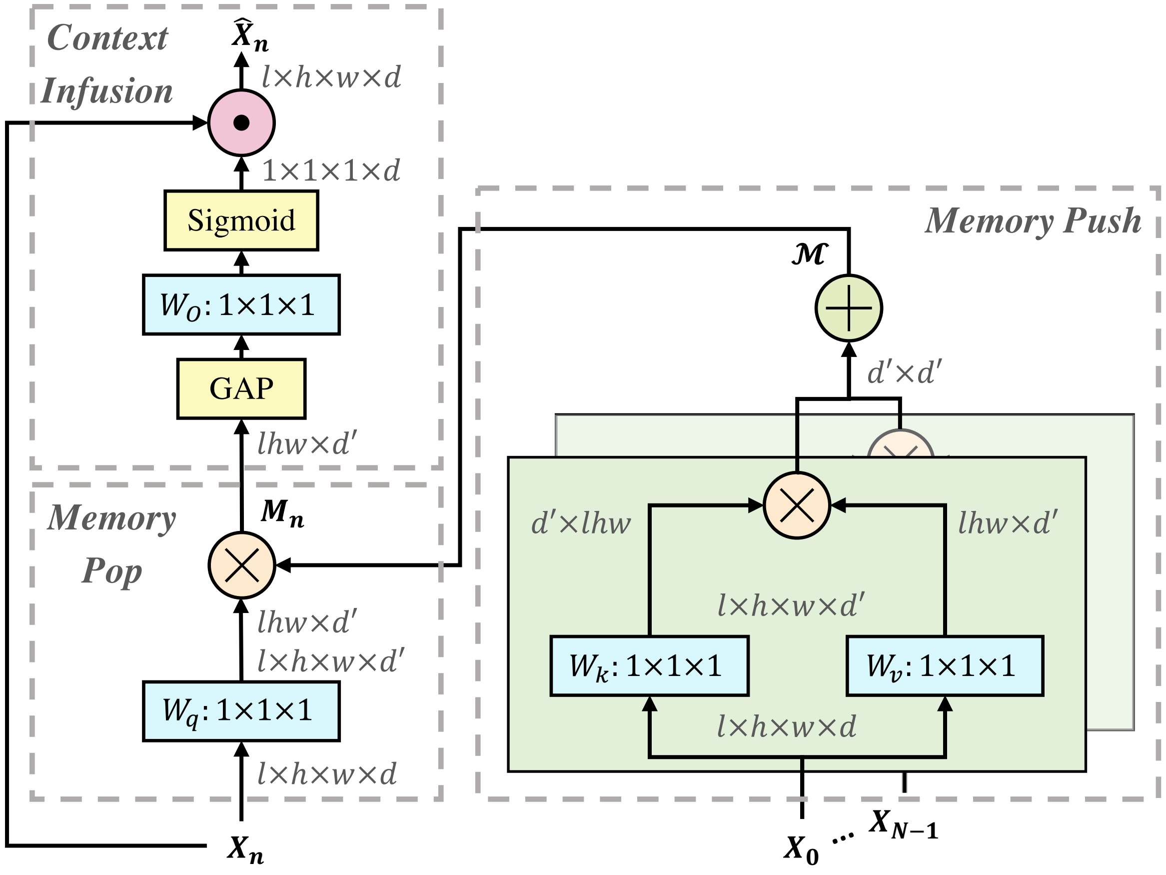
- Memory interaction
  - Associate memory

$$\mathcal{M} = Push(\{X_n\}_{n=0}^{N-1}) = \frac{1}{N}\sum_{n=0}^{N-1}(X_n W_k)^T (X_n W_v)$$

$$M_n = Pop(\mathcal{M}, X_n) = (X_n W_q)\mathcal{M}$$

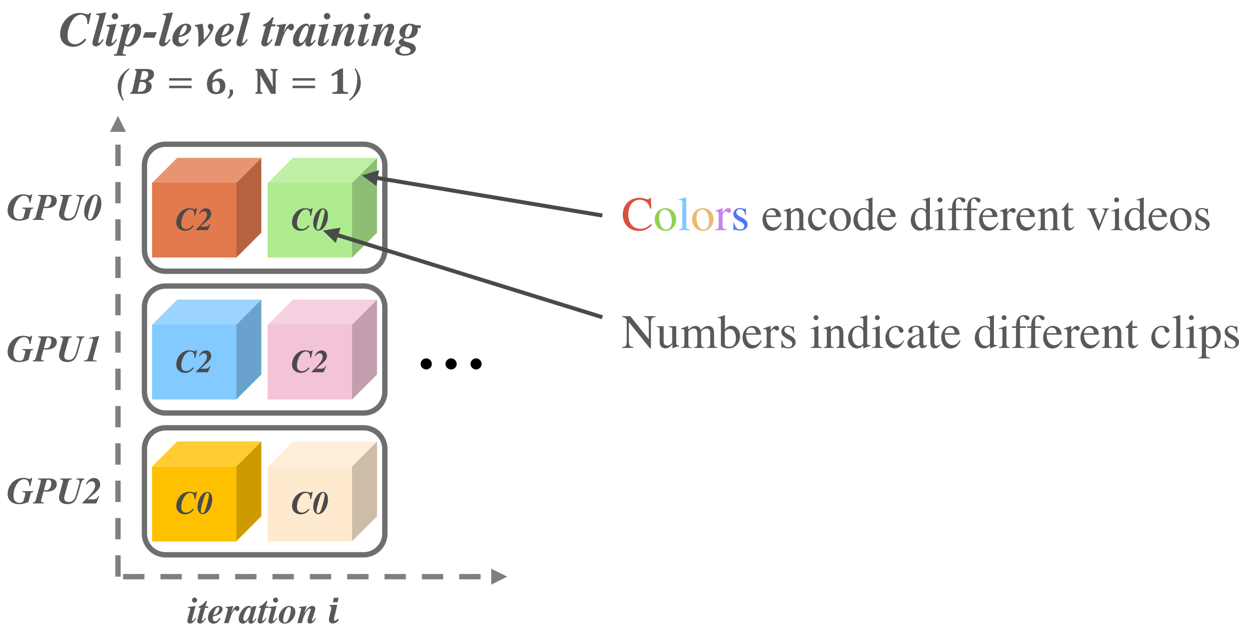
- Context infusion
  - Feature gating

$$\hat{X}_n = \sigma(Pool(M_n)W_O) \odot X_n + X_n$$



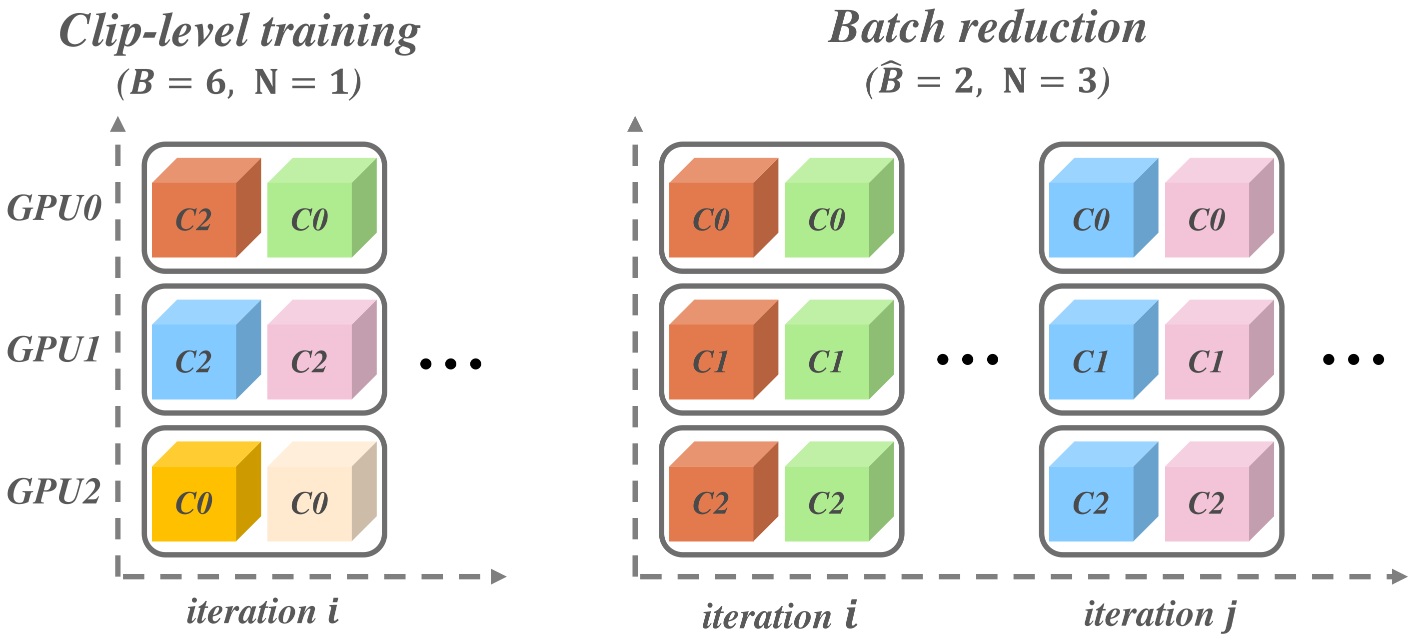
# Coping with GPU Memory Constraint

- Strategy 1: Batch reduction
  - Reduce the batch size $B$ by a factor of $N$: $\hat{B} = round(B/N)$
  - Simple, applicable to most settings in practice



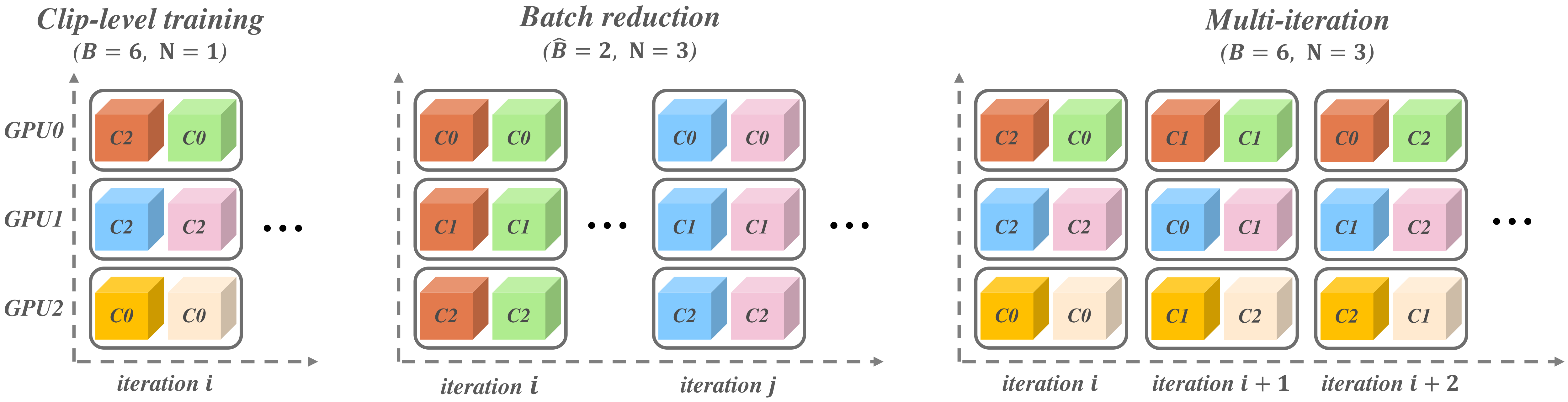
# Coping with GPU Memory Constraint

- Strategy 1: Batch reduction
  - Reduce the batch size $B$ by a factor of $N$: $\hat{B} = round(B/N)$
  - Simple, applicable to most settings in practice



# Coping with GPU Memory Constraint

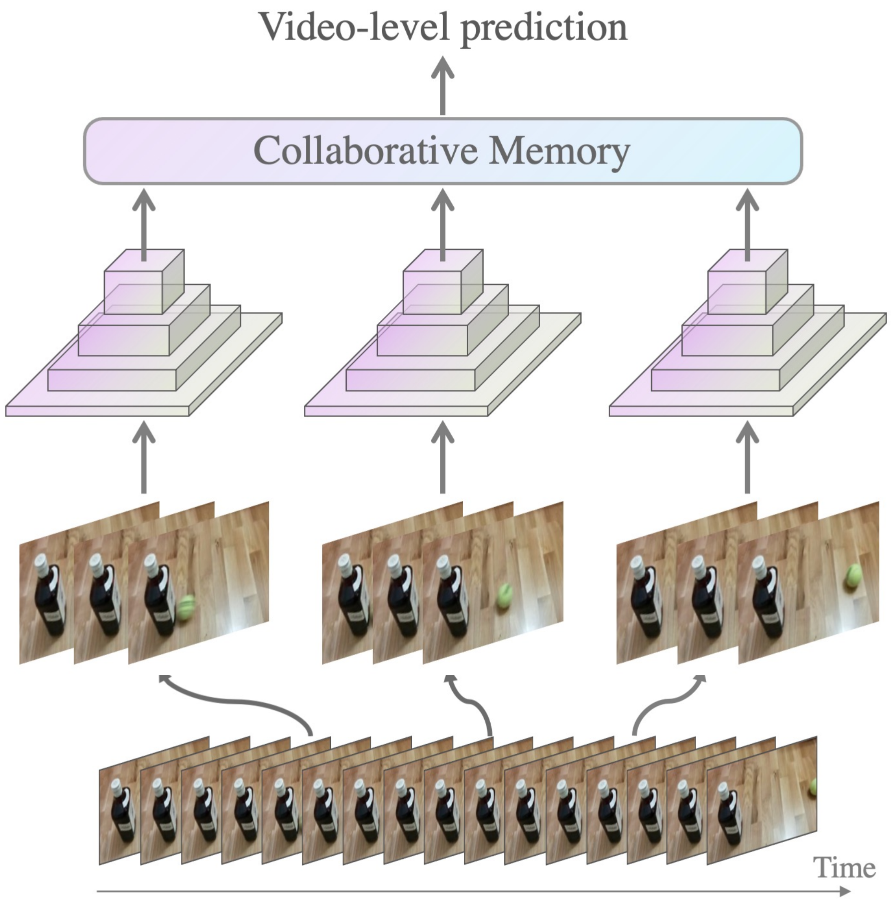
- Strategy 1: Batch reduction
  - Reduce the batch size $B$ by a factor of $N$: $\hat{B} = round(B/N)$
  - Simple, applicable to most settings in practice
- Strategy 2: Multi-iteration
  - Unroll the training of $N$ clips into $N$ consecutive iterations
  - Allow to process long videos with arbitrarily large $N$



# Our Approach

- End-to-end video-level learning with Collaborative Memory (CM)
  - Optimize the *clip-based* model using *video-level* information collected from the whole video
- Multi-clip sampling at each iteration
  - Ensure sufficient temporal coverage of the video
- Collaborative memory
  - Accumulate information from multiple clips
  - Share the video-level context back with individual clips
- Video-level supervision
  - Joint optimization of multiple clips with a video-level loss

$$\mathcal{L}_{video} = \frac{1}{N}\sum_{n=0}^{N-1}\mathcal{L}\left(h(\hat{X}_n)\right) + \alpha\mathcal{L}\left(\frac{1}{N}\sum_{n=0}^{N-1}h(\hat{X}_n)\right)$$

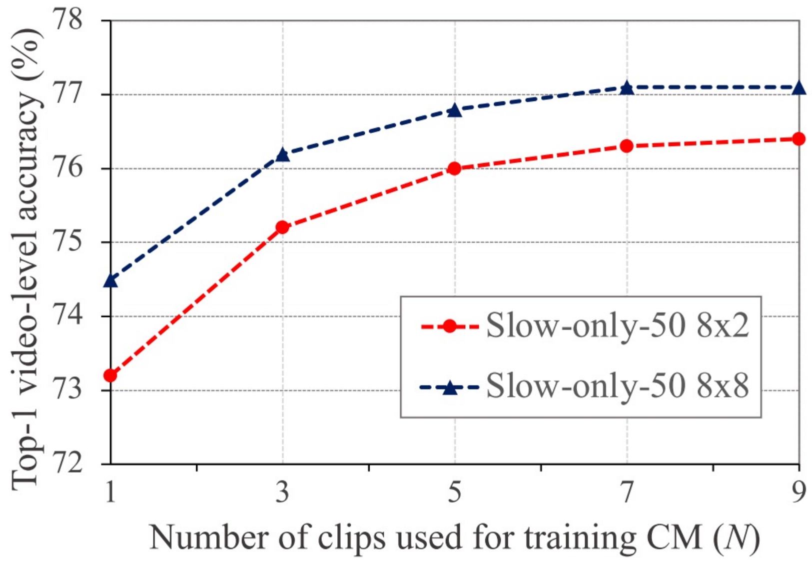


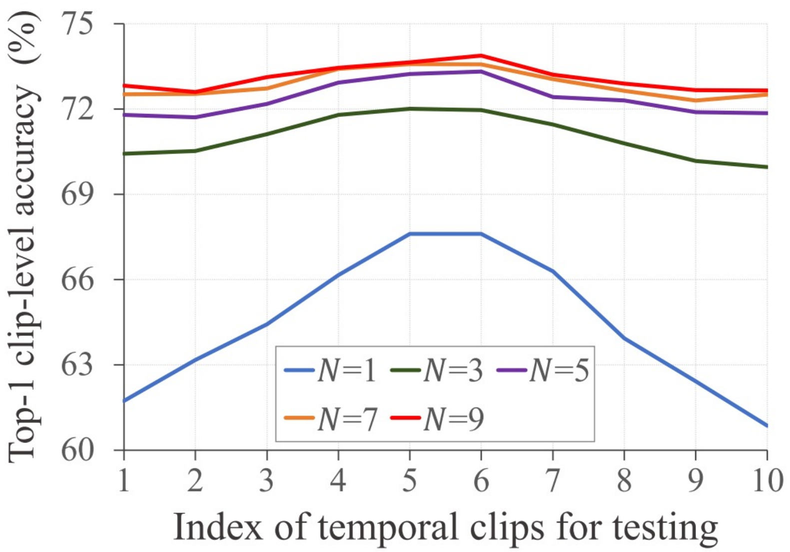
# Experiments

- Evaluating CM for video-level learning
  - Experiments on Kinetics-400 dataset
  - We use Slow-only network (Feichtenhofer et al) with 50 layers and the input clip length is 8×8 (frames × stride)
  - We first train the backbone following its original schedule, then re-train it in conjunction with our collaborative memory for video-level learning

# Evaluating CM for Video-level Learning

- Impact of temporal coverage on video-level learning
  - Ablate the number of clips $N$ used for training CM ($N = 1$  of clips $N$ used clip-level training)




- Video-level learning with CM significantly improves the *video-level* accuracy
- ***2.6%*** improvement over single-clip baseline with $N = 9$

- *Clip-level* accuracy is significantly improved especially for clips near the boundary of the video

# Evaluating CM for Video-level Learning

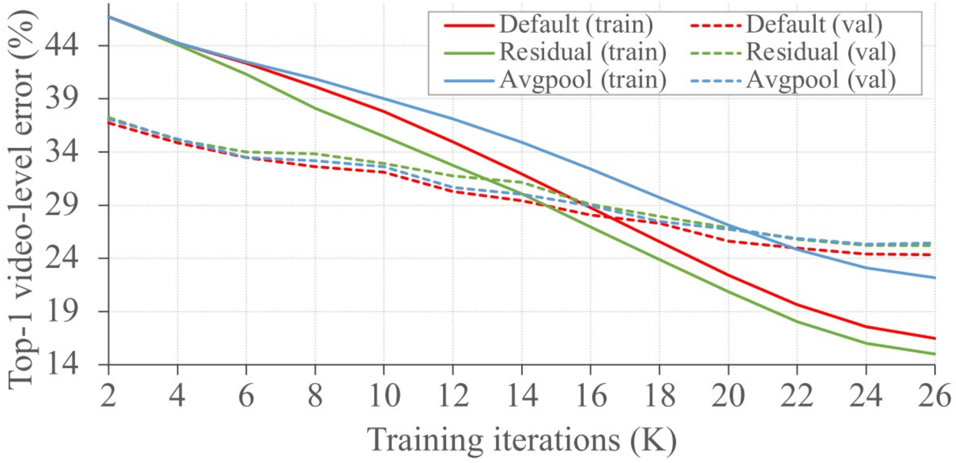
- Impact of temporal coverage on video-level learning
  - Ablate the number of clips $N$ used for training CM ($N = 1$ of clips $N$ used clip-level training)
- Generalization to different video backbones
  - Our CM framework does not make any assumption about the backbone

| Model | Baseline | Ours | $\triangle$ | FLOPs |
|---|---|---|---|---|
| Slow-only-50 8×8 [11] | 74.4 | **76.8** | +2.4 | 1.03× |
| I3D-50+NL 32×2 [52] | 74.9 | **77.5** | +2.4 | 1.02× |
| R(2+1)D-50 16×2 [48] | 75.7 | **78.0** | +2.3 | 1.01× |
| SlowFast-50 4×16 [11] | 75.6 | **77.8** | +2.2 | 1.02× |
| SlowFast-50 8×8 [11] | 76.8 | **78.9** | +2.1 | 1.03× |

# Ablation Studies

- Comparing different design choices for the memory mechanism
  - Pooling for memory interaction (avgpool) achieves inferior results due to the lack of inter-clip interaction
  - Removing feature gating operation (residual) results in performance drop due to over-fitting to the video-level context during training

| Setting | Associative | Gating | Top-1 |
|---|---|---|---|
| Multi-clip (w/o memory) | | | 75.5 |
| CM (avgpool) | | ✓ | 75.8 |
| CM (residual) | ✓ | | 76.0 |
| CM (default) | ✓ | ✓ | **76.8** |



# Ablation Studies

- Please refer to the paper for more ablation studies on different components of our framework and training strategies

| Multi-clip | Memory | End-to-end | Top-1 |
|---|---|---|---|
| | ✓ | ✓ | 74.5 |
| ✓ | | ✓ | 75.5 |
| ✓ | ✓ | | 75.9 |
| ✓ | ✓ | ✓ | **76.8** |

(a) Evaluating **different components** of our video-level learning framework.

| Setting | Associative | Gating | Top-1 |
|---|---|---|---|
| Multi-clip (w/o memory) | | | 75.5 |
| CM (avgpool) | | ✓ | 75.8 |
| CM (residual) | ✓ | | 76.0 |
| CM (default) | ✓ | ✓ | **76.8** |

(b) Comparing **different designs** of our collaborative memory mechanism.

| Setting | #Param. | Top-1 |
|---|---|---|
| $\alpha = 1$ | 49.2 M | 76.8 |
| $\alpha = 2$ | 40.9 M | 76.8 |
| $\alpha = 4$ | 36.7 M | **76.8** |
| $\alpha = 8$ | 34.6 M | 76.4 |

(c) Varying **channel reduction ratio** $\alpha = d/d^{'}$.

| Model | Stage-wise | Top-1 |
|---|---|---|
| Slow-only | | 76.1 |
| Slow-only | ✓ | **76.8** |
| R(2+1)D | | 77.7 |
| R(2+1)D | ✓ | **78.0** |

(d) **Stage-wise training** *vs.* training everything from scratch.

| Model | Batch reduction | Multi-iteration | Top-1 |
|---|---|---|---|
| Slow-only | | ✓ | 76.6 |
| Slow-only | ✓ | | **76.8** |
| R(2+1)D | | ✓ | 77.9 |
| R(2+1)D | ✓ | | **78.0** |

(e) Comparing different ways of training CM: **batch reduction** *vs.* multi-iteration.

| Model | Temporal stride | | | | CM |
|---|---|---|---|---|---|
| | **2** | **4** | **8** | **16** | |
| Slow-only | 73.2 | 74.3 | 74.4 | 74.4 | **76.8** |
| R(2+1)D | 75.7 | 76.4 | 75.0 | 72.2 | **78.0** |

(f) Comparing CM with backbones using clips with **large temporal strides**.

# Comparison with the State-of-the-Arts

- Kinetics-400 and Kinetics-700 dataset
  - Achieve state-of-the-art results without pre-training on other datasets or using optical flow

| **Methods** | **Pretrain** | **Only RGB** | **GFLOPs × crops** | **Top-1** |
|---|---|---|---|---|
| I3D [5] | ImageNet | ✗ | 216×N/A | 75.7 |
| S3D-G [58] | ImageNet | ✗ | 142.8×N/A | 77.2 |
| LGD-3D-101 [38] | ImageNet | ✗ | N/A | 81.2 |
| I3D-101+NL [52] | ImageNet | ✓ | 359×30 | 77.7 |
| ip-CSN-152 [47] | Sports1M | ✓ | 109×30 | 79.2 |
| CorrNet-101 | Sports1M | ✓ | 224×30 | 81.0 |
| MARS+RGB [6] | none | ✓ | N/A | 74.8 |
| DynamoNet [8] | none | ✓ | N/A | 77.9 |
| CorrNet-101 [50] | none | ✓ | 224×30 | 79.2 |
| SlowFast-101 8×8 [11] | none | ✓ | 106×30 | 77.9 |
| SlowFast-101 16×8 [11] | none | ✓ | 213×30 | 78.9 |
| SlowFast-101+NL 16×8 [11] | none | ✓ | 234×30 | 79.8 |
| **Ours** (R(2+1)D-101 32×2) | none | ✓ | 243×30 | 80.5 |
| **Ours** (SlowFast-101 8×8) | none | ✓ | 128×30 | 80.0 |
| **Ours** (SlowFast-101+NL 8×8) | none | ✓ | 137×30 | **81.4** |

+2.5%

| **Methods** | **Pretrain** | **GFLOPs × crops** | **Top-1** |
|---|---|---|---|
| SlowFast-101+NL 8×8 [11] | K600 | 115×30 | 70.6 |
| SlowFast-101+NL 16×8 [11] | K600 | 234×30 | 71.0 |
| SlowFast-50 4×16* | K600 | 36×30 | 66.1 |
| SlowFast-101 8×8* | K600 | 126×30 | 69.2 |
| SlowFast-101+NL 8×8* | K600 | 135×30 | 70.2 |
| **Ours** (SlowFast-50 4×16) | K600 | 37×30 | 68.3 |
| **Ours** (SlowFast-101 8×8) | K600 | 128×30 | 70.9 |
| **Ours** (SlowFast-101+NL 8×8) | K600 | 137×30 | **72.4** |

+2.2%

# Comparison with the State-of-the-Arts

- Charades dataset
  - Longer-range activities (30 seconds on average) than Kinetics, multi-label classification
  - Outperform other recent work on long-range temporal modeling (*e.g.*, Timeception (Hussein et al), LFB (Wu et al))

| **Methods** | **Pretrain** | **GFLOPs × crops** | **Top-1** |
|---|---|---|---|
| TRN [60] | ImageNet | N/A | 25.2 |
| I3D-101+NL [52] | ImageNet+K400 | 544 × 30 | 37.5 |
| STRG [53] | ImageNet+K400 | 630 × 30 | 39.7 |
| Timeception [23] | K400 | N/A | 41.1 |
| LFB (I3D-101+NL) [55] | K400 | N/A | 42.5 |
| SlowFast-101+NL [11] | K400 | 234×30 | 42.5 |
| AVSlowFast-101+NL [57] | K400 | 278×30 | 43.7 |
| SlowFast-50 16×8* | K400 | 131×30 | 39.4 |
| SlowFast-101+NL 16×8* | K400 | 273×30 | 41.3 |
| **Ours** (SlowFast-50 16×8) | K400 | 135×30 | 42.9 |
| **Ours** (SlowFast-101+NL 16×8) | K400 | 277×30 | **44.6** |

*+3.3%*

# Collaborative Memory for Action Detection

- AVA dataset
  - Sample multiple clips within a certain temporal window $[t-w, t+w]$ to detect action at time $t$

| Methods | Pretrain | mAP |
|---|---|---|
| ACRN [43] | K400 | 17.4† |
| AVSF-50 4×16 [57] | K400 | 27.8† |
| AT (I3D) [13] | K400 | 25.0 |
| LFB(R50+NL) [55] | K400 | 25.8 |
| R50+NL* [55] | K400 | 23.6 |
| SF-50 4×16* [11] | K400 | 23.6 |
| **Ours** (R50+NL) | K400 | 26.3 |
| **Ours** (SF-50 4×16) | K400 | 25.8 |

*+2.2%*

| Methods | Pretrain | mAP |
|---|---|---|
| AVSF-101 8×8 [57] | K400 | 28.6† |
| AIA(SF-50 4×16) [45] | K700 | 29.8† |
| AIA(SF-101 8×8) [45] | K700 | 32.3† |
| SF-101+NL 8×8 [11] | K600 | 29.0 |
| SF-50 4x16* [11] | K700 | 26.9 |
| SF-101 8x8* [11] | K700 | 29.0 |
| **Ours** (SF-50 4×16) | K700 | 29.8 |
| **Ours** (SF-101 8×8) | K700 | 31.6 |

*+2.6%*

# Conclusion

- We presented an end-to-end learning framework that optimizes classification models using video-level information
- Our approach hinges on a collaborative memory mechanism that captures long-range temporal dependencies beyond short clips
- Our approach significantly improves the accuracy of video models on both action recognition and detection benchmarks