Revisiting Classifier: Transferring Vision-Language Models for Video Recognition

Wenhao Wu\textsuperscript{1,2} \hspace{1cm} Zhun Sun\textsuperscript{2} \hspace{1cm} Wanli Ouyang\textsuperscript{1,3}

\textsuperscript{1}The University of Sydney \hspace{1cm} \textsuperscript{2}Baidu Inc. \hspace{1cm} \textsuperscript{3}Shanghai AI Laboratory
Task: What is Video Recognition?

**Video Recognition**: classify the short clip or untrimmed video into pre-defined class.
Task: What is Video Recognition?

**Video Recognition:** classify the short clip or untrimmed video into pre-defined class.

- More than simply recognizing objects
- Complex person-person interaction & people-object interactions
- Videos bring motions
Video Recognition Pipeline

- Sample RGB frames or Optical Flows or RGB Diff
- 2D CNN or 3D CNN or Transformer with temporal modeling
- Video-level representation

classifier
backstroke
CLIP: A Web-scale Pre-trained Vision-Language Model

1. Contrastive pre-training

2. Create dataset classifier from label text

3. Use for zero-shot prediction

400M image-text pairs for pre-training

Radford, Alec, et al. “Learning transferable visual models from natural language supervision.” *International Conference on Machine Learning*. PMLR, 2021.
How to transfer CLIP model for video recognition?

1. The typical vision-only transferring framework

Efficient Training but limited performance, especially on zero/few shot scenario
How to transfer CLIP model for video recognition?

2. The recent vision-language transferring framework

**CLIP Pre-trained Textual Encoder**

- Textual Encoder
  - a video of a person {CLS}

**CLIP Pre-trained Visual Encoder**

- Visual Encoder
  - Videos

**Good performance but**: 
- More parameters
- Require large batch size for contrastive learning
- More training time for convergence
How to transfer CLIP model for video recognition?

3. Our efficient vision-language transferring framework

Efficient but not effective

Existing transferring paradigm for video recognition

(a) Standard vision-only tuning paradigm

Efficient

(b) Vision-language tuning paradigm

Effective but not efficient
How to transfer CLIP model for video recognition?

3. Our efficient vision-language transferring framework

Key Observations: Revisiting Classifier

Figure. Inter-class correlation maps of “embeddings of class labels” for 20 categories on Kinetics-400. **Left:** The extracted textual vectors of class labels, **Right:** The “embeddings” from learned classifier.
How to transfer CLIP model for video recognition?

3. Our efficient vision-language transferring framework

Revisiting Classifier: *From a frozen classifier perspective*

Q: How to obtain inter-class correlation?
How to transfer CLIP model for video recognition?

3. Our efficient vision-language transferring framework

Revisiting Classifier: From a frozen classifier perspective

Q: How to obtain inter-class correlation?

A1: Transferring visual statistic knowledge.

A2: Transferring textual semantic knowledge.

(c) Revisiting the classifier for efficient tuning
How to transfer CLIP model for video recognition?

3. Our efficient vision-language transferring framework

Existing transferring paradigm for video recognition

Efficient but not effective

(a) Standard vision-only tuning paradigm

Efficient

(b) Vision-language tuning paradigm

Effective but not efficient

Revisiting Classifier: From a frozen classifier perspective

Q: How to obtain inter-class correlation?

A1: Transferring visual statistic knowledge.

A2: Transferring textual semantic knowledge.

(c) Revisiting the classifier for efficient tuning
### Comparisons with SOTAs

| Method               | Input  | Pre-train | Top-1 | Top-5 | FLOPs×Views | Param |
|----------------------|--------|-----------|-------|-------|-------------|-------|
| NL I3D-101 [58]      | 128×224² | IN-1K     | 77.7  | 93.3  | 359×10×3    | 61.8  |
| MVFNet₄₄₉ [60]       | 24×224²  | IN-1K     | 79.1  | 93.8  | 188×10×3    | -     |
| SlowFast NL101 [14]  | 16×224²  | Scratch   | 79.8  | 93.9  | 234×10×3    | 59.9  |
| X3D-XXL [13]         | 16×440²  | Scratch   | 80.4  | 94.6  | 144×10×3    | 20.3  |
| MViT-B, 64×3 [11]    | 64×224²  | Scratch   | 81.2  | 95.1  | 455×3×3     | 36.6  |

**Methods with large-scale pre-training**

| Method               | Input  | Pre-train | Top-1 | Top-5 | FLOPs×Views | Param |
|----------------------|--------|-----------|-------|-------|-------------|-------|
| TimeFormer-L-2 [2]   | 96×224² | IN-21K    | 80.7  | 94.7  | 2380×1×3    | 121.4 |
| ViViT-L/16×2 [1]     | 32×320² | IN-21K    | 81.3  | 94.7  | 3992×4×3    | 310.8 |
| VideoSwin-L [36]     | 32×384² | IN-21K    | 84.9  | 96.7  | 2107×10×5   | 200.0 |
| ip-CSN-152 [51]      | 32×224² | IG-65M    | 82.5  | 95.3  | 109×10×3    | 32.8  |
| ViViT-L/16×2 [1]     | 32×320² | JFT-300M  | 83.5  | 95.5  | 3992×4×3    | 310.8 |
| ViViT-H/16×2 [1]     | 32×224² | JFT-300M  | 84.8  | 95.8  | 8316×4×3    | 647.5 |
| TokLearner-L/100 [44]| 32×224² | JFT-300M  | 85.4  | 96.3  | 4076×4×3    | 450   |
| MTV-H [66]           | 32×224² | JFT-300M  | 85.8  | 96.6  | 3706×4×3    | -     |
| CoVeR [71]           | 16×448² | JFT-300M  | 86.3  | -     | -×1×3       | -     |
| Florence [69]        | 32×384² | FLD-900M  | 86.5  | 97.3  | -×4×3       | 647   |
| CoVeR [71]           | 16×448² | JFT-3B    | 87.2  | -     | -×1×3       | -     |
| VideoPrompt ViT-B/16 [25] | 16×224² | WIT-400M  | 76.9  | 93.5  | -           | -     |
| ActionCLIP ViT-B/16 [57] | 32×224² | WIT-400M  | 83.8  | 96.2  | 563×10×3    | 141.7 |

| Ours ViT-L/14        | 32×224² | WIT-400M  | 87.1  | 97.4  | 1662×4×3    | 230.7 |
| Ours ViT-L/14        | 32×336² | WIT-400M  | 87.8  | 97.6  | 3829×1×3    | 230.7 |

### Results on ActivityNet dataset

| Method               | Top-1 | mAP |
|----------------------|-------|-----|
| ListenToLook [16]    | -     | 89.9|
| MARL [61]            | 85.7  | 90.1|
| DSANet [62]          | -     | 90.5|
| TSQNet [63]          | 88.7  | 93.7|
| NSNet [64]           | 90.2  | 94.3|
| **Ours ViT-L**       | **92.9** | **96.5** |
| **Ours ViT-L (336↑)**| **93.3** | **96.9** |

### Results on Kinetics-400 dataset

| Method               | Top-1 | mAP |
|----------------------|-------|-----|
| ListenToLook [16]    | -     | 89.9|
| MARL [61]            | 85.7  | 90.1|
| DSANet [62]          | -     | 90.5|
| TSQNet [63]          | 88.7  | 93.7|
| NSNet [64]           | 90.2  | 94.3|
| **Ours ViT-L**       | **92.9** | **96.5** |
| **Ours ViT-L (336↑)**| **93.3** | **96.9** |

### Results on UCF101 & HMDB51 dataset

| Method               | UCF-101 | HMDB-51 |
|----------------------|---------|---------|
| ARTNet [55]          | 94.3%   | 70.9%   |
| I3D [6]              | 95.6%   | 74.8%   |
| R(2+1)D [52]         | 96.8%   | 74.5%   |
| S3D-G [65]           | 96.8%   | 75.9%   |
| TSM [33]             | 95.9%   | 73.5%   |
| STM [24]             | 96.2%   | 72.2%   |
| TEINet [35]          | 96.7%   | 72.1%   |
| MVFNet [60]          | 96.6%   | 75.7%   |
| TDN [56]             | 97.4%   | 76.4%   |
| **Ours ViT-L**       | **98.1%** | **81.3%** |
| **Ours ViT-L (336↑)**| **98.2%** | **81.3%** |
## Comparison with Few-shot SOTAs

| Method         | shot | HMDB  | UCF  | ANet | K400 |
|----------------|------|-------|------|------|------|
| VideoSwin [36] | 2    | 20.9  | 53.3 | -    | -    |
| VideoPrompt [25]| 5    | 56.6  | 79.5 | -    | 58.5 |
| X-Florence [40]| 2    | 51.6  | 84.0 | -    | -    |
| Ours ViT-L     |      | | | | |
|                | 0    | 53.8  | 71.9 | 75.6 | 61.0 |
|                | 1    | **72.7** | **96.4** | **89.0** | **75.8** |
|                | 2    | **73.5** | **96.6** | **90.3** | **78.2** |
| All            |     | 80.1  | 96.9 | 91.1 | 84.7 |

Table 3. Comparisons with SOTAs on few-shot action recognition.
## Comparison with Zero-shot SOTAs

| Method    | UCF* / UCF | HMDB* / HMDB | ANet*/ANet | Kinetics-600 |
|-----------|------------|--------------|------------|--------------|
| GA [38]   | 17.3±1.1 / - | 19.3±2.1 / - | -          | -            |
| TS-GCN [15]| 34.2±3.1 / - | 23.2±3.0 / - | -          | -            |
| E2E [3]   | 44.1 / 35.3 | 29.8 / 24.8  | 26.6 / 20.0 | -            |
| DASZL [27]| 48.9±5.8 / - | - / -        | -          | -            |
| ER [7]    | 51.8±2.9 / - | 35.3±4.6 / - | -          | 42.1±1.4     |
| ResT [32] | 58.7±3.3 / 46.7 | 41.1±3.7 / 34.4 | 32.5 / 26.3 | -            |
| **Ours**  | **85.8±3.3 / 79.6** | **58.1±5.7 / 49.8** | **84.6±1.4 / 77.4** | **68.9±1.0** |

Table 4. Comparisons with SOTAs on zero-shot video recognition. We directly evaluate our method without any additional training on cross-dataset video recognition. ANet is in short for ActivityNet. * means half classes evaluation.
Some Ablation Studies

### Comparisons with vision-only framework

| Paradigm            | Batch Gather | Textual Encoder | Top-1  | V100-days |
|---------------------|--------------|-----------------|--------|-----------|
| Vision-Only         | ✓            | online          | 81.2   | 6.7 (10*) |
| Vision-Text         | ✓            | offline         | 80.7   | 6.6       |
| Ours                | ✓            | offline         | 81.5   | 3.3       |

### Comparisons with contrastive-based framework

| Method               | Top-1  | FLOPs  | Params | Throughput |
|----------------------|--------|--------|--------|------------|
| ViViT-L/16-320 [1]    | 81.3   | 3992G  | 310.8M | 4.2 vid/s* |
| Ours ViT-B/32         | 78.5   | 23.7G  | 71.6M  | 322.5 vid/s|
| Ours ViT-B/16         | 81.5   | 90.3G  | 69.9M  | 126.5 vid/s|
| Ours ViT-L/14         | 85.4   | 415.4G | 230.4M | 35.5 vid/s |

Exploration of different frozen classifiers

Analysis on inference efficiency
Conclusion

• A simple yet effective transferring method from a **frozen classifier** perspective

• *Improving both the performance and the convergence speed of visual classification*

• *Superior performance on both general and zero-shot/few-shot recognition*

• *Codes & models have be available*  
  https://github.com/whwu95/Text4Vis
THANKS

🔥 Codes & Models
https://github.com/whwu95/Text4Vis

👩‍💋‍👦 Contact
Wenhao Wu
Email: whwu.ucas@gmail.com
Homepage: https://whwu95.github.io