Generative Model-Based Feature Knowledge Distillation for Action Recognition

Guiqin Wang[1,3], **Peng Zhao**[1,3*], **Yanjiang Shi**[1,3], **Cong Zhao**[2, 3], **Shusen Yang**[2, 3]

[1] School of Computer Science and Technology, Xi'an Jiaotong University
[2] School of Mathematics and Statistics, Xi'an Jiaotong University
[3] National Engineering Laboratory for Big Data Analytics, Xi'an Jiaotong University

Abstract

Knowledge distillation (KD), a technique widely employed in computer vision, has emerged as a de facto standard for improving the performance of small neural networks. However, prevailing KD-based approaches in video tasks primarily focus on designing loss functions and fusing cross-modal information. This overlooks the spatial-temporal feature semantics, resulting in limited advancements in model compression. Addressing this gap, our paper introduces an innovative knowledge distillation framework, with the generative model for training a lightweight student model. In particular, the framework is organized into two steps: the initial phase is Feature Representation, wherein a generative model-based attention module is trained to represent feature semantics; Subsequently, the Generative-based Feature Distillation phase encompasses both Generative Distillation and Attention Distillation, with the objective of transferring attention-based feature semantics with the generative model. The efficacy of our approach is demonstrated through comprehensive experiments on diverse popular datasets, proving considerable enhancements in video action recognition task. Moreover, the effectiveness of our proposed framework is validated in the context of more intricate video action detection task. Our code is available at https://github.com/aaai-24/Generative-based-KD.

Introduction

In recent years, various deep learning technologies have achieved significant success in the domain of intelligent video analysis (Foo et al. 2023; Yang et al. 2022a). Specifically, action recognition stands as a pivotal task within intelligent video analysis, entailing the categorization of action instances into corresponding labels. Recently, substantial enhancements have been achieved in the performance of action recognition (Sun et al. 2022). Intuitively, a larger model often corresponds to improved performance. This perspective has prompted numerous researchers to devise intricate backbone for capturing video-centric feature semantics(*e.g.*, C3D (Xu, Das, and Saenko 2017), I3D (Carreira and Zisserman 2017), S3D (Xie et al. 2018)). These efforts have yielded commendable results in the field of action recognition. However, the deployment of a larger backbone introduces extremely high resource and memory constraints, making it impractical for edge devices with limited resources. To address it, Hinton et al. proposed Knowledge Distillation(KD) as a solution, which facilitates the transfer of learned knowledge from a heavyweight model(teacher model) to a lightweight model(student model).

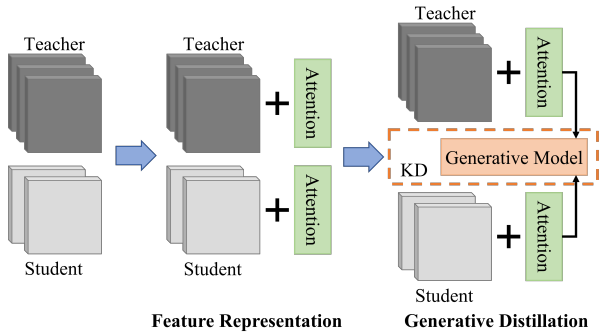

Figure 1: Our proposed framework includes two stages: Stage 1 adds an attention module to represent feature semantics; Stage 2 builds a generative-based KD module to distill feature knowledge from teacher model.

In accordance with the fundamental objective of model compression, KD methodologies primarily encompass two investigational paradigms: logits-based and feature-based. Logits-based KD (Hinton, Vinyals, and Dean 2015; Li et al. 2023) entails the condensation of a large teacher model into a more compact student counterpart, achieved through the conveyance of "dark knowledge" via soft labels generated from the teacher's output. Following the introduction of the approach by Romero et al. in 2015, comparative assessments have consistently validated the superiority of feature-based KD methodologies (Xu et al. 2020; Zhang and Ma 2020; Yang et al. 2022c) across a diverse array of tasks (Zhao et al. 2022). Consequently, scholarly attention has prominently shifted towards the extraction of knowledge from intricate features residing within intermediary layers of model.

However, the majority of feature-based KD methods (Zhao et al. 2022; Zhang and Ma 2020; Xu et al. 2020) are centered on the design of diverse loss functions to match feature maps, disregarding the inherent significance of vari-

*Corresponding author: Peng Zhao (p.zhao@xjtu.edu.cn)

ational feature semantics. In fact, particularly in the context of videos, features inherently contain intense semantic information (Quader et al. 2020), primarily stemming from both temporal and spatial variations. These feature semantics play a crucial role in enhancing the precision of the student model by transferring the variations of features from the teacher model. Regrettably, prevailing solutions often overlook such semantics, resulting in trivial improvements in the context of video tasks. Consequently, current KD methodologies predominantly cater to image-related tasks (Zhao et al. 2022; Yang et al. 2022b; Lin et al. 2022). Within the domain of video action recognition, the predominant focus of KD methodologies centers on cross-modal distillation, aimed at elevating the accuracy of models (Liu et al. 2021; Thoker and Gall 2019; Dai, Das, and Bremond 2021). Therefore, a noticeable research gap emerges in the realm of KD-based methods conducive for compressing 3D-CNN models, particularly within the context of video action analysis.

To address these issues, we proposed a novel generative model-based feature knowledge distillation framework for video action recognition. This framework enables the efficient transfer of feature semantics from heavyweight models (teacher model) to lightweight models (student model) between intermediate layers. Specifically, our proposed approach consists of two components. Firstly, we designed a Feature Representation module that leverages a generative-based attention model to acquire feature semantics within the 3D-CNN architecture. Secondly, we built a generative-based KD module, encompassing both Generative Distillation and Attention Distillation processes. This module is designed to distill attention-based feature information from the teacher model, as illustrated in Fig. 1. The Generative Distillation component is tailored to optimize the Feature Representation module of the student model by matching reconstructed features, thereby facilitating the learning of the generative-based attention model. Subsequently, the Attention Distillation process is orchestrated to transfer attention-based feature semantics by matching attention maps, conditioned on unchanged attention-based feature distributions.

To our best knowledge, we are the first to consider the temporal variation of feature semantics and study the generative model mechanism in KD. We design a generative model-based KD framework that effectively enhances the compression performance of 3D-CNN models. Our main contributions are summarized as follows:

- To distill temporal-spatial feature, we design a novel generative model-based attention module to represent feature semantics within the 3D-CNN architecture.
- We build a new framework that introduces the novel concept of utilizing a generative model for distilling attention-based feature. Particularly, our KD framework is the first to compress 3D-CNNs on video, with a generative model to transfer the temporal-spatial information.
- Based on extensive experiments, our approach demonstrates remarkable performance improvements across various network architectures on two prominent action recognition datasets. Additionally, we extend our framework to more complex task, action detection, which also provides considerable performance enhancements.

Related Work

Action Recognition

Most previous work of CNN architectures on action recognition can be categorized into two groups: 3D CNNs (Carreira and Zisserman 2017; Xu, Das, and Saenko 2017; Ji et al. 2012; Hara, Kataoka, and Satoh 2017) and partial 3D CNNs (Tran et al. 2018; Qiu, Yao, and Mei 2017; Xie et al. 2018). 3D CNNs were first proposed in (Carreira and Zisserman 2017; Tran et al. 2015), which consider a video as a stack of frames to learn spatiotemporal features of action by 3D convolution kernels. Furthermore, Carreira and Zisserman proposed I3D (Carreira and Zisserman 2017) to capture spatial and temporal information by fusing RGB and optical flow based on the 3D convolution. Partial 3D CNNs had been proposed in (Tran et al. 2018; Qiu, Yao, and Mei 2017), which replaced 3D convolutions with depth-wise separable convolutions to reduce resource cost. Meanwhile, Xie et al. replaced 3D convolutions with 2D convolutions to reduce computational complexity. However, 3D CNNs consume a substantial amount of resources to achieve high accuracy, which makes them unsuitable for deployment on resource-constrained devices. While the utilization of Partial 3D CNNs mitigates the resource demands, they still exhibit an accuracy gap when compared to full-fledged 3D CNNs.

Generative Model

Generative model (Goodfellow et al. 2014; Kingma and Welling 2014) develops rapidly in recent years by combining with deep learning. GAN (Goodfellow et al. 2014) maximises the approximate real data distribution information between a subset of the generating variables and the output of a recognition network. However, GAN learns distribution implicitly and lacks of sample diversity. VAE (Kingma and Welling 2014) approximates the real distribution by optimizing the variational lower bound on the marginal likelihood of data. However, VAE is not suitable to model the distribution of multi-modal output (Sohn, Lee, and Yan 2015). In order to transfer attention to guide knowledge distillation, we use conditional VAE (Sohn, Lee, and Yan 2015) to model the feature semantic distribution conditioned on attention value.

Knowledge Distillation

Knowledge distillation was first proposed in (Hinton, Vinyals, and Dean 2015), which transfers the output probability distributions via soft labels produced by teacher. Furthermore, Romero et al. proposed to distill the feature representation from penultimate layer, named feature knowledge distillation. Xu et al. proposed feature normalized knowledge distillation to reduce the impact of label noise. Additionally, Zagoruyko and Komodakis proposed attention knowledge distillation, which tries to match the attention map to transfer feature knowledge. Concurrently, there are efforts to integrate logits-based methodologies with feature-based approaches (Zhao et al. 2020; Shen et al. 2019), all with the goal of enhancing overall model performance.

However, these approaches primarily concentrate on aligning feature and attention maps, thereby potentially neglecting the underlying feature semantics.

In the context of knowledge distillation applied to action recognition, the majority of endeavors (Crasto et al. 2019; Stroud et al. 2020; Thoker and Gall 2019) predominantly focus on cross-modal distillation strategies. These strategies aim to enhance model performance by effectuating the transfer of optical flow (teacher model) insights to RGB flow (student model). However, the existing landscape of knowledge distillation-driven frameworks for action recognition is deficient in an established approach to model compression and the reduction of computational overhead.

Method

In this section, we first introduce the principle of feature KD in action recognition task. Then we present the framework of our KD framework and introduce the two sub-modules.

Definition

Feature Distillation The philosophy of knowledge distillation is to train the compact student model to approximate the capability of the cumbersome teacher model. In specific, suppose we have a pre-trained teacher model T and an untrained student model S, which are parameterized as neural networks in this work. Let's denote the output feature map $F \in R^{T\times C\times HW}$ of 3D-CNN, where T, C and HW represent time, channels, and spatial dimensions, severally. For better illustration, we denote F_T and F_S as the feature maps from the layer of the teacher and student model. For feature KD, the distillation distance between student model and teacher model is calculated by the two feature maps:

$$\mathcal{L}_{KD} = \frac{1}{n}\|f(F_T) - f(F_S)\|_2^2, \tag{1}$$

where $f(\cdot)$ is an explicit mapping function, n is the temporal dimension. The student model is encouraged to minimize the objective function $\mathcal{L}_{KD}$ to mimic the teacher model. Nonetheless, since the feature semantics is neglected in this fashion, the student is not capable of learning the temporal dependence from the teacher in action recognition.

Feature Representation For action recognition task, we first define an attention module to represent feature semantics, which also effectively improves recognition performance. We learn a generative model to optimize attention λ by leveraging feature F and action classes C. For simplifying the problem, we transform the optimization target using Bayes' theorem:

$$\begin{aligned}\max_{\lambda\in[0,1]} \log p(\lambda|F,C) &= \max_{\lambda\in[0,1]} \log p\left(C|F,\lambda\right) + \log p\left(F|\lambda\right) \\ &\quad + \log p\left(\lambda\right) - \log p\left(F,C,\lambda\right) \\ &\simeq \max_{\lambda\in[0,1]} \log p(C|F,\lambda) + \log p(\lambda|F),\end{aligned} \tag{2}$$

in the last step, we discard the constant term $\log p\left(F,C,\lambda\right)$ and set λ as a uniform distribution.

As Equ. 2 means, when we optimize the attention value, we not only use feature semantics and attention to improve the categories performance (the first term), but also ensure the attention-based feature distribution consistent with the original feature distribution (the second term).

Feature Knowledge Distillation

This work employs the feature knowledge distillation to help the student model learn the feature representation. We extract the semantics, expressed as attention map, from a trained teacher model and ask the student model to mimic it.

The pipeline of our KD framework is illustrated in Fig. 2, which includes two steps(*i.e.*, Generative Distillation and Attention Distillation). Given a video feature F, in the step 1, the teacher model leverages a pre-trained generative model, CVAE, to transfer the knowledge of feature and attention, which is to ensure the categories accuracy. In the step 2, the teacher would produce attention map λ, which represents the feature semantics of teacher model.

Generative Distillation In order to reconstruct the feature semantics, we introduce a *Generative Distillation* module by transferring attention-based feature from teacher model. Furthermore, conditional variational autoencoder(CVAE) effectively reconstructs the feature semantic by learning the correspondence between feature map and attention map. In specific, as Fig. 2(a) shows, we freeze the student model and the attention module, update the CVAE module by leveraging the CVAE of teacher model. For updating CVAE, we build an action-based generative problem:

$$p_\phi(F_t|\lambda_t) = \mathbb{E}_{p_\phi(z_t|\lambda_t)}[p_\phi(F_t|\lambda_t, z_t)], \tag{3}$$

where z_t is the latent variable, ϕ indicates the learnable model, $p_\phi(z_t|\lambda_t)$ indicates the prior model, and $p_\phi(F_t|\lambda_t, z_t)$ is the posterior model, which is the decoder procedure of the generative model. Notably, the latent variable z_t is sampled from the learned prior distribution, which is set as the process of feature reconstruction.

During the training procedure, our target is to let the student model learn the reconstructed feature from teacher model, which is aiming at fitting the distribution of the teacher model and improving the the reconstructed quality of the student model. In specific, we propose the generative loss $\mathcal{L}_{KD-gen}$ to optimize the feature reconstruction:

$$\mathcal{L}_{KD-gen} = \|f^T(F^T, A^T) - f^S(F^S, A^S)\|_2^2, \tag{4}$$

where S and T indicate the student model and teacher model, respectively, F represents the feature map, A is the attention map and $f(\cdot)$ is the attention-based reconstructed feature function. Remarkably, we set the same input, the feature map and attention map of student, in the CVAE module of teacher and student model. The same input is to allow student to focus on the attention-based feature reconstruction of the teacher, without being affected by the differences in feature and attention maps extracted by the different model.

During the CVAE training, as Fig. 2(a) shows, our target is to minimize the evidence lower bound(ELBO) loss $\mathcal{L}_{cvae}$:

$$\mathcal{L}_{cvae} = \log p_\psi(F_t|\lambda_t, z_t) + \alpha \cdot KL(q_\phi(z_t|F_t, \lambda_t)||p_\psi(z_t|\lambda_t)), \tag{5}$$

where z_t is the latent variable and $\alpha = 0.1$ is an empirical hyperparameter. In specific, as Fig. 3(a) shows, firstly, we

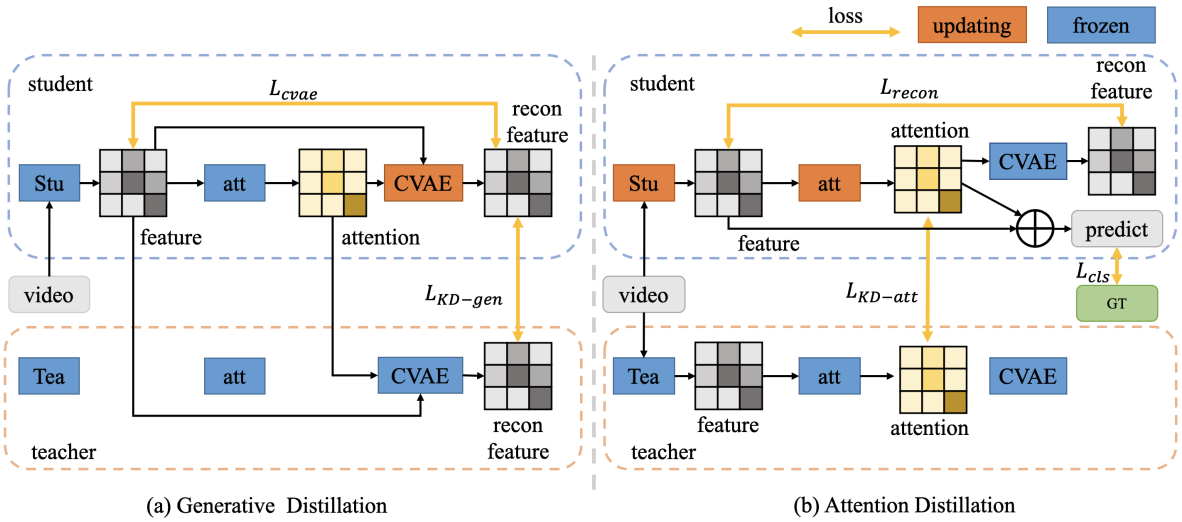

Figure 2: The pipeline of our KD framework. Our KD model is trained alternatively in two stages. In stage 1, the Generative Distillation is trained with self-reconstruction loss $\mathcal{L}_{CVAE}$ and generation distillation loss $\mathcal{L}_{KD-gen}$, which is to match reconstructed feature. In stage 2, the Attention Distillation is updated with representation loss $\mathcal{L}_{recon}$, attention distillation $\mathcal{L}_{KD-att}$ and classification loss $\mathcal{L}_{clf}$, which is to distill the attention-based feature semantics.

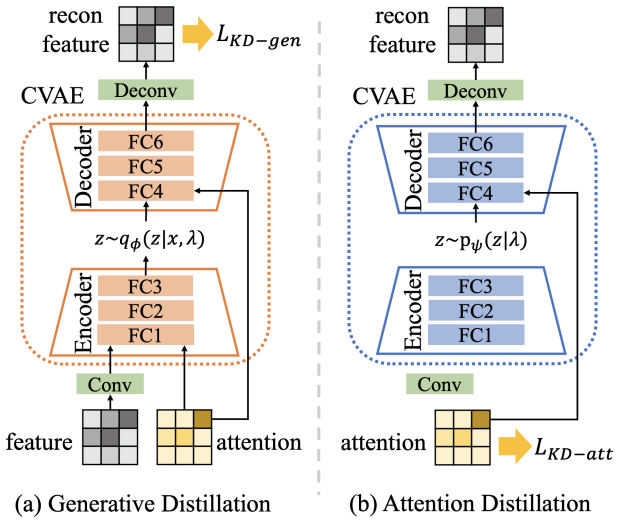

Figure 3: The architecture of our proposed KD framework. It includes the Generative Distillation module and the Attention Distillation module.

use Encoder to generate the intermediate variable z_t, with function f_z by the feature F_t and the attention λ_t, as follows:

$$z = f_z\left(f_{fc_i}\left(\lambda_i, \theta_i\left(f_i\right)\right)\right), \tag{6}$$

where f_{fc_i} is the liner model, and θ_i is a $1 \times 1 \times 1$ 3D convolution, aiming at concatenating λ_i for f_z by reducing the channels' number and adding learnable model parameters. f_z is a mean and variance function for the intermediate variable z, which generates the latent variable Z. Then, we use Decoder to generate the reconstructed feature map $\widehat{F}$, which leverages latent variable Z and attention λ , as follows:

$$\widehat{F} = \theta_i'\left(f_{fc_i}\left(Z, \lambda_i\right)\right), \tag{7}$$

where θ_i' is a $1 \times n \times n$(n is the kernel size) 3D deconvolution, which is aiming at generating the reconstructed feature map $\widehat{F}$ by adding the number of channels. We use $\mathcal{L}_{cvae}$ to measure the difference between $\widehat{F}$ and F.

Attention Distillation As Equ. 2 shows, the Generative Distillation mainly optimizes the first term $\log p(C|F,\lambda)$, this section mainly focuses on the second term $\log p(\lambda|F)$. The attention represents the feature distribution of the video, so the student model can learn the representation ability of the teacher model by learning the attention map (Quader et al. 2020; Zagoruyko and Komodakis 2017). Specifically, as Fig. 2 (b) shown, we freeze the CVAE, update the student model and attention module by leveraging attention distillation. For the video feature F, we use the attention A to represent feature semantics, as follows

$$A = Sigmod(GN(f_{Conv1D}(F))), \tag{8}$$

where GN indicates the group normalization. Furthermore, we calculate the attention-based feature F' as follows:

$$F' = \sigma \times (A \times \theta(F)), \tag{9}$$

where $\theta(\cdot)$ indicates the 3D transposed convolution, σ is the scaling factor, which is defined as:

$$\sigma = \frac{\|F\|}{\|A \times \theta(F)\|}. \tag{10}$$

As Fig. 2 (b) shows, for distilling the attention, we propose the attention distillation loss L_{KD-att}, defined as:

$$L_{KD-att} = \|A^T - A^S\|_2^2, \tag{11}$$

where A^T and A^S indicate the attention map of the teacher model and the student model, respectively.

During the training, for action recognition, we encourage higher capability of the action classification. This amounts to minimizing the following loss:

$$\mathcal{L}_{clf} = \sum_{c=1}^{C+1} -y_c(x) \log(p_c(x)), \tag{12}$$

where $y_c(x)$ and $p_c(x)$ indicate action probability distribution of the ground truth and the predicted result respectively.

In the meanwhile, as shown in Fig. 3(b), we use Decoder to generate reconstructed feature map, which is aiming at maintaining the distribution of attention-based feature during the feature representation. This amounts to minimizing the reconstruction loss $\mathcal{L}_{recon}$:

$$\begin{aligned} \mathcal{L}_{recon} &= -\log(\sum_{l=1}^{L} p_\psi\left(f_t|\lambda_t, z_t\right)) \\ &\backsimeq -\sum_{t=1}^{T} \log\left\{\frac{1}{L}\sum_{l=1}^{L} p_\psi\left(f_t|\lambda_t, z_t^{(l)}\right)\right\} \\ &\backsimeq \|f_t - f_\psi\left(\lambda_t, z_t\right)\|^2, \end{aligned} \tag{13}$$

where $z_t^{(l)}$ is generated by z_t and λ_t in the encoder of generative model. Especially, following (Shi et al. 2020), in the last step, we set L as 1.

Training and Inference In our proposed framework, attention module requires attention-based feature as the input. Therefore, as shown in Fig. 2, we consider alternating the training of the attention module with the backbone module.

As Fig. 2(a) shows, in Stage 1, we freeze the student backbone and activate the CVAE module, which is to optimize the generative distillation procedure by the features extracted from the backbone and the generated attention. The overall loss function of the stage 1 is as follows:

$$\mathcal{L}_{GD} = \mathcal{L}_{CVAE} + \beta \cdot \mathcal{L}_{KD-gen}, \tag{14}$$

where β is an empirical hyperparameter, $\beta = 0.01$.

As Fig. 2(b) shows, in Stage 2, we freeze the CVAE module and activate the student backbone, which is to optimize the attention distillation procedure by the attention-based reconstructed feature, the attention distribution and the predicted action probability, as follows:

$$\mathcal{L}_{AD} = \mathcal{L}_{recon} + \mathcal{L}_{clf} + \gamma \cdot \mathcal{L}_{KD-att}, \tag{15}$$

where γ is an empirical hyperparameter, $\gamma = 0.1$. Notably, during the inference procedure, we only exploit student model with attention module to predict action label, not including generative model.

Experiments

In this section, we first describe datasets and evaluation metrics. Then, we evaluate our model's effectiveness followed by main result and ablation study.

Datasets and Evaluation Metrics

To validate the effectiveness of our model, we conduct extensive experiments on commonly-used action recognition benchmark UCF101 (Soomro, Zamir, and Shah 2012) and HMDB51 (Kuehne et al. 2011), commonly-used action detection benchmark THUMOS14 (Jiang et al. 2014).

UCF101 It consists of 13320 action videos, including 101 action categories, which has 3 official splits and each split divides the training set and test set at a ratio of 7:3.

HMDB51 It consists of 6849 video clips, which contains 51 action categories and each category includes at least 101 video clips. It has the same split ratio with UCF101 dataset.

THUMOS14 It contains 101 categories of videos and is composed of four parts: training, validation, testing and background set. Each set includes 13320, 1010, 1574 and 2500 videos, respectively. Following the common setting (Jiang et al. 2014), we used 200 videos in the validation set for training, 213 videos in the testing set for evaluation.

Evaluation Metrics We follow the standard evaluation protocol and report accuracy as evaluation metric. In Specific, we report the Top-1 and Top-5 on action recognition. We report the mean Average Precision(mAP) at the different intersections over union(IoU) thresholds on action detection.

Implementation Details

On UCF101 and HMDB51, we use tvl1 (Brox, Bregler, and Malik 2009) to extract optical frames. The length of the clip is set to 64. We resize the frame to 256 for UCF101 and 240 for HMDB51. We use SGD optimizer for student model and Adam optimizer for KD module. On THUMOS14, we sample RGB and optical flow at 10 fps and split video into clips of 256 frames. Adjacent clips have a temporal overlap of 30 frames during training and 128 frames during testing. The size of frame is set to 96×96.

We adopt offline distillation strategy and transferred knowledge by alternately training generative distillation module and attention distillation module.

Main Results

To verify the effectiveness of our method, we compare our method with other distillation methods on the action recognition and the action detection task. In the Table 1, we adopt the I3D (Carreira and Zisserman 2017) as the teacher model and the Top-I3D (Xie et al. 2018) as the student model. In the Table 2, we adopt the AFSD (Lin et al. 2021)as the teacher model, which is the CNN-based sate-of-the-art method on the action detection, and the Top-I3D-based AFSD as the student model. The Top-I3D replaces partial 3D convolutional blocks of the I3D network with 2D convolutional blocks, which reduces the parameters and computation cost.

As Table 1 shows, for the different datasets of action recognition, our method outperforms the previous KD methods and achieves significant improvement. Specifically, on the UCF101, based on the student model, our method gains a significant increase of 2.5% and 2.4% on the Top-1 and the

Model	Knowledge	UCF101		HMDB51		FLOPS(G)
		Top-1	Top-5	Top-1	Top-5	
Teacher	-	91.9	98.8	69.0	88.8	111.3
Student	-	64.1	82.1	52.0	77.6	45.5
KD (Hinton, Vinyals, and Dean 2015)	Logits	65.2	79.2	53.1	78.2	45.5
CTKD (Li et al. 2023)	Logits	65.6	83.5	52.6	78.1	45.5
FN (Xu et al. 2020)	Feature	65.5	79.4	52.9	77.3	45.5
MGD (Yang et al. 2022c)	Feature	65.4	79.9	54.1	77.9	45.5
SimKD (Chen et al. 2022)	Feature	66.1	81.0	53.6	78.9	45.5
AT (Zagoruyko and Komodakis 2017)	Attention	66.2	80.5	52.9	77.1	47.4
CTKD (Zhao et al. 2020)	Logits+Feature	65.7	80.1	53.9	78.8	45.5
Ours	Feature	**66.6**	**84.5**	**54.5**	**79.0**	47.4

Table 1: Validation accuracy and computation cost on UCF101 and HMDB51. We set I3D as the teacher model, Top-I3D as the student model. For fair comparison, we keep the same training configuration for all methods.

Top-5 accuracy, which finally reaches 66.6% and 84.5%, respectively. On the more complex dataset HMDB51, based on the student model, our method also achieves 2.5% and 1.4% increase, which reaches 54.5% and 79.0%.

For fairness, we adopt the same backbone (I3D-based teacher and Top-I3D-based student) with other knowledge distillation methods. In particular, our approach outperforms other feature-based KD methods (*e.g.*, FN (Xu et al. 2020), SimKD (Chen et al. 2022)), which also utilize the feature distillation to transfer knowledge but without modeling critical area. Furthermore, our approach outperforms AT (Zagoruyko and Komodakis 2017), which also utilizes the attention distillation but without explicit feature semantics. The results demonstrate the superior performance of our approach with attention-based feature semantics modeling.

In addition to experimenting on different datasets, we expanded our knowledge distillation method on different task. We conduct experiment on the Thumos14 dataset for action detection task and the comparison results are summarized in Table 2. Again, based on the student model, our proposed method obtains a significant improvement of 1.1% average mAP, surpassing the other works(*e.g.* KD (Hinton, Vinyals, and Dean 2015)). The consistent superior results on different tasks justify the effectiveness of our proposed method. As shown in Table 2, on the more complex task, action detection, our method achieves improvement in all tIOU $[0.1 : 0.1 : 0.5]$ and reaches the 33.7% average mAP.

In contrast to the teacher model, as shown in Tables 1 and 2, the student model demonstrates a discernible decrease in accuracy across different video tasks. Nonetheless, it is noteworthy that the computational expense associated with the student amounts to merely approximately 40.9% of that of the teacher. Concurrently, with the help of our framework, the student model gains considerable improvement, with a marginal increment in computational overhead.

Ablation Study

To demonstrate the reasonableness of our framework, we analyze the effect of each functions in this subsection.

Contribution of each sub-module We study each sub-module influence on the overall performance of action recognition. We start from the baseline which only includes classification loss. Next we add feature representation loss L_{recon}, note that it involves both attention module and attention-based feature representation module. We then introduce the generative distillation loss $L_{KD-cvae}$ and attention distillation loss L_{KD-att}, respectively.

Table 3 summarizes the performances by considering one more factor at each stage on UCF101. We first observe that adding the feature representation loss L_{recon} and the KD loss L_{KD-gen},L_{KD-att} largely enhance the performance of the action recognition model. The L_{recon} is aiming at generating attention to represent feature semantics, which also improves the ability of feature representation. The two distillation losses encourage the feature learning in the student model by transferring the attention knowledge and the feature knowledge. As shown in Table 3, based on the baseline, our feature representation contributes an increase 1.1% , 1.2% and reaches 65.2%, 83.3% on the Top-1 and Top-5 respectively. Based on this, our two distillation losses brings 2.1%, 1.9% and 2.0%, 1.4% on the Top-1 and Top-5 respectively, which finally reach 66.2%, 84.0% and 66.1%, 83.5%. When introducing both distillation losses simultaneously, compared with the baseline, our proposed method improve 2.5% and 2.4% on the Top-1 and Top-5 accuracy.

Effectiveness of attention module For transferring the feature semantics, We distill the attention by representing spatial-temporal feature. In specific, during the procedure of generating the attention, we leverage the generative model CVAE to optimize the attention module, which is aiming at improving the accuracy of the model and learning the feature information. To see this, we conduct the experiment by directly adding our attention-based feature representation module into different model, I3D and Top-I3D.

As shown in Table 4, compared to the I3D, our attention representation module contributes an increase of 0.5%, 0.3% and the performance finally reaches 92.4%, 99.1% on the Top1 and the Top5. Compared to the Top-I3D model, our attention representation module contributes an increase

mAP(%)/tIOU	0.1	0.2	0.3	0.4	0.5	AVG	FLOPS(G)
Teacher	71.8	70.6	67.3	62.4	55.5	65.5	84.4
Student	37.9	36.7	33.1	30.1	25.2	32.6	36.0
KD (Hinton, Vinyals, and Dean 2015)	38.5	37.0	34.3	31.0	24.6	33.1	36.0
SimKD (Chen et al. 2022)	38.5	37.1	34.6	30.9	25.1	33.2	36.0
Ours	**39.4**	**37.5**	**34.9**	**31.5**	**25.3**	**33.7**	39.9

Table 2: Validation accuracy and computation cost on Thumos14. The teacher model is AFSD, student model is Top-I3D-based AFSD. We keep the same configuration for all methods.

baseline	L_{recon}	$L_{KD-cvae}$	L_{KD-att}	Top-1	Top-5
✓				64.1	82.1
✓	✓			65.2	83.3
✓	✓	✓		66.2	84.0
✓	✓		✓	66.1	83.5
✓	✓	✓	✓	**66.6**	**84.5**

Table 3: Performance of each sub-module on UCF101.

Model	Top-1	Top-5
I3D	91.9	98.8
I3D+Att	**92.4**	**99.1**
Top-I3D	64.1	82.1
Top-I3D+Att	**64.7**	**82.9**

Table 4: Performance of attention-based semantic representation module on different student models on UCF101.

of 0.6%, 0.8% and the performance finally reaches 64.7%, 82.9% on the Top1 and the Top-5. Both results show that adding our attention representation module can represent the feature semantics and improve the model performance.

Generalization of our method To verify the generalization of our proposed KD framework, we expand the experiment on different student model (Xie et al. 2018)(*i.e.*, Top-I3D, Bottom-I3D and I2D). As shown in Table 5, our method is effective to all different student models. Compared to the Top-I3D model, our KD framework contributes an increase of 2.1%, 2.8% and the performance finally reaches 66.2%, 84.9% on the Top1 and the Top5. Compared to the Bottom-I3D model, our KD framework contributes an increase of 0.6%, 0.5% and the performance finally reaches 53.3%, 72.9% on the Top1 and the Top5. In particular, in the experiment with I2D, which is a 2D-based convolutional neural network, our method boosts the Top-1 accuracy by 6.4%, the Top-5 accuracy by 5.2% and the performance finally reaches 58.6%, 76.0%, which even achieves comparable performance with 3D convolutional neural network.

Furthermore, we present a comparative evaluation of our method against the feature-based approach proposed by FN (Xu et al. 2020), thereby demonstrating the enhanced advancements and efficiency achieved by ours. As Fig. 4 shows, our method achieves better performance improvement (1.4% vs 2.1% on Top-I3D, 0.2% vs 0.6% on Bottom-

Model	Top-1	Top-5
Top-I3D	64.1	82.1
Top-I3D+Ours	**66.2**	**84.9**
Bottom-I3D	52.7	72.7
Bottom-I3D+Ours	**53.3**	**72.9**
I2D	52.2	70.8
I2D+Ours	**58.6**	**76.0**

Table 5: Performance of our proposed framework on different student model on UCF101.

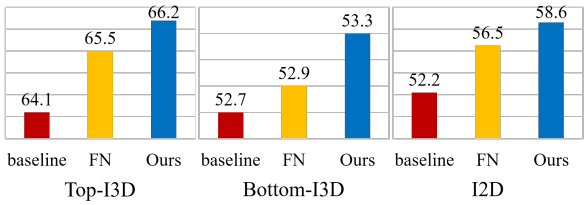

Figure 4: Accuracy of our proposed framework on different student models with FN on UCF101.

I3D, 4.3% vs 6.4% on I2D) on different student models. Significantly, our method demonstrates a noteworthy augmentation of 6.4% on the I2D, leading to a final performance level of 58.6%, which proves the efficacy of our approach in distilling knowledge from 3D to 2D-CNNs.

Conclusions

In this paper, we propose a novel generative model-based knowledge distillation framework to transfer video feature semantics through attention representation. Our framework mainly encompasses two steps: Feature Representation and Generative-based KD(Feature Distillation and Attention Distillation). By leveraging the attention mechanisms, we capture feature semantics and effectuate KD through generative modeling. In the realm of 3D-CNNs distillation, our method exhibits a remarkable performance advancement over existing approaches across video action area. The results demonstrate the efficacy of our approach in enhancing KD performance via generative model-based feature distillation. Consequently, this study establishes the groundwork for an innovative KD framework with a specific focus on 3D model distillation.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grants 2021YFB2401300, 2022YFA1004100, and 2020YFA0713900; and in part by the National Natural Science Foundation of China under Grants 62172329, U1811461, U21A6005, and 11690011

References

Brox, T.; Bregler, C.; and Malik, J. 2009. Large displacement optical flow. In *CVPR*.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*.

Chen, D.; Mei, J.-P.; Zhang, H.; Wang, C.; Feng, Y.; and Chen, C. 2022. Knowledge distillation with the reused teacher classifier. In *CVPR*.

Crasto, N.; Weinzaepfel, P.; Alahari, K.; and Schmid, C. 2019. Mars: Motion-augmented rgb stream for action recognition. In *CVPR*.

Dai, R.; Das, S.; and Bremond, F. 2021. Learning an augmented rgb representation with cross-modal knowledge distillation for action detection. In *ICCV*.

Foo, L. G.; Gong, J.; Fan, Z.; and Liu, J. 2023. System-status-aware Adaptive Network for Online Streaming Video Understanding. In *CVPR*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. *NeurIPS*.

Hara, K.; Kataoka, H.; and Satoh, Y. 2017. Learning spatio-temporal features with 3d residual networks for action recognition. In *ICCV*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Ji, S.; Xu, W.; Yang, M.; and Yu, K. 2012. 3D convolutional neural networks for human action recognition. *TPAMI*.

Jiang, Y.-G.; Liu, J.; Roshan Zamir, A.; Toderici, G.; Laptev, I.; Shah, M.; and Sukthankar, R. 2014. THUMOS Challenge: Action Recognition with a Large Number of Classes. http://crcv.ucf.edu/THUMOS14/.

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *ICLR*.

Kuehne, H.; Jhuang, H.; Garrote, E.; Poggio, T.; and Serre, T. 2011. HMDB: a large video database for human motion recognition. In *ICCV*.

Li, Z.; Li, X.; Yang, L.; Zhao, B.; Song, R.; Luo, L.; Li, J.; and Yang, J. 2023. Curriculum temperature for knowledge distillation. In *AAAI*.

Lin, C.; Xu, C.; Luo, D.; Wang, Y.; Tai, Y.; Wang, C.; Li, J.; Huang, F.; and Fu, Y. 2021. Learning salient boundary feature for anchor-free temporal action localization. In *CVPR*.

Lin, S.; Xie, H.; Wang, B.; Yu, K.; Chang, X.; Liang, X.; and Wang, G. 2022. Knowledge Distillation via the Target-Aware Transformer. In *CVPR*.

Liu, Y.; Wang, K.; Li, G.; and Lin, L. 2021. Semantics-aware adaptive knowledge distillation for sensor-to-vision action recognition. *TIP*.

Qiu, Z.; Yao, T.; and Mei, T. 2017. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*.

Quader, N.; Bhuiyan, M. M. I.; Lu, J.; Dai, P.; and Li, W. 2020. Weight excitation: Built-in attention mechanisms in convolutional neural networks. In *ECCV*.

Romero, A.; Ballas, N.; Kahou, S. E.; Chassang, A.; Gatta, C.; and Bengio, Y. 2015. FitNets: Hints for Thin Deep Nets. In *ICLR*.

Shen, C.; Xue, M.; Wang, X.; Song, J.; Sun, L.; and Song, M. 2019. Customizing student networks from heterogeneous teachers via adaptive knowledge amalgamation. In *ICCV*.

Shi, B.; Dai, Q.; Mu, Y.; and Wang, J. 2020. Weakly-supervised action localization by generative attention modeling. In *CVPR*.

Sohn, K.; Lee, H.; and Yan, X. 2015. Learning structured output representation using deep conditional generative models. *NeurIPS*.

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Stroud, J.; Ross, D.; Sun, C.; Deng, J.; and Sukthankar, R. 2020. D3d: Distilled 3d networks for video action recognition. In *WACV*.

Sun, Z.; Ke, Q.; Rahmani, H.; Bennamoun, M.; Wang, G.; and Liu, J. 2022. Human action recognition from various data modalities: A review. *TPAMI*.

Thoker, F. M.; and Gall, J. 2019. Cross-modal knowledge distillation for action recognition. In *ICIP*.

Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*.

Tran, D.; Wang, H.; Torresani, L.; Ray, J.; LeCun, Y.; and Paluri, M. 2018. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*.

Xie, S.; Sun, C.; Huang, J.; Tu, Z.; and Murphy, K. 2018. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *ECCV*.

Xu, H.; Das, A.; and Saenko, K. 2017. R-c3d: Region convolutional 3d network for temporal activity detection. In *ICCV*.

Xu, K.; Rui, L.; Li, Y.; and Gu, L. 2020. Feature normalized knowledge distillation for image classification. In *ECCV*.

Yang, S.; Zhang, L.; Xu, C.; Yu, H.; Fan, J.; and Xu, Z. 2022a. Massive data clustering by multi-scale psychological observations. *National Science Review*.

Yang, Z.; Li, Z.; Jiang, X.; Gong, Y.; Yuan, Z.; Zhao, D.; and Yuan, C. 2022b. Focal and global knowledge distillation for detectors. In *CVPR*.

Yang, Z.; Li, Z.; Shao, M.; Shi, D.; Yuan, Z.; and Yuan, C. 2022c. Masked generative distillation. In *ECCV*.

Zagoruyko, S.; and Komodakis, N. 2017. Paying More Attention to Attention: Improving the Performance of Convolutional Neural Networks via Attention Transfer. In *ICLR*.

Zhang, L.; and Ma, K. 2020. Improve object detection with feature-based knowledge distillation: Towards accurate and efficient detectors. In *ICLR*.

Zhao, B.; Cui, Q.; Song, R.; Qiu, Y.; and Liang, J. 2022. Decoupled knowledge distillation. In *CVPR*.

Zhao, H.; Sun, X.; Dong, J.; Chen, C.; and Dong, Z. 2020. Highlight every step: Knowledge distillation via collaborative teaching. *IEEE Transactions on Cybernetics*.