Learning Temporally Invariant and Localizable Features via Data Augmentation for Video Recognition

Anonymous ECCV submission

Paper ID 0000

Abstract. Deep-Learning based video recognition has shown promising improvements along with the development of large-scale datasets and spatio-temporal network architectures. In image recognition, learning spatially invariant features is a key factor for improving recognition performance and robustness. Data augmentation based on visual inductive priors such as crop, flip, rotation, or photometric jittering is a representative approach to achieve these features. Recent state-of-the-art recognition solutions are relied on modern data augmentation strategies that exploit mixture of augmentation operations. In this study, we extend these strategies to the temporal dimension for videos to learn temporally invariant, or temporally localizable features to cover temporal perturbations, or complex actions in videos. Based on our novel temporal data augmentation algorithms, video recognition performances are improved in a limited amount of training data, compared to spatial-only data augmentation algorithms, including the 1st Visual Inductive Priors (VIPriors) for data-efficient action recognition challenge. Furthermore, learned features are temporally localizable that cannot be achieved from the spatial augmentation algorithms.

1 Introduction

A lot of augmentation techniques have been proposed to increase recognition performance and robustness for an environment of limited training data, or to prevent overconfidence and overfitting of large-scale data such as ImageNet [23]. These techniques can be categorized into data-level augmentation [24, 33, 8, 29, 9, 18, 10, 34], data-level mixing [52, 50, 49, 27, 42, 26], and in-network augmentation [37, 13, 19, 12, 48, 21, 41]. Data augmentation is an important component for recent state-of-the-art self-supervised learning [16, 4, 31], semi-supervised learning [45, 2, 1, 35], self-learning [46], and generative models [51, 53, 54, 20] because of its ability to learn invariant features.

Purpose of data augmentation in image recognition is to enhance generalization ability via learning spatially invariant features. Augmentations such as geometric (crop, flip, rotation, *etc.*) and photometric (brightness, contrast, color, *etc.*) transformations can model uncertain variances in a dataset. Recent algorithms have shown state-of-the-art performances in terms of complexity-accuracy

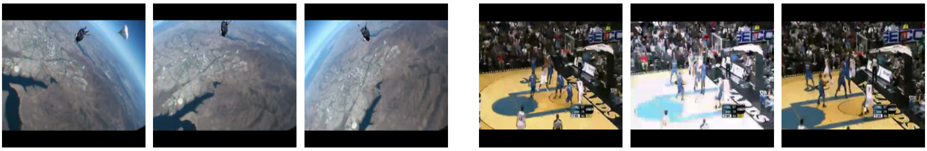


Fig. 1: Example clips of temporal perturbations. *Left*: Geometric perturbation across frames in sky-diving video due to extreme camera and object movements. *Right*: Photometric perturbation across frames in the basketball stadium due to camera flashes.

tradeoff [29, 9], or robustness [17, 18]. Some approaches [50, 49] learn localizable features that can be used as transferable features to the localization-related tasks such as object detection and image captioning. They learn simultaneously what to and where to focus on for the recognition.

Despite evolving through numerous algorithms in image recognition, there are little explorations of data augmentation and regularization in video recognition. In videos, temporal variations and perturbations should be considered as well as spatial ones. For example, Fig. 1 depicts temporal perturbations across frames in a video. This perturbation can be one of the geometric perturbations such as translation, rotation, scale, *etc.*, or the photometric perturbations such as brightness, contrast, *etc.*. To handle these perturbations, not only well-studied spatial augmentations but also temporally varying data augmentations should be considered generally.

In this paper, we propose several extensions toward temporal robustness. More specifically, temporally invariant and localizable features can be modeled via data augmentations. We extend such examples of well-studied recent spatial augmentation techniques: data-level augmentation and data-level mixing. To the best of our knowledge, it is very first study that deeply analyzes temporal perturbation modeling via data augmentation in video recognition.

Contribution of this paper is summarized as follows:

- We propose an extension of RandAugment [9], called RandAugment-T, as data-level augmentation for video recognition. It can model temporally varying level of augmentation operations.
- We propose temporal extensions of CutOut [10], MixUp [52], and CutMix [50] as examples of deleting, blending, and mixing data samples. Considering temporal dimension improves recognition performances and temporal localization abilities.
- Recognition results of proposed extensions on UCF-101 [36] subset for 1st Visual Inductive Priors (VIPriors) for data-efficient action recognition challenge, and HMDB-51 [25] dataset show performance improvements compared to spatial-only versions in a simple baseline.

2 Related Works

2.1 Data augmentation

Data-level augmentation In the beginning, to enlarge the generalization performance of a dataset and to reduce overfitting problem of preliminary networks, various data augmentation methods such as rotate, flip, crop, color jitter [23], and scale jittering [33] are proposed. CutOut [10] deletes a square-shaped box at random location to encourage the networks focus on various properties of image, not rely on the most discriminative regions. Hide-and-Seek [34] is a similar approach, but it deletes multiple regions that are sampled from the grid patches.

Recently, the methodology of combining more than one augmentation operations has been proposed. Cubuk *et al.* [8] propose a reinforcement learning-based approach to search the optimal data augmentation policy in a given dataset. However, since the search space is too large, it requires extensive time to find the optimal policy. Although an approach to mitigate this problem is proposed [29], it is still hard and time-consuming to get to the optimal augmentation strategy. To solve this, Cubuk *et al.* [9] propose RandAugment that randomly sample augmentation operations from the candidate list and cascade them. Hendrycks *et al.* [18] propose an approach called AugMix that blend images parallelly that are augmented by the operations sampled from set of candidates like RandAugment.

These techniques can model uncertain spatial perturbations such as geometric transform, photometric transform, and both of them. Since researches have focused on static images, applying these approaches into videos is a straightforward extension.

Data-level mixing Together with data augmentation algorithms, augmentation strategies using multiple samples have been proposed. Zhang *et al.* [52] propose an approach to manipulate images with more than one image, called MixUp. They make a new sample by blending two arbitrary images and interpolate their one hot ground-truth labels. This encourages the model to behave linearly in-between training examples. CutMix [50] combines the concepts of CutOut and MixUp, taking the best of both worlds. It replaces square-shaped deleted region in CutOut with a patch from another image. This encourages the model to learn not only what to recognize, but also where to recognize. It can be interpreted as spatially localizable feature learning. Inspired by CutMix, several methods to increase the generality have been proposed. CutBlur [49] proposed CutMix-like approach to solve the restoration problem using mixing between low-resolution and high-resolution images. They also proposed CutMixUp that is a combination of MixUp and CutMix. CutMixUp blends the two images in the one of the mask of CutMix to relax extreme changes in boundary pixels. Attribute Mix [27] uses the masks of any shape, not only squre-shaped mask. Attentive CutMix [42] also discards the square-shaped mask. It uses multiple patches sampled from the grid, and replaces the regions with another image. Smoothmix [26] focus on the 'strong edge' problem caused by the boundary of masks.

Although numerous data manipulation methods including deleting, blending, and mixing, successfully augment many image datasets, their ability when applied to video recognition to learn temporally invariant and localizable features, is not explored yet.

In-network augmentation Apart from the data-level approaches, several researches have proposed in-network augmentation algorithms. They usually design stochastic networks to augment in the feature-level in order to reduce predictive variance and to learn more high-level augmented features rather than to learn features from the low-level augmentations. Dropout [37] is a very first approach to regularize the overfitted models. Other approaches such as DropBlock [13], Stochastic depth [19], Shake-Shake [12] and ShakeDrop [48] regularizations have been proposed. Manifold-MixUp [41] propose mixing strategy like MixUp, but in the feature space. The most similar approach with this study is a regularization method for video recognition, called Random Mean Scaling [21]. It randomly adjusts spatio-temporal feature in the video networks. In contrast, our approaches focus on data-level manipulations and extending from the spatial-only algorithms into temporal worlds.

2.2 Video recognition

For video action recognition, like image recognition area, various architectures have been proposed to capture spatio-temporal features from videos. In [39], Tran *et al.* proposed C3D that extracts features containing objects, scenes and action information through 3D convolutional layers, and then simply passed through a linear classifier. In [40], a (2+1)D convolution that focuses on layer factorization rather than 3D convolution is proposed, which is composed with 2D spatial convolution followed by 1D temporal convolution. In addition, non-local block [44] and GloRe module [6] are suggested to capture long range dependencies via self-attention and graph-based modules. By plugging them into 3D ConvNet, the network can learn long distance relations in both space and time. Another approach is two stream architectures [43, 38, 32]. In [3], a two-stream 3D ConvNet inflated from deep image classification network and pre-trained features is proposed and achieves state-of-the-art performance by pre-training it with Kinetics dataset, a large-scale action recognition dataset. Based on this architecture, Xie *et al.* [47] combined a top-heavy model design, temporally separable convolution, and spatio-temporal feature gating blocks to make low-cost and meaningful features. Recently, SlowFast [11] networks that consists of a slow path for semantic information and a fast path for rapidly changing motion information, show competitive performance with different frame rate sampling strategy. In addition to this, RESOUND [28] proposed a method to reduce the static bias of the dataset, an Octave convolution [5] is proposed to reduce spatial redundancy by dividing the frequency of features, and debiasing loss function [7] is proposed to mitigate the strong scene bias of the networks and focus on the actual action information.

Since the advent of the large-scale Kinetics dataset, most action recognition studies have pre-trained the backbone on Kinetics, which guarantees basic performances. However, based on the results of [15], architectures with large numbers of parameters are significantly overfitted when learning from the scratch on relatively small dataset such as UCF-101 [36] and HMDB-51 [25]. It indicates that training without a pre-trained backbone is a challenging issue. Compared to existing researches that have been focused on novel dataset and architectures, we focus on the regularization techniques such as data augmentation, to prevent overfitting via learning invariance and robustness in terms of spatially and temporally.

3 Methods

3.1 Data-level temporal data augmentations

First, we extend existing RandAugment [9] for video recognition. RandAugment has two hyper-parameters to optimize. One is the number of augmentation operation to apply, N, and the other is the magnitude of the operation, M. Grid search of these two parameters in a given dataset produces state-of-the-art performances in image recognition.

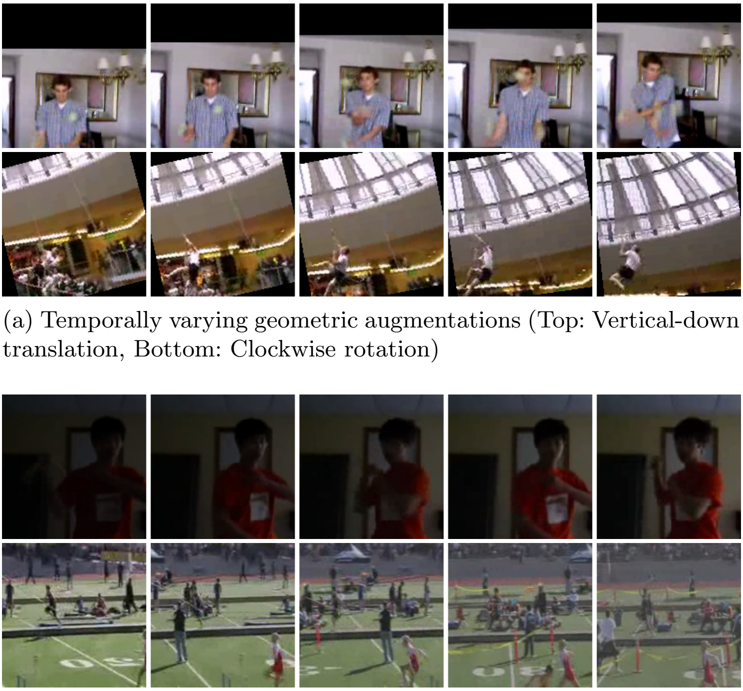
```
def randaugment_T(X, N, M1, M2):
"""Generate a set of distortions.

Args:
X: Input video (T x H x W)
N: Number of augmentation transformations
to apply sequentially.
M1, M2: Magnitudes for both temporal ends.
"""

ops = np.random.choice(transforms, N)
M = np.linspace(M1, M2, T)
return [[op(X, M[t]) for t in range(T)]
     for op in ops]
```

Fig. 2: Python code for RandAugment-T based on numpy.

For video recognition, RandAugment is directly applicable to every frame of an video, however, this limits temporal perturbation modeling. To cover temporally varying transformations, we propose RandAugment-T that linearly interpolates between two magnitudes from the first frame to the last frame in a video clip. Pseudo-code of RandAugment-T is described in Fig. 2. It receives three hyper-paremeters: N, M1, and M2. N is the number of operations, which is same as RangAugment. M1 and M2 indicate magnitudes for both temporal ends, which can be any combination of levels. Set of augmentation operations (`transforms` in Fig. 2) is identical with RandAugment. However, `rotate`, `shear-x`, `shear-y`, `translate-x`, and `translate-y` can model temporally varying geometric transformations such as camera movement or object movement (Fig. 3(a)), and `solarize`, `color`, `posterize`, `contrast`, `brightness`, and `sharpness` can model photometric transformations such as brightness or contrast change due to auto-shot mode in a camera (Fig. 3(b)). Remained operations (`identity`, `autoContrast`, and `equalize`) have no magnitudes that are applied to evenly across frames.

3.2 Data-level temporal deleting, blending, and mixing

Regularization techniques for image recognition such as CutOut [10], MixUp [52], and CutMix [50] can be applied identically across frames in a video. CutMixUp



(a) Temporally varying geometric augmentations (Top: Vertical-down translation, Bottom: Clockwise rotation)

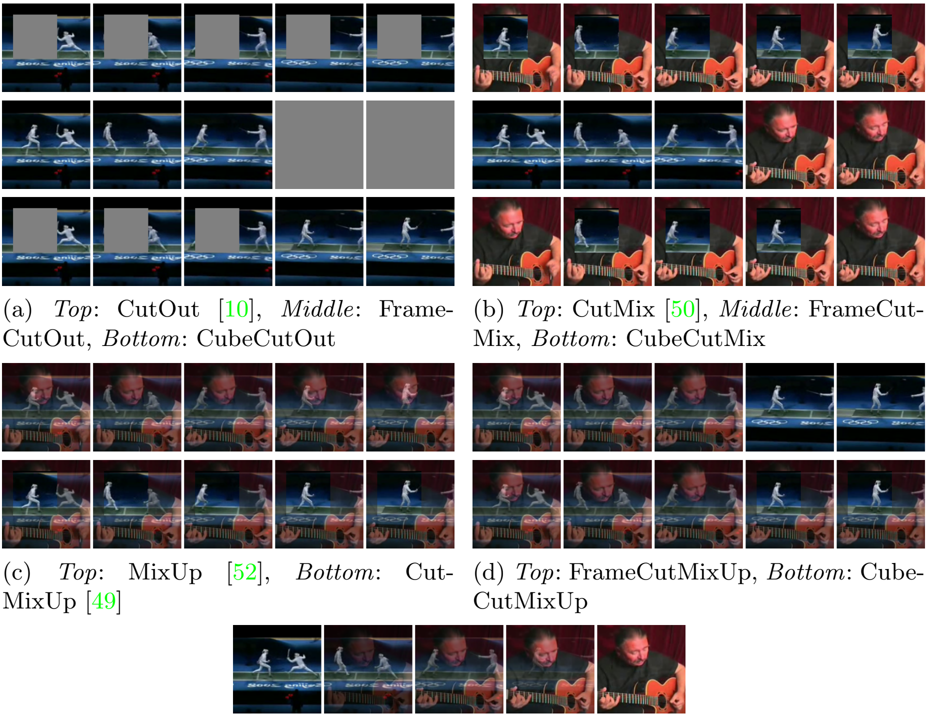
(b) Temporally varying photometric augmentations (Top: Increasing brightness, Bottom: Decreasing contrast)

Fig. 3: Example of temporally varying data augmentation operations for RangAugment-T

is a combination of MixUp and CutMix, which is proposed in [49] also can be applied to recognition to relax the unnatural boundary changes.

In this section, we propose temporal extensions of above algorithms. FrameCutOut and CubeCutOut is the temporal and spatio-temporal extension of CutOut (Fig 4 (a)), respectively. CutOut encourages the network to better utilize the full context of the image, rather than relying on a small portion of specific spatial regions. Similarly, FrameCutOut encourages the network to better utilize the full temporal context, and the full spatio-temporal context by CubeCutOut.

FrameCutMix and CubeCutMix is the extension of CutMix [50] (Fig 4 (b)). CutMix is designed for learning of spatially localizable features. Cut and paste mixing between two images encourages the network to learn where to recognize. Similarly, FrameCutMix and CubeCutMix is designed for learning of temporally and spatio-temporally localizable features in a video. Like CutMix, mixing ratio λ is sampled from beta distribution $Beta(\alpha, \alpha)$, where α is a hyper-parameter, and locations for random frames or random spatio-temporal cubes are selected based on λ.



(a) *Top*: CutOut [10], *Middle*: FrameCutOut, *Bottom*: CubeCutOut

(b) *Top*: CutMix [50], *Middle*: FrameCutMix, *Bottom*: CubeCutMix

(c) *Top*: MixUp [52], *Bottom*: CutMixUp [49]

(d) *Top*: FrameCutMixUp, *Bottom*: CubeCutMixUp

(e) FadeMixUp

Fig. 4: Visual comparison of data-level deleting, blending, and mixing for videos. Desired ground-truth labels are calculated by the ratio of each class: *Fencing* and *PlayingGuitar*.

Like CutMixUp [49], which is the unified version of MixUp [52] and CutMix [50], FrameCutMixUp and CubeCutMixUp can be designed in similar way (Fig 4 (c) and (d)) to relax extreme boundary changes between two samples. For these blend+mix algorithms, MixUp is applied between two data samples by mixing ratio λ_1, and the other hyper-parameter λ_2 is sampled from $Beta(2, 2)$. Based on λ_2, region mask $\mathbf{M}$ is selected randomly like CutMix to mix again between MixUp-ed sample and one of the two samples. Mixed data and desired ground-truth labels are formulated as below.

$$\begin{aligned} \tilde{x} &= \begin{cases} (\lambda_1 x_A + (1-\lambda_1)x_B) \odot \mathbf{M} + x_A \odot (\mathbf{1} - \mathbf{M}) & \text{if } \lambda_1 < 0.5 \\ (\lambda_1 x_A + (1-\lambda_1)x_B) \odot \mathbf{M} + x_B \odot (\mathbf{1} - \mathbf{M}) & \text{if } \lambda_1 \geq 0.5 \end{cases} \\ \tilde{y} &= \begin{cases} (\lambda_1\lambda_2 + (1-\lambda_1))y_A + (1-\lambda_1)\lambda_2 y_B & \text{if } \lambda_1 < 0.5 \\ \lambda_1\lambda_2 y_A + (1-\lambda_1\lambda_2)y_B & \text{if } \lambda_1 \geq 0.5 \end{cases} \end{aligned} \tag{1}$$

where $\tilde{x}$, $\tilde{y}$, and $\odot$ indicate mixed data, modified label, and element-wise multiplication, respectively.

Table 1: Comparison between deleting, blending, and mixing frameworks.

	Type	Delete			Mix			Blend		Blend + Mix		
	Name	CutOut [10]	Frame CutOut	Cube CutOut	CutMix [50]	Frame CutMix	Cube CutMix	MixUp [52]	Fade MixUp	CutMixUp [49]	Frame CutMixUp	Cube CutMixUp
Axis	Spatial	✓		✓	✓		✓			✓		✓
	Temporal		✓	✓		✓	✓		✓		✓	✓

Finally, we propose another extension of MixUp, called FadeMixUp. Inspired by fade-in, fade-out, dissolve overlap effects in videos, from MixUp, mixing ratio is smoothly changing along with temporal frames (Fig 4 (e)). In FadeMixUp, list of the mixing ratio $\tilde{\lambda}_t$ of a frame t, is calculated by linear interpolation between $\lambda - \gamma$ and $\lambda + \gamma$, where λ is the mixing ratio of MixUp and γ is sampled from $Uniform(0, min(\lambda, 1-\lambda))$. Because the adjustments of mixing ratio at the both ends are symmetric, label is same as MixUp.

$$\begin{aligned}\tilde{x_t} &= \tilde{\lambda_t} X_{A_t} + (1 - \tilde{\lambda_t}) X_{B_t} \\ \tilde{y} &= \lambda y_A + (1 - \lambda) y_B,\end{aligned} \tag{2}$$

FadeMixUp can be modeled for temporal variations and can learn temporally localizable feature without sharp boundary changes like other mixing algorithms. Since many videos include these overlapping effects at the scene change, FadeMixUp can be applied naturally.

Summarization of deleting, blending, and mixing data augmentation algorithms are described in Table 1. In the table, checkmark indicates the elements (pixels) can be changed along the spatial or temporal axis by augmentation methods. Compared to existing algorithms [10, 50, 52, 49], our proposed methods are extended temporally and spatio-temporally.

4 Experiments

4.1 Experimental Settings

For video action recognition, we train and evaluate on the UCF-101 [36] and HMDB-51 [25] dataset. UCF-101 originally consists of 13,320 videos with 101 classes. It consists of three train/test splits, but we used the modified split provided by the 1st VIPriors action recognition challenge that consists 4,795 training videos and 4,742 validation videos. HMDB-51 consists of 6,766 videos with 51 classes. We use original three train/test splits for training and evaluations.

Our experiments are trained and evaluated on a single GTX 1080-ti GPU and implemented by PyTorch framework. We use SlowFast-50 [11] as backbone network with 64 temporal frames because it is more lightweight and faster than other networks such as C3D [39], I3D [3], and S3D [47] without any pre-training and optical-flow. For baseline, basic data augmentation such as random crop with size 160, random scale jittering between [160, 200] for short side of video, and random horizontal flip is applied. For optimization, batch size is set to 16,

Table 2: Data Augmentation Results

	Range	Top-1 Acc.	Top-5 Acc.
Baseline		49.37	73.62
RandAugment	Spatial	66.87	88.04
	Temporal	67.33	88.42
	Temporal+	**69.23**	**89.20**
	Mix	68.24	89.25

Table 3: Data Deleting Results

	Top-1 Acc.	Top-5 Acc.
Baseline	**49.37**	**73.62**
CutOut	46.01	69.80
FrameCutOut	47.60	71.32
CubeCutOut	47.45	72.06

Table 4: Data Mixing Results

	Top-1 Acc.	Top-5 Acc.
Baseline	49.37	73.62
CutMix($\alpha = 2$)	50.81	75.62
FrameCutMix($\alpha = 2$)	51.29	74.99
FrameCutMix($\alpha = 5$)	**53.10**	**76.61**
CubeCutMix($\alpha = 2$)	51.86	74.34
CubeCutMix($\alpha = 5$)	51.81	75.16

Table 5: Data Blending Results

	Top-1 Acc.	Top-5 Acc.
Baseline	49.37	73.62
MixUp	59.60	82.56
FadeMixUp	59.22	82.24
CutMixUp	59.35	81.99
FrameMixUp	**60.67**	**83.47**
CubeMixUp	59.85	82.20

learning rate is set to 1e-4, weight decay of 1e-5 is used, learning rate warmup [14] and cosine learning rate scheduling [30] is used with Adam optimizer [22]. We train the all models for 150 epochs. For evaluation, we sample 10 clips uniformly along temporal axis, and average softmax predictions.

4.2 Data-level temporal data augmentations

Table 2 shows recognition results on UCF-101 validation set of VIPriors challenge. For all result tables, **bold** is the best one and underline is the second best . RandAugment-spatial indicates original implementation without temporal variations. In temporal version, M1 of Fig. 2 is sampled from $Uniform(0.1, M2)$ and M2 is set to M of spatial RandAugment. For temporal+, M1 and M2 are set to M$-\delta$ and M$+\delta$, respectively, where δ is sampled from $Uniform(0, 0.5 \times M)$. For Mix in Table 2, it randomly chooses spatial or temporal+. Results show that applying RandAugment solely improves recognition performance drastically. Among them, temporal expended RandAugment-T (temporal+) shows the best performance. For all RandAugment results, to produce the best accuracy, grid search of two hyper-parameters: N $\in [1, 2, 3]$ and M $\in [3, 5, 10]$, is used.

4.3 Data-level temporal deleting, mixing, and blending

For data deleting like CutOut [10], results of it and its temporal extensions, FrameCutOut and CubeCutOut, are described in Table 3. For CutOut, 80×80 spatial patch is randomly deleted, and for FrameCutOut, 16 frames are randomly deleted. For CubeCutOut $80 \times 80 \times 16$ cube is randomly deleted. Results show that deleting patches, frames, or spatio-temporal cubes hurts recognition performance in the limited number of dataset. Among them, CutOut shows the worst performances.

For data mixing like CutMix [50] and its extensions, results are described in Table 4. We apply the mixing probability of 0.5 for all methods and different

Table 6: Temporal Augmentation Results on HMDB51 Dataset

	Split-1		Split-2		Split-3		Average	
	Top-1 Acc.	Top-5 Acc.	Top-1 Acc.	Top-5 Acc.	Top-1 Acc.	Top-5 Acc.	Top-1 Acc.	Top-5 Acc.
Baseline	36.60	67.25	37.19	65.75	32.88	65.82	35.56	66.27
RandAug	47.45	79.21	47.12	76.86	47.45	77.97	47.34	78.01
RandAug-T	**48.17**	**79.35**	**47.84**	**77.00**	**48.37**	**78.17**	**48.13**	**78.17**
CutOut	**34.71**	**65.49**	**32.35**	63.79	31.76	62.94	**32.94**	**64.07**
FrameCutOut	31.05	61.57	32.16	**65.36**	**31.87**	**64.18**	31.69	63.70
CubeCutOut	33.01	63.99	32.04	64.25	30.59	62.81	31.88	63.68
CutMix	33.95	64.27	33.69	66.84	31.24	63.53	32.96	64.88
FrameCutMix	34.97	**65.56**	34.84	**67.91**	33.27	63.53	34.36	65.67
CubeCutMix	**35.10**	65.10	**35.95**	65.62	**36.54**	**67.97**	**35.86**	**66.23**
MixUp	38.95	68.10	**40.72**	70.92	40.20	71.31	39.96	70.11
CutMixUp	**40.92**	**71.07**	40.16	71.55	39.28	71.48	40.12	71.37
FrameMixUp	40.33	70.98	40.52	70.85	39.02	70.65	39.96	70.83
CubeMixUp	40.72	70.65	40.70	**72.88**	**40.92**	**71.83**	**40.78**	**71.79**
FadeMixUp	39.80	70.39	40.46	71.70	39.61	70.00	39.96	70.70

Table 7: Model Evaluation for VIPriors Challenge

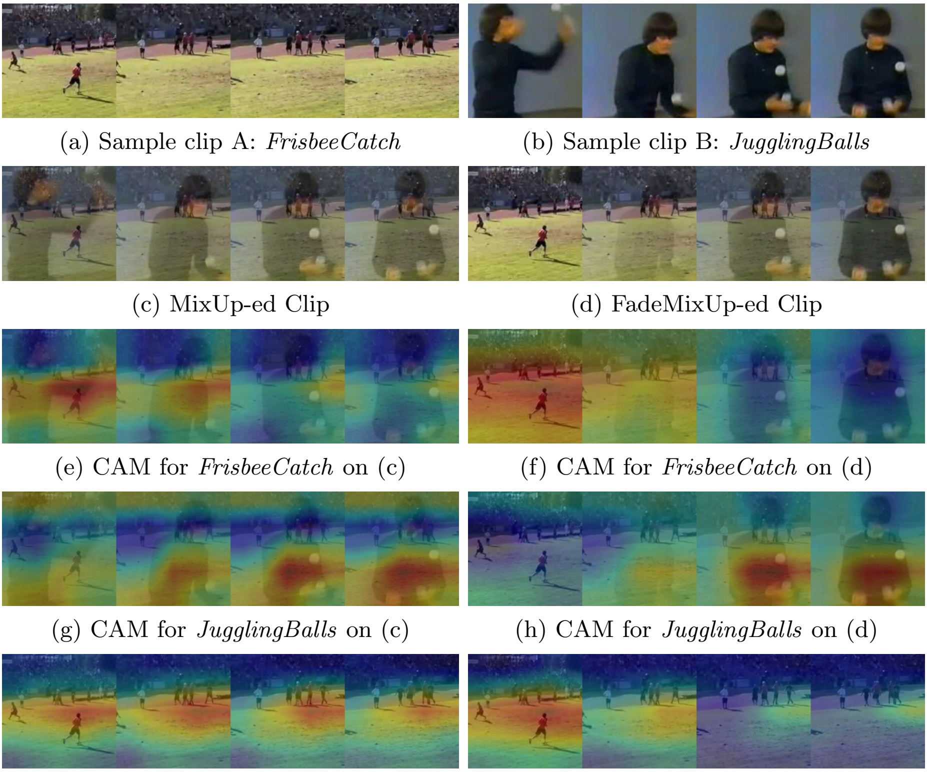
Train Data	Test Data	Augmentation	Regularization	Others	Top-1 Acc.	Top-5 Acc.
Train	Val				49.37	73.62
Train	Val	RandAug-T			69.23	89.20
Train	Val	RandAug-T	FadeMixUp		68.73	89.27
Train	Val	RandAug-T	FrameMixUp		**69.70**	**89.84**
Train+Val	Test				68.99	-
Train+Val	Test	RandAug-T			81.43	-
Train+Val	Test	RandAug-T	All Methods	Ensemble	**86.04**	-

hyper-parameters α. Since object size in the action recognition dataset is smaller than that of ImageNet [23], mixing ratio should be sampled in the region close to 0.5 by sampling large α in the beta distribution. Results show that the temporal and spatio-temporal extensions outperform spatial-only mixing strategy. Since probability of object occlusion is lower at temporal mixing than spatial mixing, FrameCutMix performance is the most improved.

Finally, for data blending, compared to MixUp [2] and CutMixUp [49], temporal and spatio-temporal extensions show slightly superior performance that are described in Table 5. Compared to deleting and mixing augmentations, blending shows the best performances. Since the number of training data is limited, linear convex combination of samples easily and effectively augments in the sample space.

4.4 Results on HMDB-51 dataset

To check the generalization to other dataset, we train and evaluate on HMDB-51 dataset with its original splits. Generally, recognition performance in HMDB-51



(a) Sample clip A: *FrisbeeCatch*

(b) Sample clip B: *JugglingBalls*

(c) MixUp-ed Clip

(d) FadeMixUp-ed Clip

(e) CAM for *FrisbeeCatch* on (c)

(f) CAM for *FrisbeeCatch* on (d)

(g) CAM for *JugglingBalls* on (c)

(h) CAM for *JugglingBalls* on (d)

(i) CAM for *FrisbeeCatch* on (a)

(j) CAM for *FrisbeeCatch* on (a)

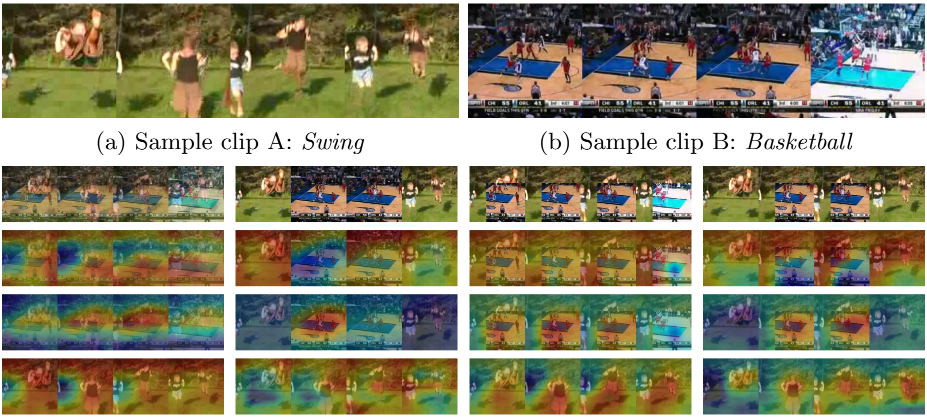
Fig. 5: Class actionvation maps. *Left*: MixUp, *Right*: FadeMixUp

is inferior to the performance of UCF-101 due to its limited number of training samples. We use same model and hyper-parameters as in UCF-101.

Results in Table 6 show that temporal extensions generally outperforms spatial-only versions, and similar to UCF-101, RandAugment and blending methods show the best accuracies.

4.5 1st VIPriors action recognition challenge

Based on the comprehensive experimental results, we attend the 1st VIPriors action recognition challenge. In this challenge, any pre-training and using external dataset is not allowed. Performances on various models are described in Table 7. For validation, applying both RandAugment-T and FrameMixUp show the best result. For test set, total 3,783 videos are provided without ground truths. Therefore, we report the results based on the challenge leaderboard. Combination of training and validation dataset, total 9,537 videos are used for training the final challenge entries. From the baseline accuracy, 68.99%, adapting RandAugment-



(a) Sample clip A: *Swing*

(b) Sample clip B: *Basketball*

(c) MixUp

(d) FrameCutMix

(d) CutMix

(d) CubeCutMix

Fig. 6: Class actionvation maps. For (c)-(f), from the top to the bottom row: mixed clips, CAMs for *Swing*, CAMs for *Basketball*, and CAMs for *Swing* on pure clip (a), respectively.

T only improves the performance up to 81.43%. Finally, we submitted ensembled version of different models that are trained using RandAugment-T and various mixing and blending augmentations, to produce 86.04% Top-1 accuracy.

4.6 Discussions

Why the improvement are not large? Although the temporal extensions generally outperform spatial-only versions in data augmentation algorithms, the performance improvements might be not large enough. Possible reasons of this are three-fold, the first one is the lack of enough training data, and the second one is the lack of temporal perturbation, and the last one is the datasets are used for experiments consists trimmed videos. Both UCF-101 and HMDB-51 dataset have little temporal perturbations. Therefore, applying spatial augmentation is enough to learn the contexts. And both dataset are trimmed that have little temporal occlusions, which means there is no room to learn the ability to localize temporally. For deleting and mixing, compared to the image dataset, since the action region is relatively small, removing spatial region can hurts the basic recognition performance if the number of training data is not enough. In contrast, for blending, although it is unnatural image as said in [50], it can exploit full region of frames. Therefore it produces reasonable performance improvements.

Spatio-temporal class activation map visualization We visualize the learned feature using class activation map [55] in Fig. 5. In the SlowFast network, we use the feature of the last convolutional layer in SlowPath. Fig. 5 (a) and (b) are example clips. Fig. 5 (c) and (d) are the visualization of MixUp-ep and

FadeMixUp-ed clips, respectively. In Fig. 5 (f) and (h) compared to Fig. 5 (e) and (g), FadeMixUp features are more localized temporally than that of MixUp. In Fig. 5 (j) compared to Fig. 5 (i), activations of FadeMixUp is spatio-temporally localized better than that of MixUp in the pure clip A.

Fig. 6 compares spatio-temporal localization abilities between MixUp, CutMix, FrameCutMix, and CubeCutMix. Compared to MixUp, as said in their paper [50], CutMix can localize spatially for basketball field and the person on swing. However, compared to CubeCutMix, activations of CutMix is not localized temporally well. FrameCutMix also cannot localize feature like MixUp, but it can separate the weights of activation separately in temporal axis.

5 Conclusion

In this paper, we proposed several extensions of data-level augmentation and data-level deleting, blending, and mixing augmentation algorithms from the spatial, or image domain into temporal and spatio-temporal, or video domain. Although applying spatial data augmentation itself increases the recognition performance in a limited amount of dataset, extending temporal and spatio-temporal data augmentation boosts the performance. Moreover, our models trained on temporal augmentation have abilities to localize temporally and spatio-temporally that cannot be achieved from the model trained on spatial augmentations only. Our next step will be an extension to the large-scale dataset such as Kinetics [3], or untrimmed videos.

References

1. Berthelot, D., Carlini, N., Cubuk, E.D., Kurakin, A., Sohn, K., Zhang, H., Raffel, C.: Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In: International Conference on Learning Representations (2019) 1
2. Berthelot, D., Carlini, N., Goodfellow, I., Papernot, N., Oliver, A., Raffel, C.A.: Mixmatch: A holistic approach to semi-supervised learning. In: Advances in Neural Information Processing Systems. pp. 5049–5059 (2019) 1, 10
3. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017) 4, 8, 13
4. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. arXiv preprint arXiv:2002.05709 (2020) 1
5. Chen, Y., Fan, H., Xu, B., Yan, Z., Kalantidis, Y., Rohrbach, M., Yan, S., Feng, J.: Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3435–3444 (2019) 4
6. Chen, Y., Rohrbach, M., Yan, Z., Shuicheng, Y., Feng, J., Kalantidis, Y.: Graph-based global reasoning networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 433–442 (2019) 4
7. Choi, J., Gao, C., Messou, J.C., Huang, J.B.: Why can't i dance in the mall? learning to mitigate scene bias in action recognition. In: Advances in Neural Information Processing Systems. pp. 853–865 (2019) 4
8. Cubuk, E.D., Zoph, B., Mane, D., Vasudevan, V., Le, Q.V.: Autoaugment: Learning augmentation strategies from data. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 113–123 (2019) 1, 3
9. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 702–703 (2020) 1, 2, 3, 5
10. DeVries, T., Taylor, G.W.: Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552 (2017) 1, 2, 3, 5, 7, 8, 9
11. Feichtenhofer, C., Fan, H., Malik, J., He, K.: Slowfast networks for video recognition. In: Proceedings of the IEEE international conference on computer vision. pp. 6202–6211 (2019) 4, 8
12. Gastaldi, X.: Shake-shake regularization. arXiv preprint arXiv:1705.07485 (2017) 1, 4
13. Ghiasi, G., Lin, T.Y., Le, Q.V.: Dropblock: A regularization method for convolutional networks. In: Advances in Neural Information Processing Systems. pp. 10727–10737 (2018) 1, 4
14. Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., He, K.: Accurate, large minibatch sgd: Training imagenet in 1 hour. arXiv preprint arXiv:1706.02677 (2017) 9
15. Hara, K., Kataoka, H., Satoh, Y.: Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 6546–6555 (2018) 5
16. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9729–9738 (2020) 1

17. Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations. Proceedings of the International Conference on Learning Representations (2019) 2
18. Hendrycks, D., Mu, N., Cubuk, E.D., Zoph, B., Gilmer, J., Lakshminarayanan, B.: Augmix: A simple data processing method to improve robustness and uncertainty. arXiv preprint arXiv:1912.02781 (2019) 1, 2, 3
19. Huang, G., Sun, Y., Liu, Z., Sedra, D., Weinberger, K.Q.: Deep networks with stochastic depth. In: European conference on computer vision. pp. 646–661. Springer (2016) 1, 4
20. Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., Aila, T.: Training generative adversarial networks with limited data. arXiv preprint arXiv:2006.06676 (2020) 1
21. Kim, J., Cha, S., Wee, D., Bae, S., Kim, J.: Regularization on spatio-temporally smoothed feature for action recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12103–12112 (2020) 1, 4
22. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 9
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012) 1, 3, 10
24. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012) 1
25. Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: Hmdb: a large video database for human motion recognition. In: 2011 International Conference on Computer Vision. pp. 2556–2563. IEEE (2011) 2, 5, 8
26. Lee, J.H., Zaigham Zaheer, M., Astrid, M., Lee, S.I.: Smoothmix: A simple yet effective data augmentation to train robust classifiers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 756–757 (2020) 1, 3
27. Li, H., Zhang, X., Xiong, H., Tian, Q.: Attribute mix: Semantic data augmentation for fine grained recognition. arXiv preprint arXiv:2004.02684 (2020) 1, 3
28. Li, Y., Li, Y., Vasconcelos, N.: Resound: Towards action recognition without representation bias. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 513–528 (2018) 4
29. Lim, S., Kim, I., Kim, T., Kim, C., Kim, S.: Fast autoaugment. In: Advances in Neural Information Processing Systems. pp. 6665–6675 (2019) 1, 2, 3
30. Loshchilov, I., Hutter, F.: Sgdr: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016) 9
31. Misra, I., Maaten, L.v.d.: Self-supervised learning of pretext-invariant representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6707–6717 (2020) 1
32. Ryoo, M.S., Piergiovanni, A., Tan, M., Angelova, A.: Assemblenet: Searching for multi-stream neural connectivity in video architectures. arXiv preprint arXiv:1905.13209 (2019) 4
33. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (2015) 1, 3

34. Singh, K.K., Lee, Y.J.: Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In: 2017 IEEE international conference on computer vision (ICCV). pp. 3544–3553. IEEE (2017) 1, 3
35. Sohn, K., Berthelot, D., Li, C.L., Zhang, Z., Carlini, N., Cubuk, E.D., Kurakin, A., Zhang, H., Raffel, C.: Fixmatch: Simplifying semi-supervised learning with consistency and confidence. arXiv preprint arXiv:2001.07685 (2020) 1
36. Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402 (2012) 2, 5, 8
37. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research **15**(1), 1929–1958 (2014) 1, 4
38. Stroud, J., Ross, D., Sun, C., Deng, J., Sukthankar, R.: D3d: Distilled 3d networks for video action recognition. In: The IEEE Winter Conference on Applications of Computer Vision. pp. 625–634 (2020) 4
39. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 4489–4497 (2015) 4, 8
40. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M.: A closer look at spatiotemporal convolutions for action recognition. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 6450–6459 (2018) 4
41. Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., Bengio, Y.: Manifold mixup: Better representations by interpolating hidden states. In: International Conference on Machine Learning. pp. 6438–6447 (2019) 1, 4
42. Walawalkar, D., Shen, Z., Liu, Z., Savvides, M.: Attentive cutmix: An enhanced data augmentation approach for deep learning based image classification. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3642–3646. IEEE (2020) 1, 3
43. Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., Tang, X., Van Gool, L.: Temporal segment networks: Towards good practices for deep action recognition. In: European conference on computer vision. pp. 20–36. Springer (2016) 4
44. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7794–7803 (2018) 4
45. Xie, Q., Dai, Z., Hovy, E., Luong, M.T., Le, Q.V.: Unsupervised data augmentation for consistency training. arXiv preprint arXiv:1904.12848 (2019) 1
46. Xie, Q., Luong, M.T., Hovy, E., Le, Q.V.: Self-training with noisy student improves imagenet classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10687–10698 (2020) 1
47. Xie, S., Sun, C., Huang, J., Tu, Z., Murphy, K.: Rethinking spatiotemporal feature learning for video understanding. arXiv preprint arXiv:1712.04851 **1**(2), 5 (2017) 4, 8
48. Yamada, Y., Iwamura, M., Akiba, T., Kise, K.: Shakedrop regularization for deep residual learning. IEEE Access **7**, 186126–186136 (2019) 1, 4
49. Yoo, J., Ahn, N., Sohn, K.A.: Rethinking data augmentation for image super-resolution: A comprehensive analysis and a new strategy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8375–8384 (2020) 1, 2, 3, 6, 7, 8, 10
50. Yun, S., Han, D., Oh, S.J., Chun, S., Choe, J., Yoo, Y.: Cutmix: Regularization strategy to train strong classifiers with localizable features. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6023–6032 (2019) 1, 2, 3, 5, 6, 7, 8, 9, 12, 13

51. Zhang, H., Zhang, Z., Odena, A., Lee, H.: Consistency regularization for generative adversarial networks. In: International Conference on Learning Representations (2019) 1
52. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017) 1, 2, 3, 5, 7, 8
53. Zhao, S., Liu, Z., Lin, J., Zhu, J.Y., Han, S.: Differentiable augmentation for data-efficient gan training. arXiv preprint arXiv:2006.10738 (2020) 1
54. Zhao, Z., Singh, S., Lee, H., Zhang, Z., Odena, A., Zhang, H.: Improved consistency regularization for gans. arXiv preprint arXiv:2002.04724 (2020) 1
55. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929 (2016) 12