A Large-scale Robustness Analysis of Video Action Recognition Models (Supplementary Material)

Madeline Chantry Schiappa [1*†], Naman Biyani [2*], Prudvi Kamtam[1]
Shruti Vyas [1], Hamid Palangi [3], Vibhav Vineet [3‡], Yogesh Rawat [1‡]
CRCV, University of Central Florida [1], IIT Kanpur [2], and Microsoft Research [3]

1. Overview

This supplementary includes additional results that were not available in the main paper. Section 2 includes additional results on UCF-101P, HMDB-51P, Kinetcs-400P, and SSv2P. More specifically:

- Section 2.2 goes over the results for perturbations of varying severity in the datasets UCF-101P, HMDB-51P, Kinetics-400P, and provides more detail on SSv2P.
- Section 2.3 shows more details on the results for absolute and relative robustness scores on all three datasets.
- Section 2.4 provides further analysis on pre-training versus models from scratch on UCF-101P and HMDB-51P.
- Section 2.5 provides a more in-depth analysis on the class confusions that result from SSv2P.

Section 3 goes into more detail on the perturbations applied to generate UCF-101P, HMDB-51P, Kinetics-400P and SSv2P. Next, we will provide details on model training using perturbations as augmentations followed by details of UCF-101-DS dataset.

2. Additional results

Here we provide detailed results on all the datasets. Figure 1 summarizes the performance of different models on various datasets for different perturbations. Here we can observe the differences in the behavior of these datasets how they differ for different type of perturbations. More specifically, SSv2-P differs from other datasets such as Kinetics-400P and UCF-101P.

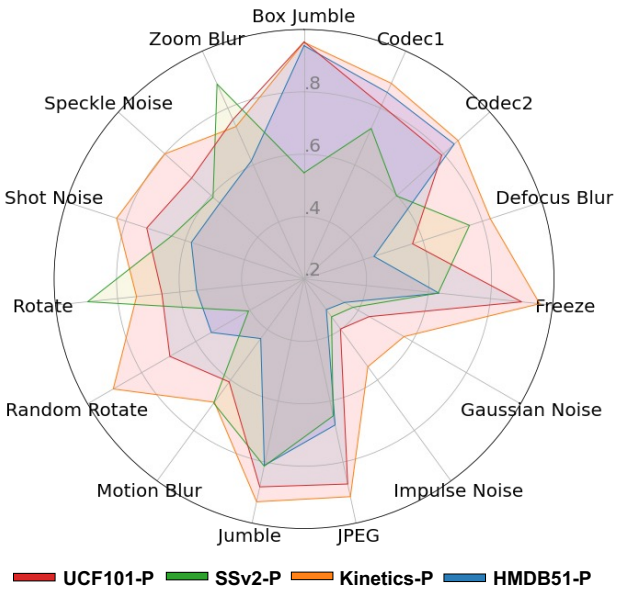

Figure 1. Mean performance on the different perturbed datasets.

2.1. Real-World Distribution Shifts

Results for models when trained on different data and evaluated on USF101-DS is shown in Figure 2. Each column is a different version of the UCF101 dataset. Mixed perturbations are a combination of all perturbations and PixMix are perturbations from [1] extended for video. For more details on the training implementation, see Section 3.1. Table 1 shows the relative robustness (γ^r) scores across the different distribution shift categories comparing models trained on the clean UCF101 dataset versus perturbed. In this case, we treat the models trained on perturbations as the original score, where $\gamma_p^r = 1-(A_p^f - A_p^{f_p})/A_p^{f_p}$. We can observe that MViT performs much worse when trained on perturbations while CNN based models like ResNet and X3D improve scores when using spatial perturbations.

*The authors contributed equally as first authors to this paper.
†Corresponding author:madelineschiappa@knights.ucf.edu
‡The authors contributed equally as supervisors to this paper.

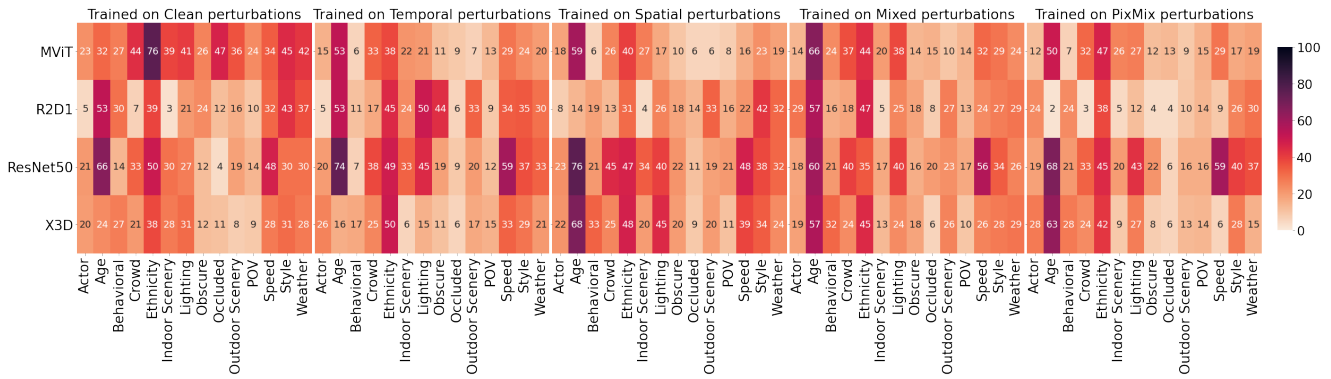

Figure 2. Model evaluation on the UCF101-DS dataset when models are trained on clean versus different combinations of perturbed datasets. Mixed perturbations are both spatial and temporal perturbations and PixMix is from [1].

Model	Mixed (γ^r)	PixMix (γ^r)	Spatial (γ^r)	Temporal (γ^r)
MViT	0.59	0.29	0.08	0.21
R2D1	1.02	0.37	0.86	**1.16**
ResNet50	1.06	**1.11**	1.16	1.12
X3D	**1.12**	0.98	**1.25**	0.90

Table 1. Relative robustness scores comparing models trained on perturbed UCF101 to models trained on clean UCF101, evaluated and averaged over UCF101-DS categories. Here, $\gamma_p^r = 1 - (A_p^f - A_p^{f_p})/A_p^{f_p}$.

2.2. Variation in severity

Figure 3, Figure 4, and Figure 5 shows the performance of all the models with different perturbations at varying severity levels for UCF-101P, Kinetics-400P, and HMDB-51P respectively. The severity level varies from 0 to 5, where 0 refers to clean videos and 5 refers to heavy perturbation. We observe that the transformer based models, MViT and Timesformer, are generally more robust as severity level increases in most of the perturbations, For some of the perturbations, such as defocus blur and motion blur, the performance of all the models drops significantly. However, there are some perturbations, such as short noise and speckle noise, where the transformer based models are robust across all the severity levels. Moreover, there are some perturbations, where all the models are found to be robust against all severity levels, such as box jumbling. We observe similar behavior in HMDB-51P dataset as well. In addition, Timesformer model is found to be robust against Translation and Random rotation for all the severity levels.

Similarly, Figure 6 shows the performance of four models with different perturbations at varying levels for SSv2P. We observe that the transformer based model, Timesformer, is generally more robust as severity level increases for the temporal perturbations. For the appearance based perturbations that are blur and noise related, all models drop in performance significantly. Figure 7 further visualizes the decrease in performance for appearance based perturbations. We observe the degradation of distinct clusters for the SSv2 dataset over increased severity for noise perturbations for all three models.

2.3. Absolute and relative robustness

Table 2, Table 4, Table 5, and Table 3 show the robustness scores for all the perturbations for all models on Kinetics-400P, UCF-101P, HMDB-51P and SSv2P respectively. In addition, Table 4, and Table 5 also show the robustness score for both pre-trained and training from scratch performance where the pre-trained weights are taken from Kinetics-400 pre-training. In Table 2, we observe that the transformer based models are generally more robust than CNN counterparts where MViT performs the best on Blur, Noise, and Digital perturbations, Timesformer performs the best on Camera motion perturbations. We also observe that R3D based model performs best on Temporal perturbations but the margin is very small when compared with other models.

In Table 4, we observe a similar behavior, where transformer based models are the best performers for all the perturbations when pre-training is used. However, we observe that when pre-training is not utilized, CNN models are better performers for Blur, Camera motion, and Temporal perturbations.

In Table 3 we observe that while the transformer based model outperforms in temporal and camera perturbations, it does not in the other appearance based categories. The transformer based model however significantly outperforms the other models in temporal perturbations while it is less significantly outperformed by other models in blur, noise, and digital. This emphasizes how important time is for the SSv2 dataset and how well transformer-based models learn temporally relevant features.

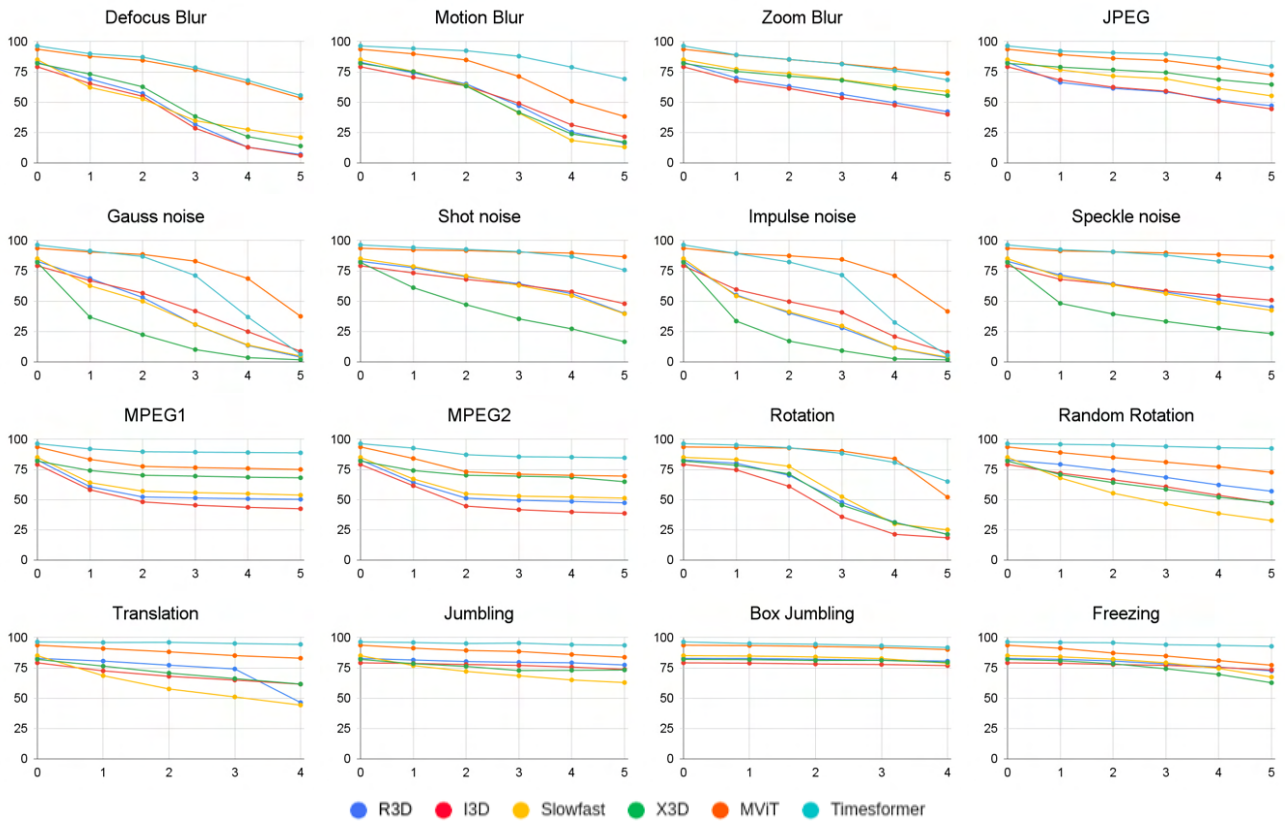

Figure 3. Robustness analysis of models with varying severity levels on UCF-101P benchmark. The y-axis shows accuracy and the x-axis represents severity level where 0 indicates performance on clean videos.

2.4. Pretraining vs Scratch

Figure 13 shows a comparison between pre-training and scratch performance for all the models across various perturbation categories on HMDB-51P dataset. We observe that although MViT is more robust against various perturbations when pre-training is used, its performance drops significantly when pre-training is not utilized. This behavior is similar to what we observe on UCF-101P dataset. More notably, we observe that for Translate and Variable rotation perturbation, the performance of MViT is worse than all the other CNN counterparts, despite the fact that it outperforms all those models when clean videos are used.

2.5. SSv2 Class Analysis

Figure 9 shows additional perturbations at severity 4 with five classes and their respective opposites. While all models struggle to significantly separate classes that are equivalent in all but direction, the Timesformer is noticeable better at maintaining clusters when at higher severities of temporal perturbations. While X3D shows better distinction of clusters from the start, as the severity increases, these clusters overlap more and more. The worse performing CNN example is unable to maintain clusters for all the temporal perturbations at increased severity.

Figure 10 shows additional confusion matrices for Timesformer and Slowfast with box jumble, jumble, shot noise, impulse noise and speckle noise perturbations at severity 4. The Timesformer architecture significantly outperforms Slowfast in class predictions for temporal predictions but suffers similarly with noise. The confusion matrices additionally show that the models are often predicting only a small selection of classes for all samples as shown by the vertical blue bars. This is especially noticeable for Slowfast on the noisy perturbations for the class “Showing something next to something”. The classes most often predicted differ when it is temporal perturbations, in which case for Slowfast the most incorrectly predicted class is “Hitting something with something” followed by “Stacking number of something”. Sample videos for these three classes are shown in Figure 11 where if there was a noise perturbation, the bottom two videos would be classified as the top video while for noisy pertrubations it would be the reverse. For the temporal confusion, is likely the case because when temporal pertrubations are applied, the actual interaction between the objects

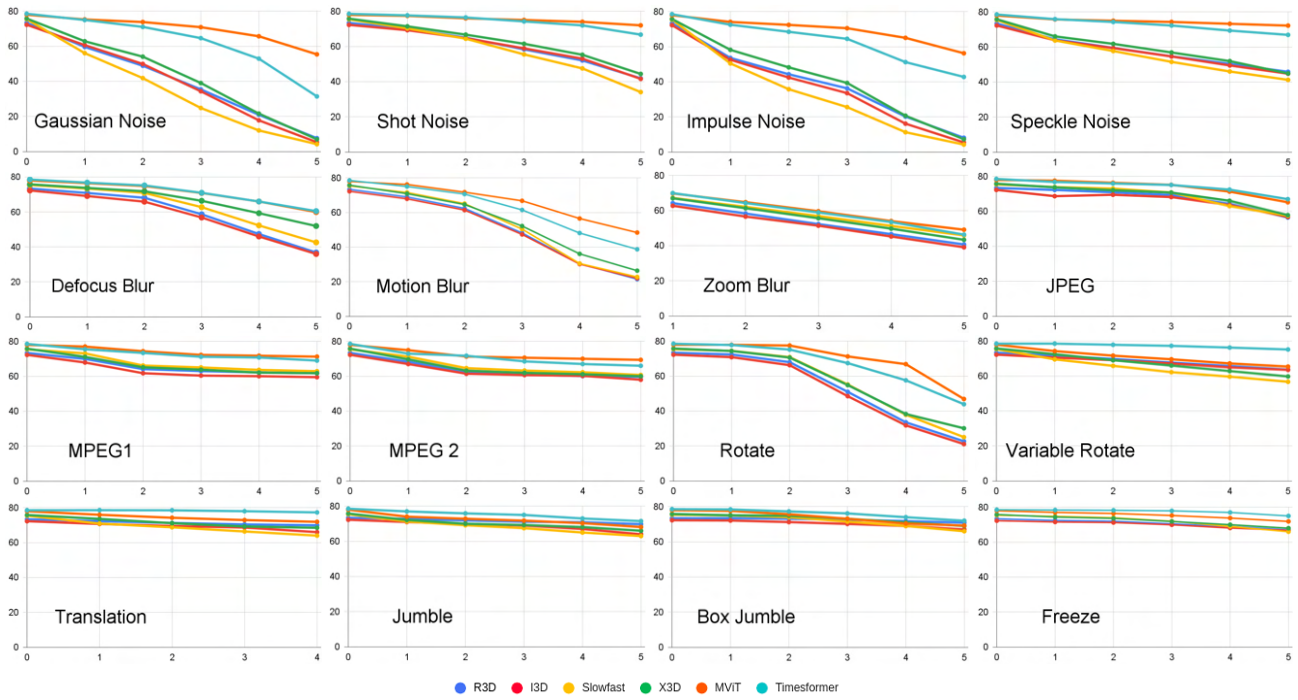

Figure 4. Robustness analysis of models with varying severity levels on Kinetics-400P benchmark. The y-axis shows accuracy and the x-axis represents severity level. When the severity is 0, this means no perturbation was applied and is the model's performance on clean videos.

is jumbled and could appear to just be one object hitting the other at random points in time.

3. Implementation details

In Figure 14, we show some sample video frames from Kinetics-400P showing different severity levels for some of the perturbations. We can see clearly that the level of perturbations increase as we move from 1 to 5. More sample videos are available in the project webpage: `bit.ly/3TJLMUF`.

The implementations of the various perturbations is provided in the project page. Following are brief details of severity levels for the perturbations:

Noise Perturbations: For gaussian noise, we increase the standard deviation of the gaussian distribution(from where we sample noise which is added) and similarly for speckle noise, we increase the standard deviation of the gaussian distribution(from where we sample noise which is multiplied with pixelwise itensities and then added) as we increase the severity levels. For impulse noise, we increase the proportion of image pixels to be replaced in the original frame with noise . For shot noise, we increase the proportion of image pixels to be replaced in the original frame with noise as we increase the severity levels.

Blur Perturbations: In defocus blur, we increase the radius of the disk which is convolved over the image to create defocus blurring effect. In motion blur, we increase the radius and sigma of the kernel which is used to create the motion blurring effect. In zoom blur, we increase as we increase the severity levels.

Digital Perturbations: For JPEG, MPEG1 and MPEG2 the amount of compression is increased as we increase the severity levels.

Camera Perturbations: For rotation, we rotate each frame by angle and the angle increases over the severity levels. For random rotation, each frame is rotated by a random angle from a range , which is increased as we increase the severity level. For translation, we randomly choose the center while cropping the image from 256x256 to 224x224 resolution,

Temporal Perturbations: For jumbling, we divide video into segments and randomly shuffle frames of those segments, the segment size increases from 4 to 64 from level 1 to 5. For box jumbling, the divided segments are jumbled, the segment size decreases from 64 to 4 as we increase the severity levels. For freezing, we increase the threshold value below which we freeze the frame(keep the previous frame

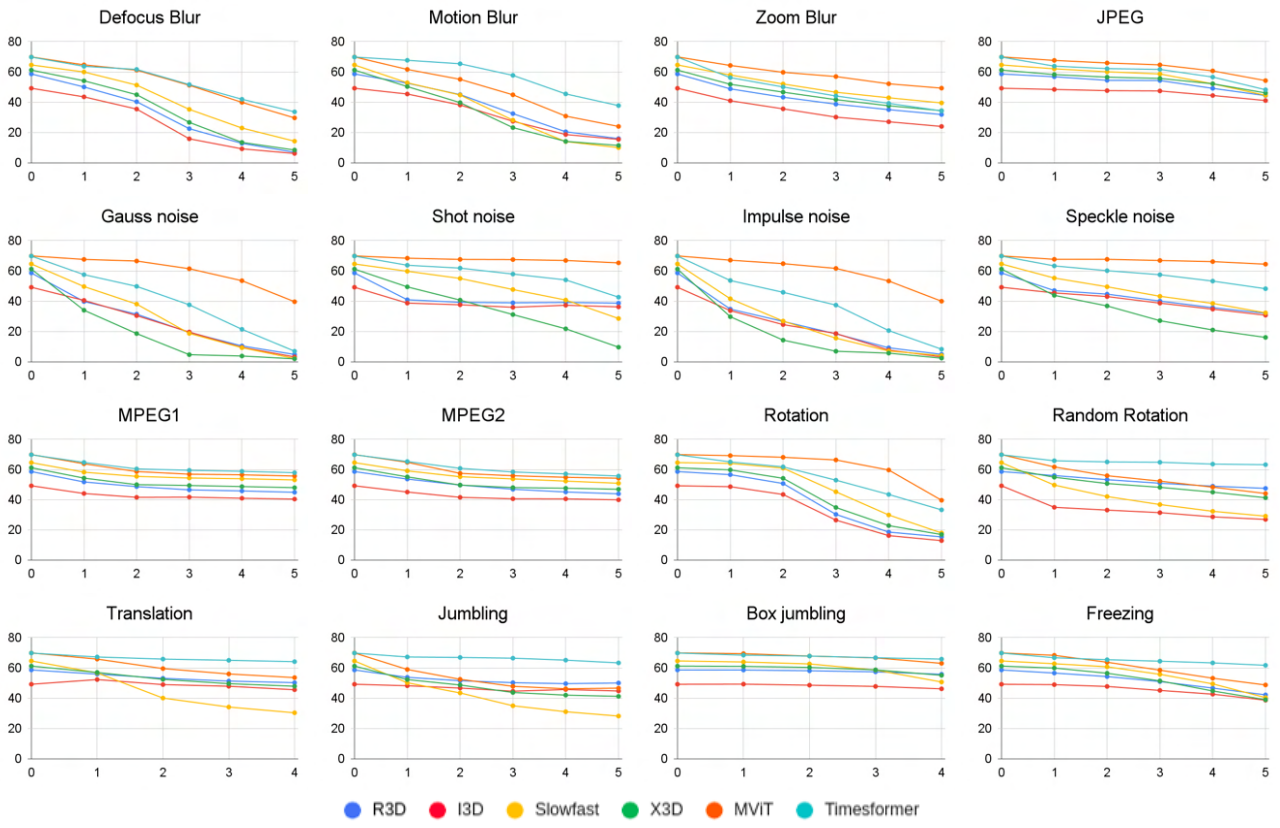

Figure 5. Robustness analysis of models with varying severity levels on HMDB-51P benchmark. The y-axis shows accuracy and the x-axis represents severity level. When the severity is 0, this means no perturbation was applied and is the model's performance on clean videos.

for that index) as we increase the severity levels. For implementation of these, we save the perturbed frame indices and use that while loading the data for the model.

For implementation of model evaluations and to get pretrained weights for the models, we used the open source video understanding codebase PySlowfast [1] . We also used the code provided by them to finetune models on UCF101 and HMDB51 dataset. Also various examples sample videos of Kinetics-400P are provided in the project webpage. Figure 15 shows sample video frame from Kinetics-400P showing different perturbations using blur and noise.

3.1. Training on Perturbations

We trained one CNN-based model, ResNet50, and one transformer-based model, MViT. Both models are pretrained on the Kinetics400 dataset. In order to understand how training on different type of perturbations may impact overall performance, we train the ResNet50 and MViT model on temporal, spatial, mixed and the state-of-the-art PixMix [1]. Originally from the image-domain, PixMix adds augmentations by mixing a given image with diverse patterns from fractals and feature visualizations. For training on temporal, a perturbation is randomly selected from *jumble*, *freeze*, and *sampling*. When evaluation on the temporal category, the perturbations are randomly chosen from *jumble*, *freeze*, *sampling*, *box jumble*, and *reverse sampling*. For training on spatial, a perturbation is randomly selected from *speckle noise*, *gaussian noise*, and *rotate*. For testing on spatial, a pertubation is randomly selected from *shot noise*, *static rotate*, *translate*, and *impulse noise*. For mixed, a perturbation category is first selected then a perturbation type from that category. For temporal, spatial and mixed during training, severities are chosen at random between 1,2, and 3 for training and 4 or 5 for testing. For PixMix [1], we apply the augmentation at severity 3 for each frame individually, in which a different fractal image is chosen for each. We trained one CNN-based model, ResNet50, and one transformer-based model, MViT. Both models are pretrained on the Kinetics400 dataset.

[1] https://github.com/facebookresearch/SlowFast

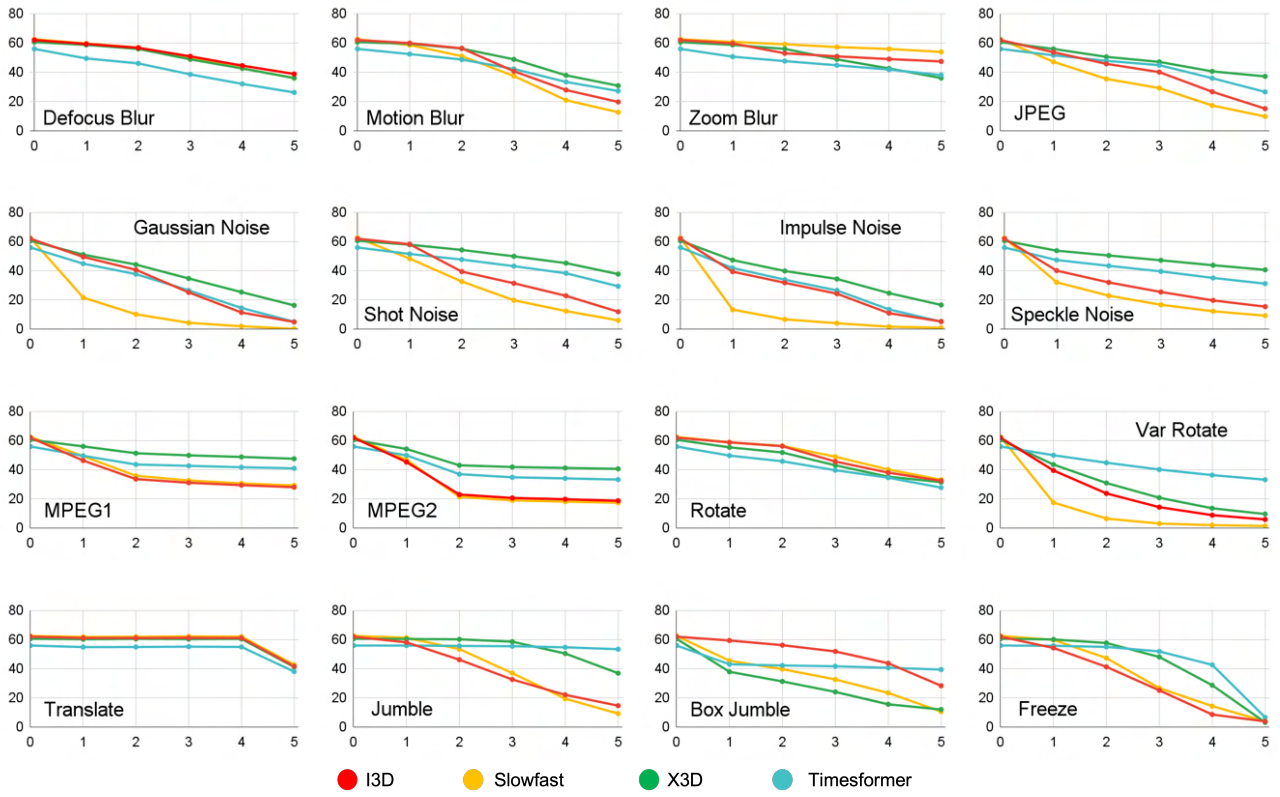

Figure 6. Robustness analysis of models with varying severity levels on SSv2P benchmark. The y-axis shows accuracy and the x-axis represents severity level. When the severity is 0, this means no perturbation was applied and is the model's performance on clean videos.

Figure 7. A comparison of embeddings for speckle noise and impulse noise for Timesformer, X3D, and Slowfast on the SSv2 dataset. The first row shows clusters at with no perturbation, the second row shows severity 2 and the third row shows severity 4. As the severity increases, the three models shown consistently decrease in their ability to form distinct clusters.

4. UCF101-DS dataset

The UCF101-DS[2] dataset consists of distribution shifts for 47 classes (see Figure 16) with 63 different distribution

[2]For more information and to download this dataset, visit https://www.crcv.ucf.edu/research/projects/ucf101-ds-action-recognition-for-real-world-distribution-

	R3D		I3D		SF		X3D		Timesformer		MViT	
Defocus Blur	.83	.77	.82	.75	.85	.80	.89	.85	.84	.80	.87	.83
Motion Blur	.73	.63	.78	.70	.73	.64	.74	.66	.80	.75	.86	.82
Zoom Blur	.79	.71	.81	.74	.85	.80	.89	.85	.91	.89	.92	.90
Blur	.78	.70	.80	.72	.80	.73	.81	.75	.84	.79	**.86**	**.82**
Gaussian	.61	.47	.61	.46	.52	.36	.61	.49	.80	.75	.90	.87
Shot	.84	.78	.85	.79	.79	.82	.84	.79	.95	.94	.96	.95
Impulse	.59	.44	.58	.42	.50	.34	.59	.46	.81	.76	.90	.87
Speckle	.82	.75	.82	.75	.77	.70	.81	.75	.93	.91	.96	.95
Noise	.71	.61	.72	.61	.64	.53	.71	.62	.87	.84	**.93**	**.91**
JPEG	.93	91	.93	.90	.94	.92	.92	.89	.95	.94	.95	.94
MPEG1	.92	.89	.90	.86	.91	.88	.90	.87	.94	.92	.95	.94
MPEG2	.89	.85	.90	.86	.89	.85	.87	.83	.91	.89	.93	.91
Digital	.91	.88	.91	.87	.91	.89	.90	.86	**.94**	.92	**.94**	**.93**
Rotation	.76	.67	.75	.65	.77	.70	.78	.71	.86	.82	.90	.87
Variable Rotation	.94	.92	.95	.93	.87	.83	.90	.87	.98	.97	.92	.90
Translate	.98	.97	.97	.96	.92	.89	.95	.93	.99	.99	.96	.95
Camera motion	.89	.85	.89	.85	.86	.81	.88	.84	**.95**	**.93**	.94	.92
Sampling	.97	.96	.97	.96	.94	.92	.96	.95	.97	.96	.96	.95
Reversing	.97	.96	.97	.96	.94	.92	.96	.95	.97	.96	.95	.94
Jumbling	.98	.97	.97	.96	.92	.89	.93	.91	.96	.95	.93	.91
Box Jumbling	.99	.99	.98	.97	.97	.96	.97	.96	.98	.97	.95	.94
Freezing	.97	.96	.97	.96	.96	.95	.96	.95	.99	.99	.97	.96
Temporal	**.98**	**.97**	.97	.96	.95	.93	.96	.94	.97	.96	.96	.95

Table 2. Absolute and relative robustness scores averaged across all severity levels for all categories of perturbations for Kinetics-400P dataset. For each category an average is also shown at the end of all sub-categories. The best models are marked as BOLD for both relative and absolute robustness for each perturbation and their categories.

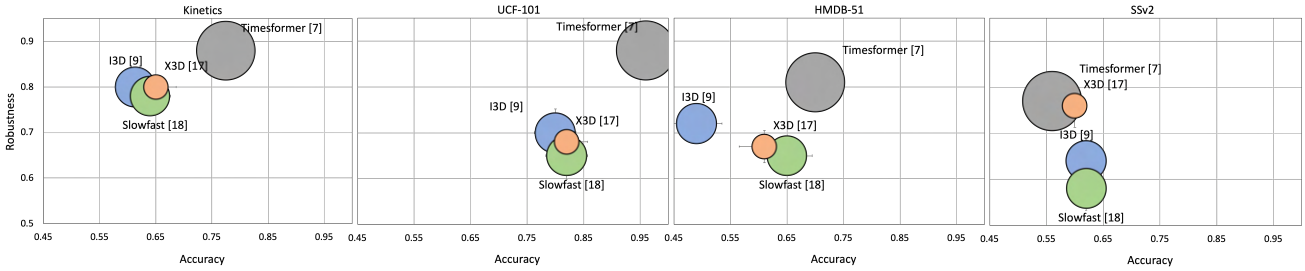

Figure 8. A performance and robustness visualization of pre-trained action recognition models on Kinetics-400P, UCF-101P, HMDB-51P, and SSv2P. The y-axis: relative robustness γ^r (higher is better), x-axis: accuracy on clean videos, and the size of circle indicates FLOPs. The transformer based model is consistently the most robust model across datasets but at the expense of a higher number of FLOPs.

shifts that can be categorized into 15 categories (see Figure 17). Table 6 shows the defined mappings and the number of clips for each category. A total of 536 unique videos were collected from YouTube and split into a total of 4,708 clips. While there are many clips per some videos, we do confirm that models will give different results for each clip for a long video. For example on video-id $v4TFEL3lPhg$, models X3D and R3D correctly classify "HighJump" 2, MViT 14 and ResNet50 9 of 31 clips. Another example of a longer video $oEm64FFEKnc$, the MViT model classifies the activity "Haircut" correctly for 32 of the 76 clips while ResNet50 classified 18.

At most two distribution-shift specific search terms are concatenated to the class names at random to form a search query, which is then used to search YouTube nd retrieve all the search results. Miscellaneous search terms such as "prank", "reaction", "unusual", etc have also been added at random to the search queries. These search results are then filtered to only download the videos with length less

shifts/

	I3D		SF		X3D		Timesformer	
Defocus Blur	.88	.81	.87	.78	.88	.80	.83	.69
Motion Blur	.79	.66	.74	.58	.86	.77	.85	.73
Zoom Blur	.90	.84	.95	.92	.96	.94	.89	.80
Blur	.85	.76	.85	.76	**.90**	.67	.85	**.74**
Gaussian Noise	.64	.42	.45	.12	.74	.57	.70	.46
Shot Noise	.71	.53	.61	.38	.88	.81	.86	.75
Impulse Noise	.60	.36	.43	.08	.72	.54	.68	.43
Speckle Noise	.64	.43	.56	.30	.87	.78	.83	.70
Noise	.63	.40	.51	.22	**.80**	**.67**	.78	.59
JPEG	.74	.58	.65	.44	.86	.76	.85	.74
MPEG1	.72	.54	.73	.57	.90	.84	.88	.78
MPEG2	.63	.41	.62	.39	.84	.73	.82	.67
Digital	.69	.51	.67	.48	**.86**	**.78**	.85	.73
Rotate	.84	.74	.85	.76	.83	.72	.84	.71
Var Rotate	.56	.30	.44	.10	.63	.39	.85	.73
Translate	.95	.92	.96	.93	.96	.93	.96	.92
Camera	.78	.65	.74	.59	.80	.67	**.88**	**.78**
Sampling	.75	.60	.78	.64	.93	.88	.99	.98
Reverse Sampling	.52	.23	.51	.22	.55	.26	.70	.46
Jumble	.73	.56	.74	.58	.93	.88	.99	.98
Box Jumble	.86	.77	.68	.48	.64	.40	.85	.74
Freeze	.65	.43	.68	.49	.79	.65	.86	.76
Temporal	.69	.50	.68	.48	.77	.61	**.89**	**.78**

Table 3. Absolute and relative robustness scores averaged across all severity levels for all categories of perturbations for SSv2P dataset. For each category an average is also shown at the end of all sub-categories. The best models are marked as BOLD for both relative and absolute robustness for each perturbation and their categories. The TimeSformer architecture shows significantly higher robustness scores as compared to the CNN-based architectures while the more appearance based perturbations there is variation between the X3D and Timesformer architecture.

than 60 seconds and height and width dimensions of at least 256x256. These videos have been manually analysed and cleaned These videos are then trimmed into smaller videos that are less than 10 seconds each. Additional caution has been taken by manually verifying each video to consist the ground truth data.

In Figure 2, we have presented the performance of different models trained with and without augmentation on this real-world dataset. We observe that while CNN models benefits from augmentations, transformer based model MViT trained on clean videos performs the best, not showing any benefits from data augmentations.

References

[1] Dan Hendrycks, Andy Zou, Mantas Mazeika, Leonard Tang, Bo Li, Dawn Song, and Jacob Steinhardt. Pixmix: Dreamlike pictures comprehensively improve safety measures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16783–16792, 2022. 1, 2, 5

	R3D-S		R3D-P		I3D-S		I3D-P		SF-S		SF-P		X3D-S		X3D-P		MViT-S		MViT-P		Times-P	
Defocus Blur	.63	.37	.53	.43	.74	.56	.54	.42	.72	.61	.55	.47	.84	.73	.60	.51	.74	.63	.80	.79	.79	.78
Motion Blur	.74	.56	.63	.55	.75	.57	.68	.60	.65	.51	.57	.50	.82	.69	.62	.54	.71	.58	.73	.62	.88	.88
Zoom Blur	.77	.61	.63	.68	.79	.64	.75	.68	.82	.74	.83	.80	.94	.89	.84	.81	.88	.82	.88	.87	.84	.83
Blur	.71	.51	.63	.55	.76	.59	.66	.57	.73	.62	.65	.59	**.86**	.76	.69	.62	.78	.67	.81	.79	.84	**.83**
Gaussian	.57	.28	.51	.41	.64	.39	.61	.50	.52	.33	.47	.38	.59	.30	.33	.18	.78	.68	.80	.79	.62	.61
Shot	.75	.58	.79	.74	.83	.71	.83	.78	.73	.63	.76	.72	.78	.62	.56	.46	.94	.91	.96	.96	.92	.92
Impulse	.55	.23	.45	.33	.59	.31	.57	.45	.57	.25	.43	.33	.58	.28	.31	.16	.76	.65	.81	.80	.60	.59
Speckle	.72	.52	.75	.70	.79	.63	.80	.75	.70	.57	.71	.66	.74	.55	.54	.42	.91	.87	.96	.95	.90	.90
Noise	.65	.40	.62	.55	.71	.51	.70	.62	.60	.45	.59	.52	.67	.44	.43	.31	.85	.78	**.88**	**.87**	.76	.75
JPEG	.97	.96	.74	.69	.98	.97	.78	.72	.93	.90	.82	.79	.97	.94	.90	.88	.97	.95	.90	.88	.91	.91
MPEG1	.89	.82	.70	.64	.91	.85	.69	.61	.90	.85	.72	.67	.96	.93	.88	.86	.95	.93	.84	.83	.93	.93
MPEG2	.88	.80	.70	.63	.91	.84	.66	.57	.88	.84	.71	.65	.96	.84	.87	.85	.95	.93	.80	.79	.91	.91
Digital	.92	.86	.71	.65	.93	.89	.71	.63	.90	.86	.75	.70	.96	.93	.89	.86	**.96**	**.94**	.84	.83	.92	.91
Rotate	.75	.57	.67	.61	.76	.59	.63	.53	.63	.62	.68	.63	.77	.61	.68	.60	.80	.71	.89	.88	.88	.88
Var Rotate	.88	.79	.85	.82	.85	.74	.81	.76	.57	.39	.63	.57	.75	.58	.76	.71	.63	.45	.87	.71	.97	.97
Translate	.92	.87	.87	.84	.88	.8	.88	.84	.67	.53	.70	.65	.82	.70	.87	.84	.65	.48	.92	.93	.98	.98
Camera	.85	.74	.80	.76	.83	.71	.77	.71	.65	.51	.67	.62	.78	.63	.77	.72	.69	.55	.90	.89	**.95**	**.95**
Sampling	.94	.90	.95	.94	.94	.90	.95	.94	.86	.81	.87	.85	.92	.86	.91	.89	.80	.72	.93	.93	.98	.98
Reversing	.94	.90	.95	.94	.94	.9	.95	.94	.86	.81	.87	.85	.92	.86	.91	.89	.80	.72	.93	.93	.98	.98
Jumbling	.96	.94	.97	.96	.96	.93	.97	.97	.78	.70	.84	.81	.93	.88	.92	.91	.80	.70	.94	.93	.98	.98
Box Jumbl	.98	.96	.98	.98	.98	.95	.98	.97	.95	.92	.96	.95	.98	.94	.98	.97	.95	.91	.98	.96	.97	.97
Freezing	.96	.91	.95	.93	.95	.91	.97	.96	.91	.87	.92	.96	.96	.93	.91	.89	.79	.69	.91	.89	.98	.98
Temporal	.95	.92	.96	.95	.95	.92	.97	.96	.86	.81	.89	.87	.94	.90	.93	.91	.81	.73	.94	.93	**.98**	**.98**

Table 4. Performance of all the models for various perturbation categories when trained from scratch and using pre-trained weights on UCF-101P dataset. S indicates training from scratch and P indicates using pre-trained weights. We use pre-trained weights from Kinetics-400 for all the models. Red values are for the absolute robustness while blue values are for relative robustness. Bold values are the best models while underlined are the second best models.

	R3D-S		R3D-P		I3D-S		I3D-P		SF-S		SF-P		X3D-S		X3D-P		MViT-S		MViT-P		Times-P	
Defocus Blur	.83	.34	.95	.45	.89	.52	.73	.45	.85	.54	.82	.57	.94	.82	.68	.48	.93	.81	.79	.70	.81	.73
Motion Blur	.87	.51	.75	.57	.91	.63	.80	.59	.84	.53	.65	.46	.89	.62	.67	.45	.87	.63	.73	.62	.85	.79
Zoom Blur	.88	.56	.81	.67	.93	.70	.82	.64	.91	.73	.83	.74	.83	.38	.81	.69	.96	.87	.87	.81	.75	.64
Blur	.86	.47	.84	.56	**.91**	.62	.78	.56	.87	.6	.73	.59	.89	.61	.72	.54	**.92**	**.77**	.80	.71	.80	.72
Gaussian	.84	.38	.63	.36	.86	.40	.72	.42	.84	.50	.59	.37	.82	.35	.51	.21	.92	.77	.88	.83	.65	.50
Shot	.92	.68	.81	.67	.93	.69	.88	.76	.96	.86	.82	.72	.91	.68	.69	.50	.98	.93	.97	.96	.86	.80
Impulse	.83	.33	.60	.32	.85	.35	.68	.36	.81	.42	.55	.30	.83	.40	.51	.19	.91	.73	.87	.82	.73	.47
Speckle	.91	.66	.81	.68	.97	.88	.89	.78	.95	.84	.79	.68	.92	.73	.68	.47	.97	.92	.97	.95	.85	.79
Noise	.90	.51	.71	.51	.9	.58	.79	.58	.89	.56	.69	.52	.87	.54	.60	.34	**.94**	.84	**.92**	**.89**	.75	.64
JPEG	.89	.97	.93	.88	.89	.97	.97	.93	.99	.97	.91	.86	.98	.94	.93	.88	.98	.94	.93	.90	.89	.84
MPEG1	.93	.72	.89	.81	.94	.75	.93	.85	.93	.78	.9	.85	.94	.90	.89	.82	.93	.79	.88	.83	.91	.87
MPEG2	.94	.78	.89	.82	.97	.89	.93	.84	.93	.80	.90	.84	.94	.78	.88	.81	.98	.93	.88	.82	.9	.86
Digital	.95	.82	.90	.84	**.97**	**.87**	.94	.87	.95	.85	.90	.85	.95	.13	.90	.84	.96	.89	.90	.85	.90	.86
Rotate	.92	.71	.76	.59	.94	.74	.80	.60	.90	.69	.79	.68	.93	.75	.76	.62	.93	.80	.91	.87	.82	.74
Var Rotate	.96	.87	.93	.88	.95	.78	.82	.63	.82	.46	.73	.59	.90	.65	.87	.78	.81	.42	.83	.75	.96	.94
Translate	.92	.69	.94	.90	.91	.61	.97	.93	.81	.40	.76	.62	.88	.58	.91	.85	.77	.32	.89	.84	.96	.94
Camera	**.93**	.76	.88	.79	.93	.72	.86	.72	.84	.52	.76	.63	.91	.66	.85	.82	.84	.52	.88	.82	.91	**.87**
Sampling	.93	.73	.89	.82	.93	.70	.92	.84	.84	.52	.76	.63	.90	.64	.85	.75	.85	.55	.81	.53	.97	.96
Reversing	.93	.73	.89	.82	.93	.70	.92	.84	.84	.52	.76	.63	.90	.64	.85	.75	.85	.55	.81	.53	.97	.96
Jumbling	.96	.87	.92	.87	.96	.81	.96	.93	.85	.52	.74	.58	.93	.76	.84	.75	.82	.46	.81	.72	.96	.94
Box Jumbl	.98	.93	.98	.97	.98	.90	.98	.96	.92	.76	.91	.86	.96	.86	.96	.93	.94	.81	.95	.92	.97	.96
Freezing	.96	.85	.92	.86	.95	.78	.95	.91	.92	.77	.89	.83	.96	.85	.89	.82	.89	.67	.89	.84	.94	.91
Temporal	.95	.82	.92	.87	.95	.80	.95	.90	.87	.62	.81	.71	.93	.75	.88	.80	.87	.60	.85	.70	**.97**	**.95**

Table 5. Performance of all the models for various perturbation categories when trained from scratch and using pre-trained weights on HMDB-51P dataset. S indicates training from scratch and P indicates using pre-trained weights. We use pre-trained weights from Kinetics-400 for all the models. Red values are for the absolute robustness while blue values are for relative robustness. Bold values are the best models while underlined are the second best models.

Figure 9. The embedding space for a sample of models on SSv2 for five classes and their respective opposite based on the direction of time. The first row is without any perturbations while the remainder are reverse sampling, jumble and freeze at severity 4.

Table 6. A summary of the distribution shifts we used to collect videos for UCF101-DS and their corresponding high-level category.

Category	Distribution Shift	Number of Clips
Actor	[animal, costume, toy]	420
Age	[kids, old_person]	47
Behavioral	[caught_on_cam!, prank, reaction, scary]	24
Crowd	[crowd]	176
Ethnicity	[african, asian, black, indian_brown]	50
Indoor Scenery	[at_home, at_the_club, at_the_gym, indoor, indoors, in_court, in_garage, mirror]	96
Lighting	[low_light, at_late_night, at_night, dark, low_light_conditions]	616
Obscure	[unsual, unusual]	378
Occluded	[obstructed, obstructed_view]	59
Outdoor Scenery	[at_the_beach, desert, in_backyard, in_garden, in_the_fields, on_the_road, outdoors, outside, underwater]	777
POV	[camera_angle, camera_angles, go_pro, on_TV, pov, pov_at_night, shaky, tutorial, upside_down]	992
Speed	[alow_mo, fastest, slowmotion, slow_mo]	251
Style	[animated, animation, filter, text_on_screen, vintage]	535
Weather	[fog, in_rain, muddy, rain, snow]	287

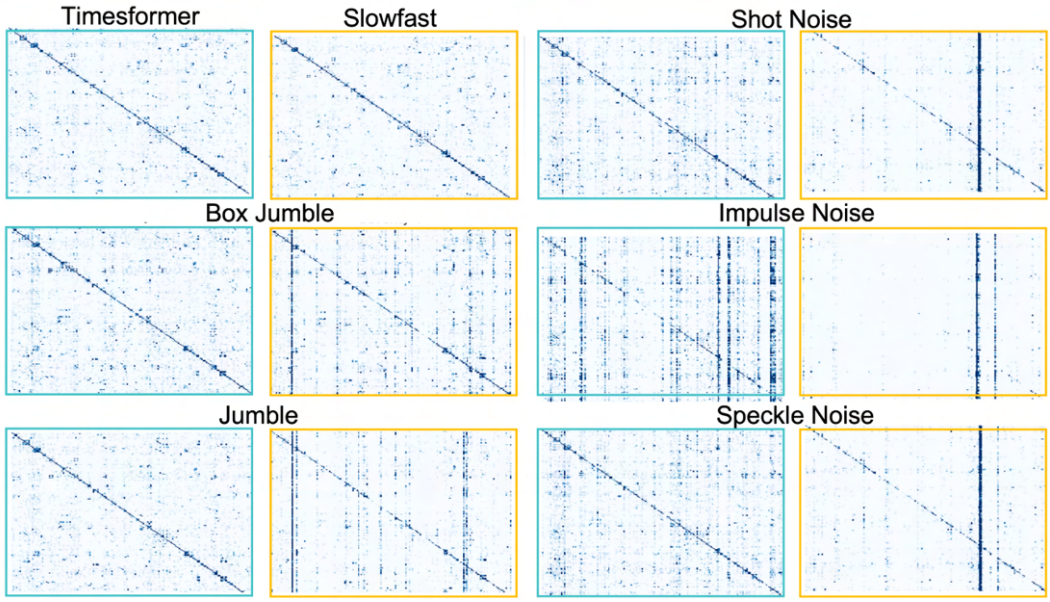

Figure 10. Confusion Matrices for the SSv2 dataset for box jumble, jumble, shot noise, impulse noise, and speckle noise perturbations at severity 4. When models fail, they are often predicting a smaller selection of classes for a majority of the samples. These classes also differ between whether the perturbations are appearance based or temporal based.

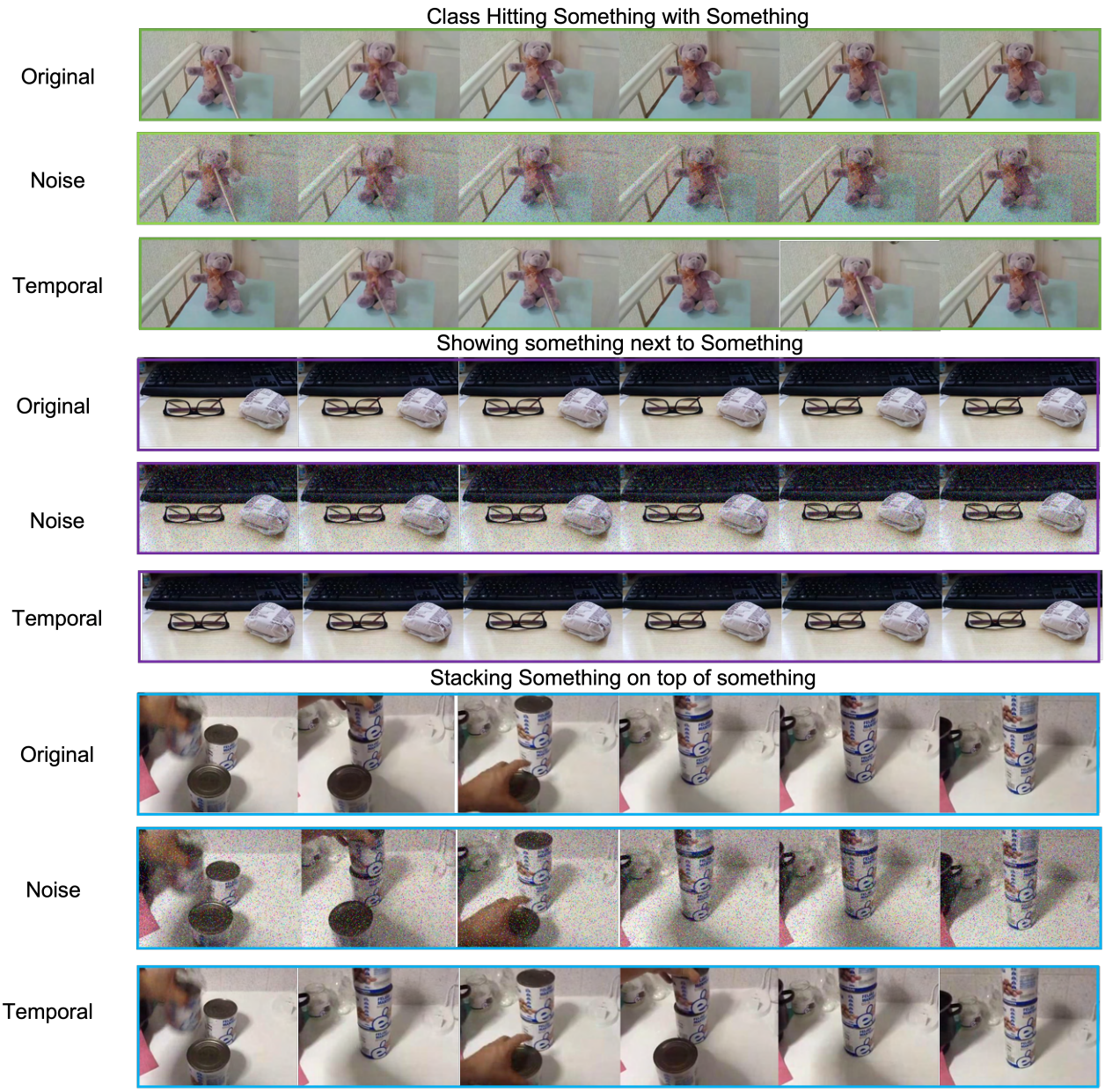

Figure 11. Examples of the most predicted classes for either temporal or noise perturbations. For noise, the CNN based model Slowfast predicts “Showing something next to something” for a majority of samples while for temporal perturbations it predicts “Hitting something with something” and “Stacking number of something”.

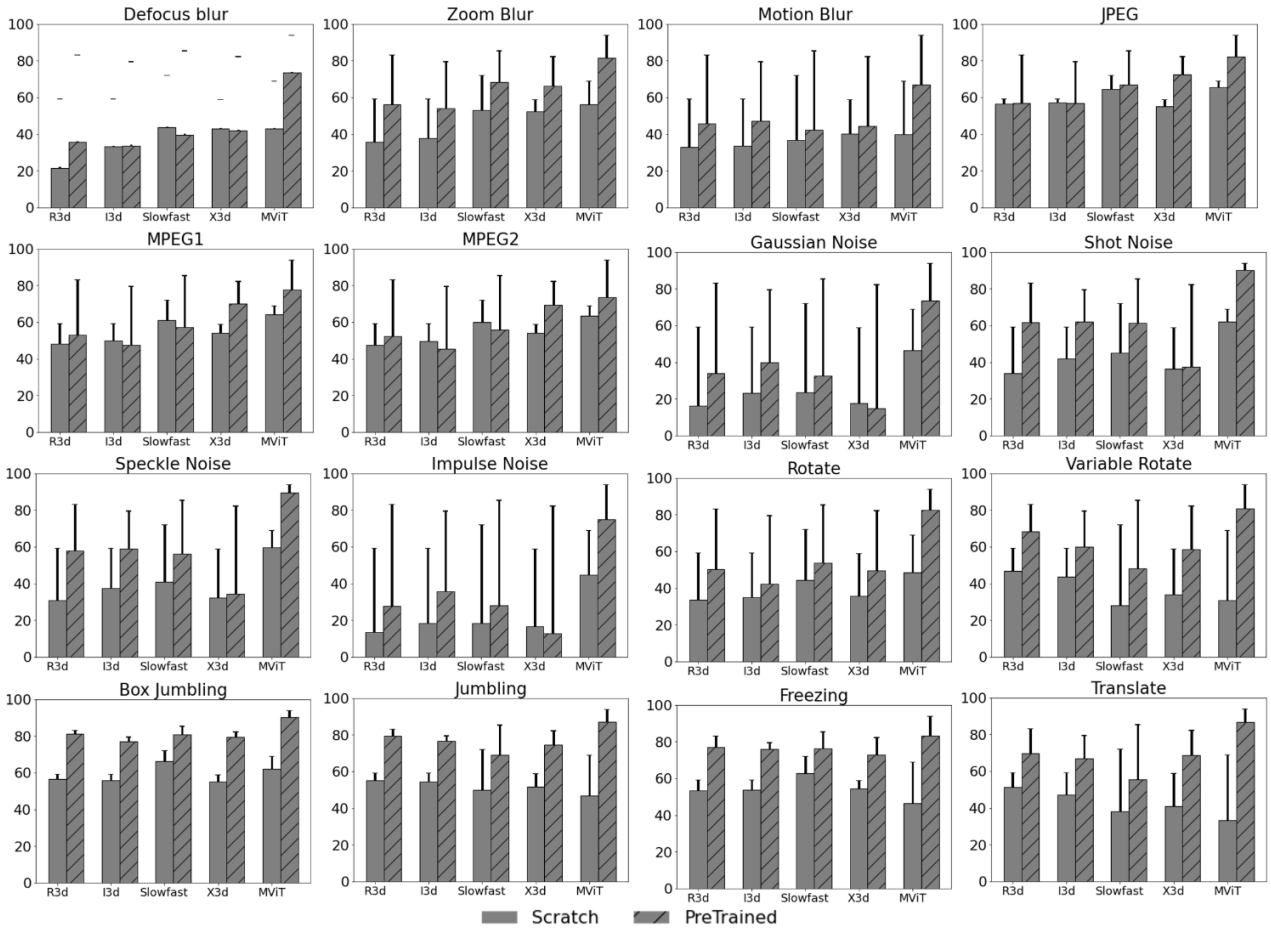

Figure 12. A comparison of model robustness against different perturbations with pretrained and scratch training on UCF-101P benchmark. The plain bar represents performance without pretrained weights and striped bar represents a pretrained model. The top extension indicates drop in performance in comparison with accuracy on clean videos.

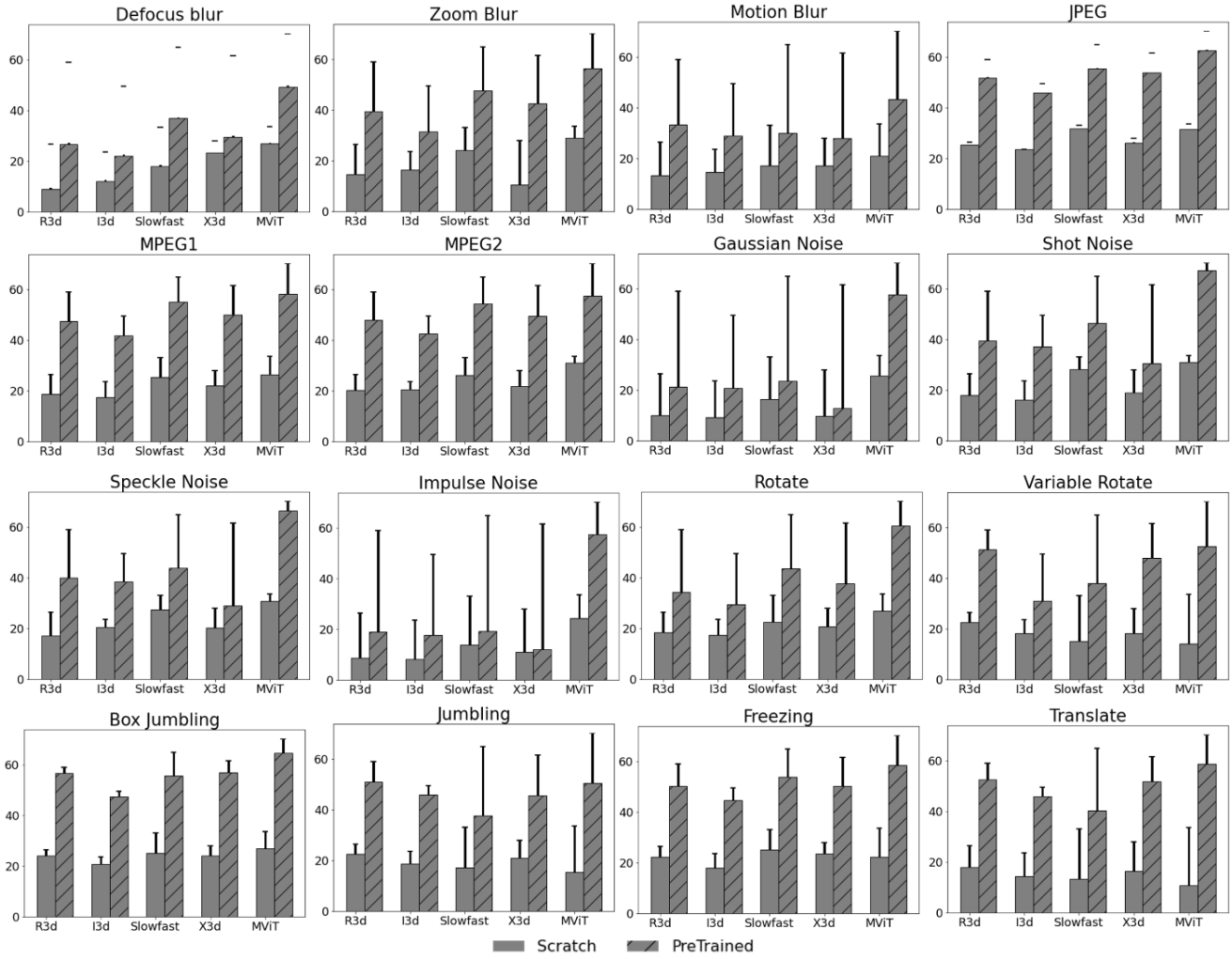

Figure 13. A comparison of model robustness against different perturbations with pretrained and scratch training on HMDB-51P benchmark. The plain bar represents performance without pretrained weights and striped bar represents a pretrained model. The top extension indicates drop in performance in comparison with accuracy on clean videos.

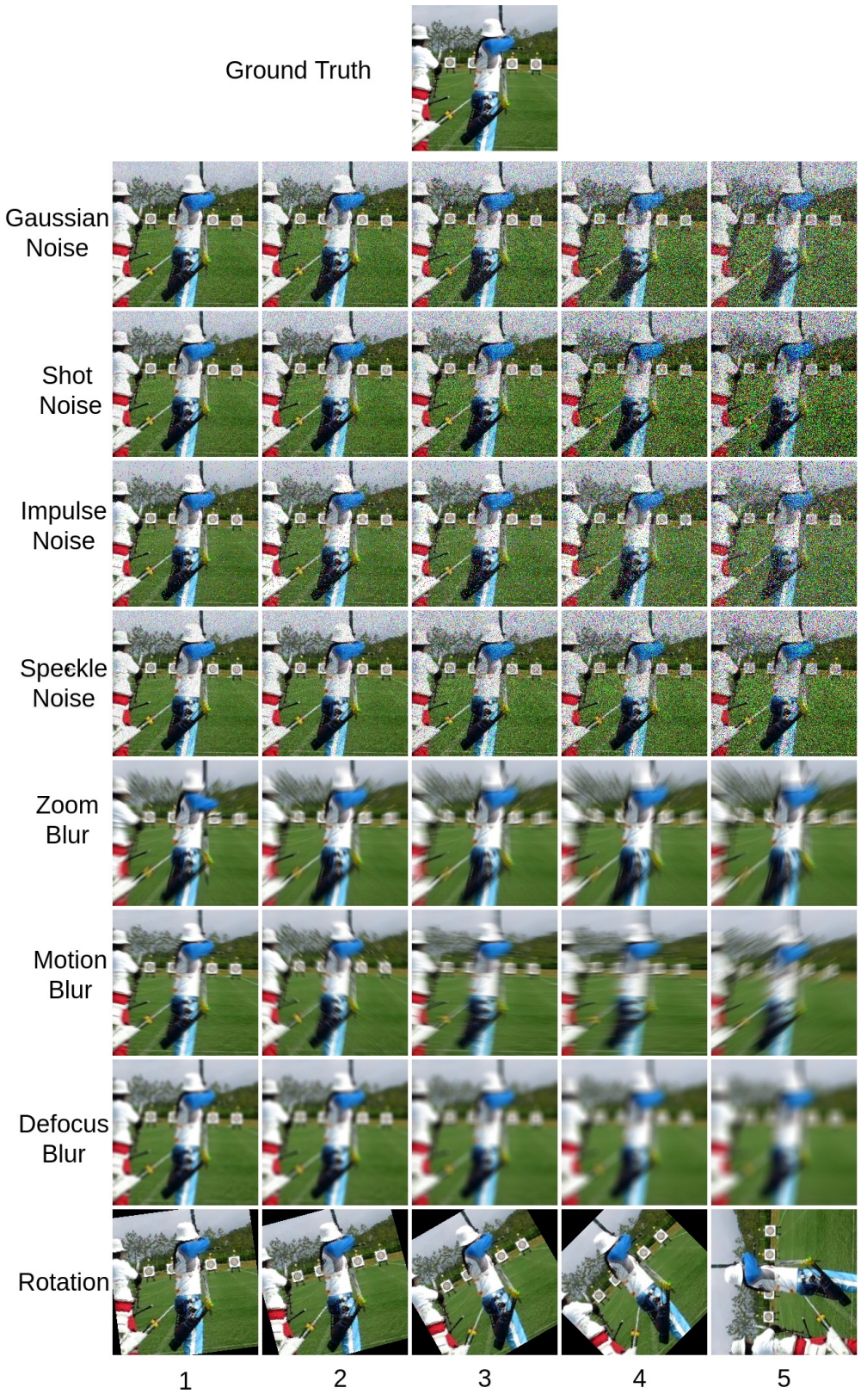

Figure 14. Sample video frames from Kinetics-400P showing different severity levels of some spatial perturbations(severity increases from left to right) .

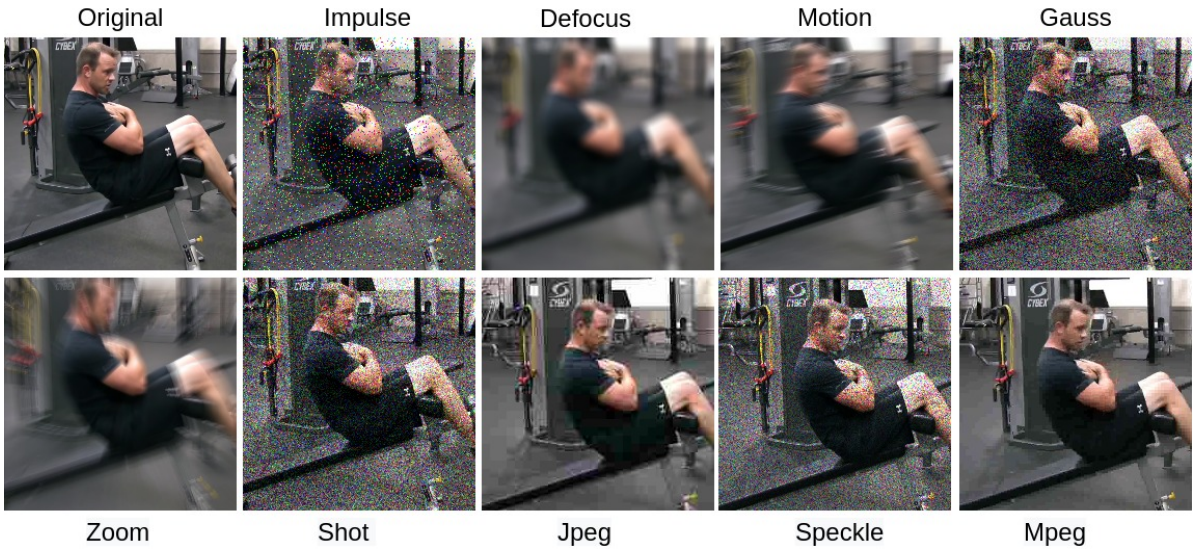

Figure 15. Sample video frame from Kinetics-400P showing different perturbations using blur and noise.

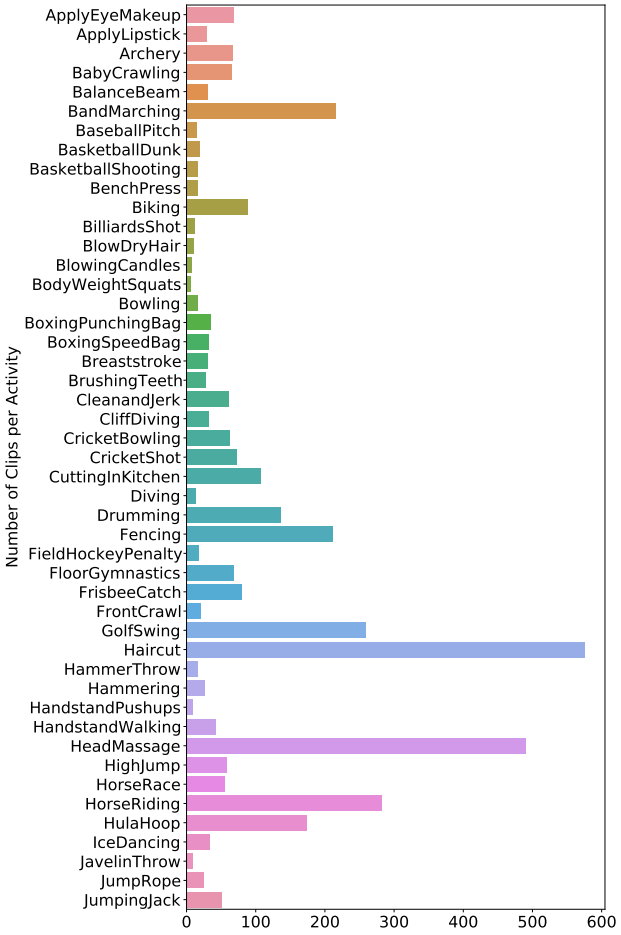

Figure 16. The number of clips per UCF101 activity.

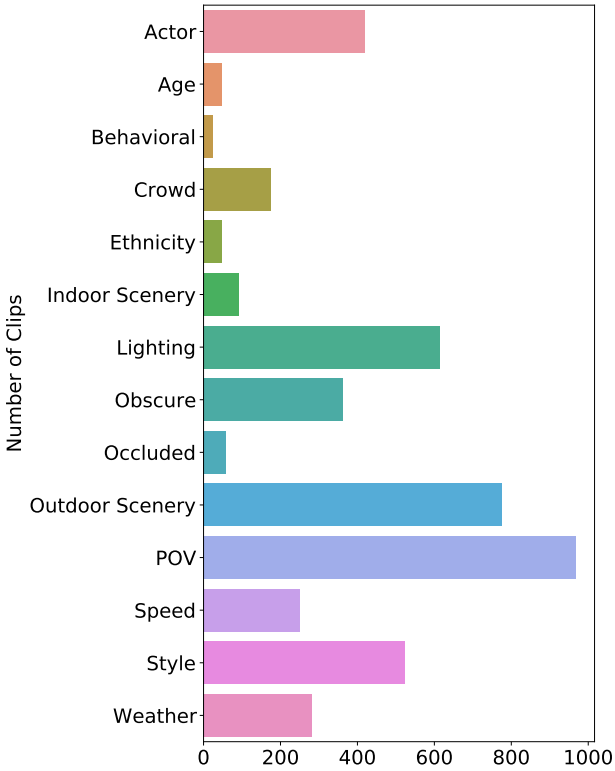

Figure 17. The number of clips per distribution shift category.