Beyond the Pixels: Exploring the Effect of Video File Corruptions on Model Robustness

Trenton Chang, Daniel Y. Fu, Yixuan Li, Christopher Ré
Stanford University, Stanford, CA
{tchang97, danfu, sharonli, chrismre}@cs.stanford.edu

Abstract

Recent work has studied the robustness of computer vision models on video data with pixel-based perturbations. However, videos are susceptible to non-pixel corruptions, such as file corruptions*, which arise from hardware errors, I/O errors, and malware. In this paper, we investigate the effect of video file corruptions on model robustness. We find that file corruptions cause performance drops of up to 77.1% on standard datasets like HMDB51 and UCF101. We analyze the effects of file corruptions qualitatively and quantitatively, characterizing the types of visual artifacts that file corruptions cause. We measure visual artifact severity with pixel-space Euclidean distance and observe under the same level of file corruption, incorrectly-classified examples are up to* 1.57 *times more corrupted than correctly-classified examples.*

1. Introduction

As video becomes an increasingly popular data modality, with applications in action recognition [16, 28], event detection [9, 26] and autonomous vehicles [2, 5], recent work in computer vision has examined the robustness of video-based machine learning (ML) models. Such work has focused on pixel-based visual corruptions (i.e. adversarial attacks [14, 33]), but video data is susceptible to file corruptions, which can arise from hardware errors, I/O errors, or malware [27, 35]. These corruptions can leave visible artifacts, like freeze-frames, smearing effects, and more, which can harm model inference. File corruptions remain under-explored in the model robustness literature.

In this paper, we go beyond pixel-space corruptions and explore the effect of file corruptions on model robustness. File corruptions are challenging to quantify, as video pixels are transformed from pixel space to a compressed non-pixel space for storage, introducing an extra degree of complexity to the video model robustness problem. We simulate two file corruption patterns: *random corruption*, where random bits throughout the video file are flipped, and *contiguous corruption*, where a segment of video bitstream is replaced with random bits (see Fig. 1).

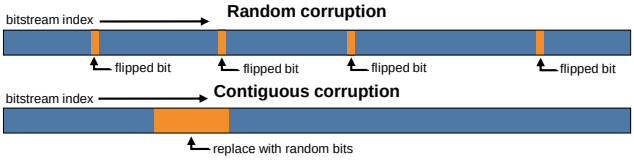

Figure 1. A visualization of contiguous (top) vs. random (bottom) corruptions.

We evaluate model performance on corrupted videos in action recognition benchmarks, and find test-time model accuracy drops up to 68.9% on HMDB51 [16] and 77.1% on UCF101 [28] under file corruptions. To study this drop, we present qualitative and quantitative analyses of the effects of file corruption on videos. Qualitatively, we notice that incorrectly classified videos exhibit more severe pixel-space perturbations than correctly classified videos. Quantitatively, we measure the severity of these artifacts via pixel-space Euclidean distance, finding that incorrectly classified examples are up to 1.57 more perturbed than correct examples at the same level of file corruption. Our results suggest that file corruptions may be a credible threat to video ML models, and that pixel-space metrics are a suitable proxy for the effects of file corruption.

2. Related Work

Compression-aware robustness studies. Previous model robustness work in computer vision has studied the effects of various components of image and video compression, such as compression rates or encoding schemes [25, 29]. In the image domain, the effects of JPEG compression on model robustness have been studied extensively [1, 6, 8, 11, 12, 36]. We view our work as complementary to prior efforts, extending robustness studies to include file corruptions.

Presented at European Conference on Computer Vision (ECCV) Workshop on Adversarial Robustness in the Real World (AROW), 2020.

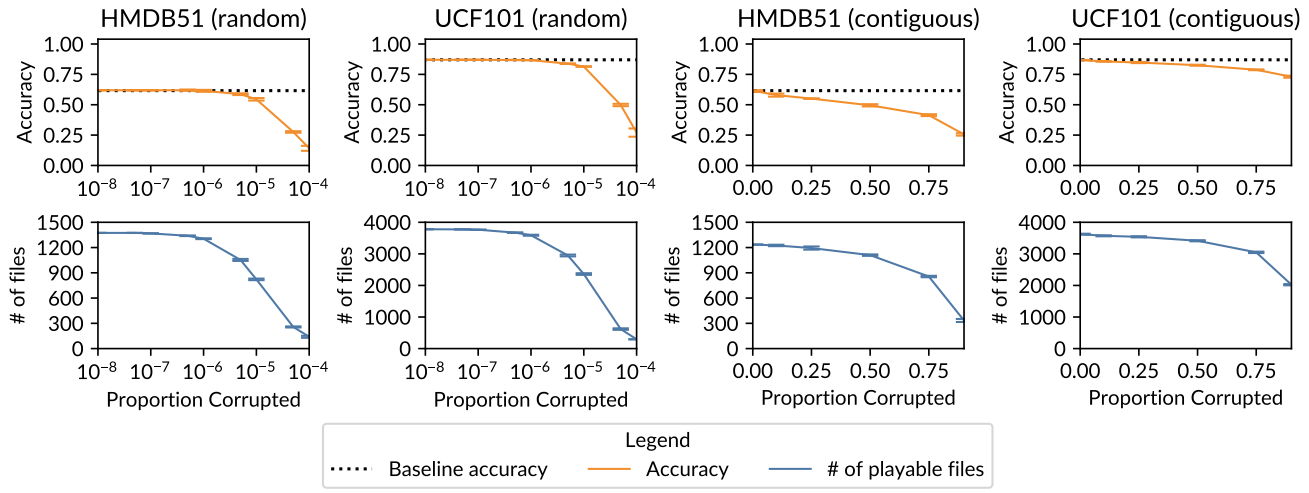

Figure 2. The number of playable files and model accuracy based on proportion of video corrupted, for random and contiguous corruptions on UCF101 and HMDB51. Error bars show standard deviation across five random seeds.

Adversarial examples in video action recognition. Previous work has extended white-box adversarial attacks on image models [3, 10, 18, 19, 30] to video action recognition models [13, 20, 33]. Jiang and Ma et al. [14] also devise a black-box adversarial attack against video models. Video file corruptions can likewise be viewed as a source of real-world adversarial examples in non-pixel space.

Robustness to perturbations. Goodfellow et. al. [10] introduces adversarial training as a defense against adversarial input, a method extended in [17, 24, 31]. Other defenses include certificate-based methods [22, 23] and self-supervision [4]. Imagenet-C and Imagenet-P [12] are ImageNet [7] extensions used for benchmarking robustness on image corruptions. Furthermore, [32, 34] investigate convolutional network robustness in the frequency domain. Defenses against video file corruptions are beyond the scope of this paper, but we view the extension of robustness techniques to file corruptions as exciting future work.

3. Simulating File Corruptions

File corruptions, a type of file system error, are erroneous changes to bits in a file. We simulate two types of file corruptions, *random corruptions* and *contiguous corruptions* (Fig. 1). In random corruptions, random bits are flipped throughout the file. Such bit-flips can occur as the result of hardware integrity issues like bus errors [35], malicious software [27], or cosmic rays [21]. In contiguous corruptions, a segment of a video bitstream is replaced with random bits. These errors commonly occur due to sudden temperature changes (thermal asperity), malware [27], or firmware writing data in the wrong location [35].

We vary the proportion of video file corrupted in our experiments, notated as $p \in [0, 1]$. For random corruptions, we flip each bit independently with probability p. For contiguous corruptions, we replace a random contiguous segment of length p times the file length with random bits.

4. Experiments

We study the impact of random and contiguous file corruptions on model robustness in action recognition. First, we find that file corruptions make many videos unplayable and degrade model performance by up to 77.1% (Section 4.1). Next, we qualitatively analyze the causes of this drop by visualizing corrupted videos, observing that misclassified videos exhibit more severe visual artifacts than correctly classified videos (Section 4.2). Finally, we quantitatively analyze the severity of visual perturbations using pixel-space Euclidean distance (Section 4.3).

Datasets and model. We fine-tune a pre-trained 3D-Resnet18 on the HMDB51 [16] and UCF101 [28] action recognition benchmarks, using a standard training setup [15]. For evaluation, we split the input clip into 16-frame segments, outputting the action class with the highest probability averaged over all segments. Before applying file corruption, we transcode videos to use the H.264 codec and an `.mp4` container.

4.1. File Playability and Model Performance

To study the impact of file corruption, we plot model accuracy on playable videos (videos that can be opened after corruption) and the number of playable videos under varying proportions of random and contiguous corruptions (Fig. 2). We find that as corruption proportion increases, more files are unplayable: 90.8% of files in HMDB51 and 93.5%

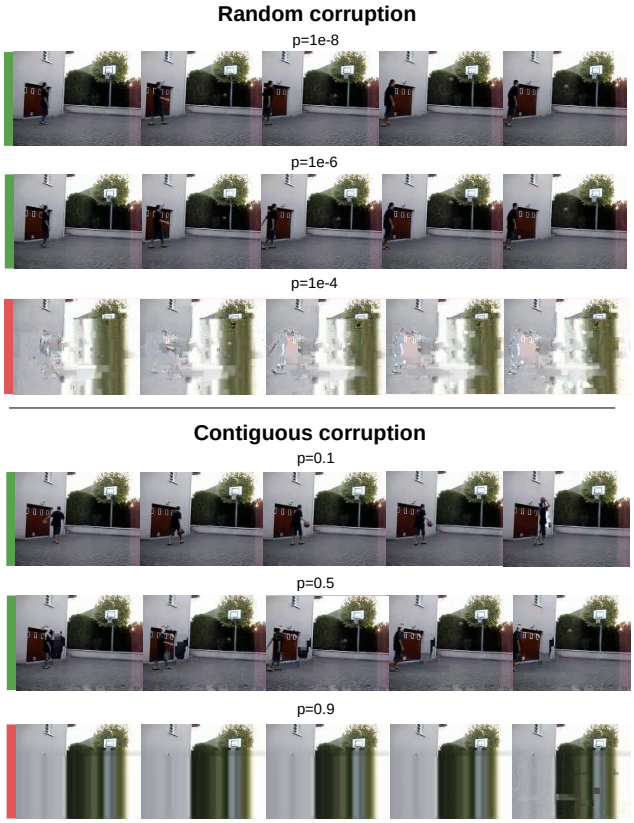

Figure 3. Frames from a clip (class Basketball) at varying corruption levels. Color shows correct vs. incorrect classification.

of files in UCF101 are unable to be loaded at $p = 1e-4$; for contiguous corruptions, 46.5% of files on UCF101 and 78.2% on HMDB51 are unplayable at $p = 0.9$.

We also find that file corruption degrades model performance on playable videos. On random corruptions, accuracy drops at corruption proportions exceeding $p = 1e-6$. For contiguous corruptions, accuracy degrades at corruption proportions over $p = 0.01$. For random corruptions, performance drops up to 68.9% on HMDB51 and 77.1% on UCF101; for contiguous corruptions, performance drops up to 58.5% on HMDB51 and 16.0% on UCF101.

Note that the curves for the number of playable files and accuracy for random corruptions degrade exponentially, while they degrade linearly for contiguous corruptions (Fig. 2). This happens because random corruptions are spread throughout the video bit-stream (Fig.1), while contiguous corruptions are localized, making random corruptions more likely to render a file unusable by destroying information segments necessary to decode a video file.

4.2. Qualitative Analysis of File Corruptions

To characterize the effects of file corruption on model performance, we qualitatively analyze file corruption by visualizing corrupted videos. Fig. 3 shows the same clip under varying corruption levels. At the highest corruption

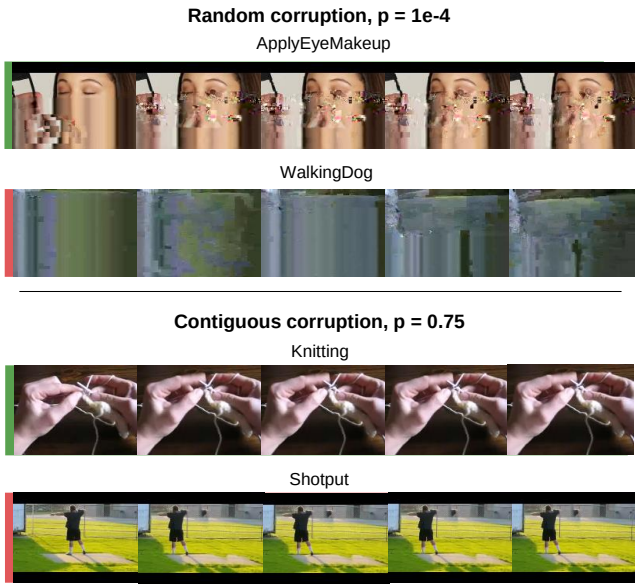

Figure 4. Frames from correctly and incorrectly classified examples, paired by corruption proportion and mode (random or contiguous).

proportions (random, $p = 1e-4$; contiguous, $p = 0.9$), only the topmost parts of the original frames are unperturbed. Furthermore, the contiguous example ($p = 0.9$) exhibits a freeze-frame effect, as the corrupted frames are nearly identical, while the randomly corrupted example ($p = 1e-4$) features moving noisy patches in the center-left of the frames. In both corruption modes, as visual artifacts worsen, the model prediction tends to become incorrect (red bars), which happens here at proportions higher than $p = 1e-6$ for random corruptions and $p = 0.5$ for contiguous corruptions.

The correlation between visual artifacts and incorrect model predictions holds on videos corrupted with the same corruption proportion and strategy. Fig. 4 shows two such pairs of clips (additional examples in Appendix A). The top pair of clips underwent random corruption ($p = 1e-4$). In the ApplyEyeMakeup clip (correctly classified), the eye and the makeup applicator are distorted but visible, whereas the WalkingDog clip (incorrectly classified) is unrecognizable. The bottom pair of clips experienced contiguous corruption ($p = 0.75$), resulting in a freeze-frame effect. In the Knitting clip (correctly classified), some movement remains between the 1st and 2nd frames from the left, while the Shotput example (incorrectly classified) is entirely static.

4.3. Quantitative Analysis of File Corruptions

We measure the severity of the visual artifacts observed in Section 4.2 using pixel-space Euclidean distance, calculated as the average Euclidean norm between pixels in corrupted vs. uncorrupted clips. Intuitively, this metric corresponds to how much each pixel in a video changed under

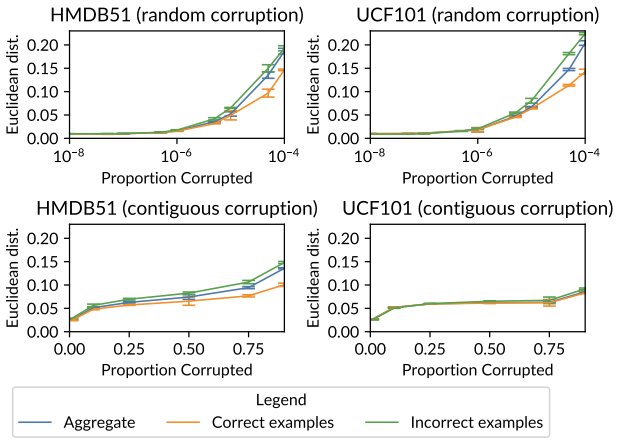

Figure 5. Pixel-space Euclidean distance vs. proportion corrupted, averaged over the entire dataset (blue), correctly-classified examples (orange), and incorrectly-classified examples (green). Error bars show standard deviation across five random seeds.

file corruption. In the case that the lengths of the clips differ (i.e., due to dropped frames), we truncate the original clip to match the corrupted clip.

Fig. 5 shows the average pixel-space Euclidean distance between corrupted and uncorrupted clips on the entire dataset (blue), on correctly-classified examples (orange), and on incorrectly-classified examples (green). On average, incorrect examples have higher Euclidean distance than correct examples, confirming our qualitative observations from Section 4.2. For random corruptions, incorrect examples are up to 1.57 times more perturbed than correct examples on average (UCF101, random corruption, top right), and for contiguous corruptions, incorrect examples are up to 1.47 times more perturbed on average (HMDB51, contiguous corruption, bottom left). This suggests that visual perturbation severity under file corruption contributes to the accuracy drop seen in Fig. 2.

5. Conclusion and Future Work

In this paper, we take a first step in investigating the effect of file corruptions on video model robustness, finding that file corruptions can result in a significant accuracy drop. We present a qualitative and quantitative analysis on the factors contributing to this drop in performance. Our results suggest that pixel-space distortions are a suitable proxy to measure the effects of file corruption. In the future, we plan to study other non-pixel space corruptions like network-level corruptions, extend our findings to other video architectures, and create defenses for non-pixel space corruptions. We hope this work motivates further exploration of non-pixel space corruptions in video data.

Acknowledgements

We thank Jared Dunnmon, Karan Goel, Sarah Hooper, and Laurel Orr for helpful discussions and feedback on drafts. We gratefully acknowledge the support of DARPA under Nos. FA86501827865 (SDH) and FA86501827882 (ASED); NIH under No. U54EB020405 (Mobilize), NSF under Nos. CCF1763315 (Beyond Sparsity), CCF1563078 (Volume to Velocity), and 1937301 (RTML); ONR under No. N000141712266 (Unifying Weak Supervision); the Moore Foundation, NXP, Xilinx, LETI-CEA, Intel, IBM, Microsoft, NEC, Toshiba, TSMC, ARM, Hitachi, BASF, Accenture, Ericsson, Qualcomm, Analog Devices, the Okawa Foundation, American Family Insurance, Google Cloud, Swiss Re, the HAI-AWS Cloud Credits for Research program, Brown Institute for Media Innovation, Department of Defense (DoD) through the National Defense Science and Engineering Graduate Fellowship (NDSEG) Program, and members of the Stanford DAWN project: Teradata, Facebook, Google, Ant Financial, NEC, VMWare, and Infosys. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views, policies, or endorsements, either expressed or implied, of DARPA, NIH, ONR, or the U.S. Government.

References

[1] Aydemir, A.E., Temizel, A., Temizel, T.T.: The effects of jpeg and jpeg2000 compression on attacks using adversarial examples (2018)

[2] Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L.D., Monfort, M., Muller, U., Zhang, J., et al.: End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316 (2016)

[3] Carlini, N., Wagner, D.: Towards evaluating the robustness of neural networks (2016)

[4] Carmon, Y., Raghunathan, A., Schmidt, L., Liang, P., Duchi, J.C.: Unlabeled data improves adversarial robustness (2019)

[5] Chen, Y., Wang, J., Li, J., Lu, C., Luo, Z., Xue, H., Wang, C.: Lidar-video driving dataset: Learning driving policies effectively. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5870–5878 (2018)

[6] Das, N., Shanbhogue, M., Chen, S.T., Hohman, F., Li, S., Chen, L., Kounavis, M.E., Chau, D.H.: Shield: Fast, practical defense and vaccination for deep learning using jpeg compression (2018)

[7] Deng, J., Dong, W., Socher, R., Li, L., Kai Li, Li Fei-Fei: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 248–255 (2009)

[8] Dziugaite, G.K., Ghahramani, Z., Roy, D.M.: A study of the effect of jpg compression on adversarial images (2016)

[9] Fu, D.Y., Crichton, W., Hong, J., Yao, X., Zhang, H., Truong, A., Narayan, A., Agrawala, M., Ré, C., Fatahalian, K.: Rekall: Specifying video events using compositions of spatiotemporal labels. arXiv preprint arXiv:1910.02993 (2019)

[10] Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples (2014)

[11] Guo, C., Rana, M., Cissé, M., van der Maaten, L.: Countering adversarial images using input transformations. CoRR **abs/1711.00117** (2017), http://arxiv.org/abs/1711.00117

[12] Hendrycks, D., Dietterich, T.: Benchmarking neural network robustness to common corruptions and perturbations (2019)

[13] Inkawhich, N., Inkawhich, M., Chen, Y., Li, H.: Adversarial attacks for optical flow-based action recognition classifiers (2018)

[14] Jiang, L., Ma, X., Chen, S., Bailey, J., Jiang, Y.G.: Black-box adversarial attacks on video recognition models. In: Proceedings of the 27th ACM International Conference on Multimedia. p. 864–872 (2019)

[15] Kataoka, H., Wakamiya, T., Hara, K., Satoh, Y.: Would mega-scale datasets further enhance spatiotemporal 3d cnns? (2020)

[16] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., Serre, T.: HMDB: a large video database for human motion recognition. In: Proceedings of the International Conference on Computer Vision (ICCV) (2011)

[17] Kurakin, A., Goodfellow, I., Bengio, S.: Adversarial examples in the physical world (2016)

[18] Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.: Towards deep learning models resistant to adversarial attacks (2017)

[19] Moosavi-Dezfooli, S.M., Fawzi, A., Fawzi, O., Frossard, P.: Universal adversarial perturbations (2016)

[20] Naeh, I., Pony, R., Mannor, S.: Flickering adversarial attacks against video recognition networks (2020)

[21] O'Gorman, T.J., Ross, J.M., Taber, A.H., Ziegler, J.F., Muhlfeld, H.P., Montrose, C.J., Curtis, H.W., Walsh, J.L.: Field testing for cosmic ray soft errors in semiconductor memories. IBM Journal of Research and Development **40**(1), 41–50 (1996)

[22] Raghunathan, A., Steinhardt, J., Liang, P.: Certified defenses against adversarial examples (2018)

[23] Raghunathan, A., Steinhardt, J., Liang, P.: Semidefinite relaxations for certifying robustness to adversarial examples (2018)

[24] Raghunathan, A., Xie, S.M., Yang, F., Duchi, J., Liang, P.: Understanding and mitigating the tradeoff between robustness and accuracy (2020)

[25] Seymour, R., Stewart, D., Ming, J.: Comparison of image transform-based features for visual speech recognition in clean and corrupted videos. EURASIP Journal on Image and Video Processing (2007). https://doi.org/doi:10.1155/2008/810362

[26] Shou, Z., Wang, D., Chang, S.F.: Temporal action localization in untrimmed videos via multi-stage cnns. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1049–1058 (2016)

[27] Sivathanu, G., Wright, C.P., Zadok, E.: Ensuring data integrity in storage: Techniques and applications. In: Proceedings of the 2005 ACM Workshop on Storage Security and Survivability. p. 26–36 (2005)

[28] Soomro, K., Zamir, A.R., Shah, M.: Ucf101: A dataset of 101 human actions classes from videos in the wild (2012)

[29] Srinivasan, V., Gul, S., Bosse, S., Meyer, J.T., Schierl, T., Hellge, C., Samek, W.: On the robustness of action recognition methods in compressed and pixel domain. In: 2016 6th European Workshop on Visual Information Processing (EUVIP). pp. 1–6 (2016)

[30] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks (2013)

[31] Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I., Boneh, D., McDaniel, P.: Ensemble adversarial training: Attacks and defenses (2017)

[32] Tsuzuku, Y., Sato, I.: On the structural sensitivity of deep convolutional networks to the directions of fourier basis functions (2018)

[33] Wei, X., Zhu, J., Su, H.: Sparse adversarial perturbations for videos (2018)

[34] Yin, D., Lopes, R.G., Shlens, J., Cubuk, E.D., Gilmer, J.: A fourier perspective on model robustness in computer vision (2019)

[35] Zhang, Y., Rajimwale, A., Arpaci-Dusseau, A.C., Arpaci-Dusseau, R.H.: End-to-end data integrity for file systems: A zfs case study. In: Proceedings of the 8th USENIX Conference on File and Storage Technologies. p. 3 (2010)

[36] Zheng, S., Song, Y., Leung, T., Goodfellow, I.: Improving the robustness of deep neural networks via stability training. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2016)