Textile & Leather Review

ISSN 2623-6281| www.tlr-journal.com | 10.31881/TLR

A Hard-label Black-box Adversarial Example Generation Algorithm on Video Models

Yulin Jing, Lijun Wu

How to cite: Jing Y, Wu L. A Hard-label Black-box Adversarial Example Generation Algorithm on Video Models. Textile & Leather Review. 2026; 9:1888-1900. https://doi.org/10.31881/TLR.2026.1888

How to link: https://doi.org/10.31881/TLR.2026.1888

Published: 25 April 2026

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License

A Hard-label Black-box Adversarial Example Generation Algorithm on Video Models

Yulin Jing*, Lijun Wu
School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610000, Sichuan, China
*jingyulin@std.uestc.edu.cn

Article
https://doi.org/10.31881/TLR.2026.1888
Published 25 April 2026

ABSTRACT

With the rapid development of deep learning, Deep Neural Networks (DNNs) have been widely applied in various fields, including intelligent visual inspection in textile industrial manufacturing. However, current DNNs still face the challenge of adversarial examples (AEs). According to the information that the researcher can obtain, AEs can be categorized into three types: white-box, score-based black-box, and hard-label black-box. Among them, hard-label black-box AEs are recognized as the most meaningful and practical. Currently, most researches on AEs target image models, there are relatively few researches on video models. To close this gap, we improve the original Monte Carlo algorithm and innovatively propose a hard-label black-box adversarial example generation algorithm for video models, called VDA. Extensive experiments show that, compared to the algorithm based on the original Monte Carlo, VDA can improve success rate by nearly 6 times under the same conditions. This research provides a new perspective for evaluating the security of video-based monitoring systems in the textile industry.

KEYWORDS

adversarial examples, hard-label, black-box, video models, textile intelligent monitoring

INTRODUCTION

In recent years, Deep Neural Networks (DNNs) have demonstrated outstanding performance in numerous video-related tasks, such as video recognition, video caption generation, and video segmentation [1,2]. In the modern textile industry, these video-based tasks are increasingly utilized for real-time production monitoring and worker safety analysis. However, relevant studies indicate that DNNs are vulnerable to adversarial examples (AEs). These AEs are typically generated by introducing perturbation into the original video that is imperceptible to humans but capable of altering the neural network's classification results [3].

This has raised concerns about the future applicability of DNNs in video-related fields, especially in industrial scenarios where reliability is paramount. Essentially, AEs expose potential security vulnerabilities of DNNs, which restricts the application prospects of DNNs in the security-critical fields. Therefore, an in-depth study of adversarial example generation algorithms can help researchers find the vulnerabilities of DNNs proactively so that they can take some measures (such as adversarial training) to enhance the security of DNNs, which is significantly important for improving the robustness of DNNs. Consequently, an increasing number of studies have focused on adversarial example generation algorithms [4,5].

To generate high-quality AEs, the process of adding perturbations to clean samples usually needs to test the DNNs-based model (also called the victim model or the target model) iteratively so that the perturbation sampling strategy can be adjusted dynamically according to the target model's output. To date, research on adversarial examples against image models has been relatively extensive. Based on the level of information available to researchers, adversarial example generation can be categorized into three types: white-box setting, score-based black-box setting, and hard-label black-box setting. In white-box setting, researchers possess complete information about the target model, including network architecture, model parameters, classification labels, and confidence scores. Under these conditions, the adversarial sample generation problem is typically transformed into an optimization problem. Solutions involve regularizing the misclassification loss function [6] or converting its dual problem into a constrained optimization problem. FGSM [7] is a first-order adversarial example generation algorithm that generates adversarial examples by maximizing the classification loss along the gradient direction. Similar to FGSM, PGD [8] is also an iterative adversarial example generation algorithm considered the strongest first-order adversarial example generation method, confining adversarial examples to a spherical space centered on the original sample with radius ε. The C&W adversarial example generation method transforms the adversarial example generation problem into an optimization problem and generates AEs by solving this problem under constrained conditions. In score-based black-box setting, researchers only obtain the classification labels and confidence scores of the targeted model, without access to its network architecture or parameters. Gradient estimation is performed based on confidence scores during adversarial sample generation [9]. Chen et al. [10] employed the FD algorithm for gradient estimation. To accelerate gradient estimation, Bhagoji et al. [9] optimized FD using dimension reduction techniques. To date, NES is considered the fastest gradient estimation algorithm, utilized by Ilyas et al. [11] to minimize model queries. In hard-label black-box setting, researchers only receive the target model's classification labels, HSJA [12] employs a sampling-based Monte Carlo algorithm for gradient estimation. Additional algorithms, such as QEBA [13] and NonLinear-BA [14], investigate the impact of low-dimensional space sampling on sampling efficiency.

Among these three types, hard-label black-box setting is considered the most practical yet challenging. In real-world applications, commercial models like Megvii Face++ and Microsoft Azure only return classification labels to users, preventing researchers from accessing their network architectures, model

parameters, or confidence scores. Currently, most adversarial example generation algorithms focus on image models [12]. Regarding video models, only a few studies exist on white-box setting and score-based black-box setting, and few hard-label black-box adversarial example generation algorithms for video models [15,16]. To fill this gap, we innovatively propose a hard-label black-box adversarial example generation algorithm, VDA. During algorithm design, we encountered several challenges:

(1) Compared to two-dimensional static images, videos typically possess four dimensions, resulting in a larger exploration space. This makes identifying effective perturbation during iterative process more challenging.

(2) While gradient estimation algorithms have been applied to video adversarial example generation, they are exclusively tailored for white-box setting or score-based black-box setting. Efficient gradient estimation for hard-label video black-box adversarial example generation remains a challenge, with no existing research addressing this.

This paper improves upon the Monte Carlo algorithm, innovatively proposing a hard-label video model black-box adversarial example generation algorithm, VDA. Experimental results demonstrate that under equivalent conditions, VDA achieves nearly tenfold improvement in adversarial example generation effectiveness compared to the original Monte Carlo algorithm.

VDA

We denote a video model by $G(x,\theta): R^{N\times H\times W\times C} \to R^K$, where x is the input video, θ are the parameters of model, and K is the number of classes of model G . Meanwhile, N、H、W、C denotes the frame number, frame height, frame width, and number of channels of the video, respectively. The goal of adversarial example generation is to introduce perturbation imperceptible to human vision onto the original video sample x, thereby generating an adversarial video sample x_{adv} that causes the model to misclassify it, i.e., $G(x_{adv},\theta) = y_{adv}$ .Here, $y_{adv} \neq y$. To ensure x and x_{adv} are visually indistinguishable, researchers typically confine x_{adv} to a spherical space centered at x with radius ε_{adv}, i.e., $\| x_{adv} - x \|_p \leq \varepsilon_{adv}$. For subsequent discussion, we next define the discriminator function D and the sign function φ:

$$D(x) = S(x)_{y_{adv}} - \max[S(x)_{y\neq y_{adv}}] \tag{1}$$

$$\varphi(x) = \begin{cases} 1 & D(x) \geq 0 \\ -1 & D(x) < 0 \end{cases} \tag{2}$$

Where $S(x)_y$ denotes the confidence score corresponding to the sample x being classified as category

y. In score-based black-box setting, researchers can simultaneously obtain the values of both the discriminator function D and the sign function φ. However, in hard-label black-box setting, researchers can only obtain the value of the sign function φ.

The overall workflow of the VDA algorithm is illustrated in Figure 1. Here, $x^{(t)}_{adv}$ denotes the adversarial video generated by the algorithm at the t iteration, $x^{(t)}_{advb}$ is the boundary adversarial video $x^{(t)}_{adv}$ projected onto the classification boundary, and x_{tgt} is the target video. In the initial phase, the researcher selects two videos , x_{src} and x_{tgt} , with corresponding labels y_{adv} and y . The objective is to progressively reduce the distance between x_{src} and x_{tgt} while preserving the classification label y_{adv} of x_{src} . When the distance diminishes sufficiently, x_{src} and x_{tgt} become visually indistinguishable. Specifically, the researcher repeats the following steps until the distance between $x^{(t)}_{adv}$ and x_{tgt} is below a specified threshold:

(1) Projection. Using a binary search algorithm, project $x^{(t)}_{adv}$ onto the classification boundaries of $x^{(t)}_{adv}$ and x_{tgt} , yielding the boundary adversarial video:

$$x^{(t)}_{advb} = \alpha \bullet x^{(t)}_{adv} + (1-\alpha) \bullet x_{tgt} \tag{3}$$

where α is the parameter for binary search.

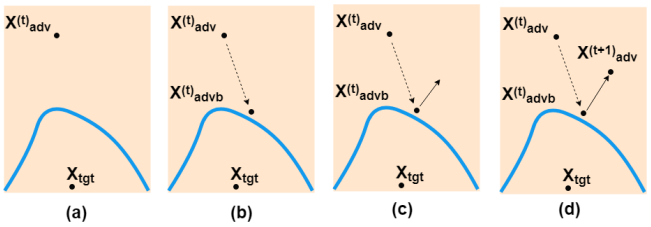


Figure 1. Framework of VDA

(2) Gradient estimation. Perform gradient estimation near $x^{(t)}_{advb}$ to compute the direction of movement of $x^{(t)}_{advb}$. In the field of image model adversarial example generation, Monte Carlo algorithms are predominantly used for gradient estimation:

$$g = \frac{1}{N}\sum_{k=1}^{N} \varphi(x^{(t)}_{advb} + \delta\mu) \bullet \mu \tag{4}$$

$$i_g = \frac{g}{\| g \|} \tag{5}$$

where δ is a fixed constant, and μ is the sampling perturbation.

In practice, we found that the original Monte Carlo algorithm is relatively inefficient. The primary reasons for this inefficiency lie in the following two aspects.

First, the algorithm only performs a qualitative analysis of perturbation based on the sign function φ, that is, assigning 1 to valid perturbation and -1 to invalid perturbation. This approach becomes inefficient when dealing with high-dimensional data.

Second, the original Monte Carlo algorithm updates samples using the unit gradient vector i_g. This approach fails to fully leverage the original gradient information. In practice, we observe that as the number of iterations increases, the gap between $x^{(t)}_{adv}$ and x_{tgt} narrows, and gradient estimation is also increasingly difficult. Consequently, $\| g \|$ becomes smaller. From this perspective, the magnitude of $\| g \|$ simultaneously indicates the stage of the iterations, yet the normalized i_g ignores this critical information.

To address these shortcomings, we improve the original Monte Carlo algorithm:

$$\rho(x) = \begin{cases} w & D(x) \geq 0 \\ -w & D(x) < 0 \end{cases} \tag{6}$$

$$g = \frac{1}{N} \sum_{k=1}^{N} \rho(x^{(t)}_{advb} + \delta\mu) \bullet \mu \tag{7}$$

Compared to the original algorithm, we first employed a weight function ρ to replace the original symbol function φ, enabling quantitative assessment of sampling perturbation μ through Equation (7). Furthermore, we omitted normalization of g after gradient estimation to preserve its inherent original information $\| g \|$.

(3) Update. Move $x^{(t)}_{advb}$ along the direction calculated in step 2) to obtain a new adversarial video $x^{(t+1)}_{adv}$.

If using the original Monte Carlo algorithm, the following equation should be used during the update:

$$x^{(t+1)}_{adv} = x^{(t)}_{advb} + \eta \bullet i_g \tag{8}$$

where η is the step size and i_g is the normalized gradient vector. However, this update method does not fully leverage the inherent magnitude variation information within the estimated gradient g. Therefore, we employ the following improved equation for sample updates:

$$x^{(t+1)}_{adv} = x^{(t)}_{advb} + \eta \bullet g \tag{9}$$

Compared to Equation (8), Equation (9) utilizes the original estimated gradient g instead of the normalized gradient vector i_g . As the number of iterations increases, the distance between $x^{(t)}{}_{adv}$ and x_{tgt} diminishes, and $\| g \|$ also decreases. Consequently, g undergoes adaptive adjustment based on the magnitude of $\| g \|$ at different stages of the iterations, effectively accommodating variations in the adversarial example generation process and enhancing the efficiency of updating adversarial examples.

As shown in Figure 1, the distance between $x^{(t+1)}{}_{adv}$ and x_{tgt} is smaller than that between $x^{(t)}{}_{adv}$ and x_{tgt} . Therefore, as the number of iterations increases, the new adversarial video gradually approaches x_{tgt} , while its classification label remains y_{adv} . This ultimately achieves the adversarial example generation effect. It is worth noting that if the algorithm is executed for the first time, wu should set x_{src} = $x^{(0)}{}_{adv}$.

EXPERIMENTS

Experimental Settings

We employed the human motion recognition dataset HMDB-51 [17] and the video recognition model C3D [18] as our evaluation dataset and model, respectively. HMDB-51 is a human motion recognition dataset and one of the most widely used datasets in the field of video recognition models. It comprises 7,000 videos across 51 categories, with 70% allocated to the training set and 30% to the test set. In our experiments, video dimensions were uniformly adjusted to $16 \times 3 \times 112 \times 112$ .The C3D model learns spatio-temporal features through 3D convolutional layers and is widely recognized as the standard benchmark for video recognition models. Drawing inspiration from the evaluation metrics used in HSJA [12] (S&P) and QEBA [13] (CVPR) for hard-label black-box adversarial example generation on image models, we employ the following metrics to assess VDA's performance:

(1) Mean Squared Error Distance (MSE): The mean squared error distance between the original target video and the adversarial video, indicating the magnitude of perturbation. A smaller MSE indicates greater similarity between the adversarial video and the target video, signifying better adversarial example generation performance.

(2) Success Rate (SRT): which is the ratio of successful adversarial videos to all test videos. The MSE and model queries of these successful adversarial videos are below specific thresholds respectively. The higher the SRT is, the more efficient the algorithm is. In our experiments, the maximum model queries are set to 400,000, while the maximum MSE could be set to different values depending on different experiments.

(3) Mean Query Numbers (MQN): which is the average model queries of all successful adversarial videos. The lower the MQN is, the better the algorithm is.

It should be noted that successful adversarial videos refer to those that cause the target model to

misclassify. Such videos may contain perturbations of varying magnitudes. We drew inspiration from the design of VBAD[16]—the first score-based video model black-box adversarial example generation algorithm (published at ACM MM)—when designing the comparative algorithm. Specifically, while preserving other components of VDA, we integrated the original Monte Carlo algorithm into the VDA architecture to form a variant called IMC-VDA. It should be noted that HSJA also employs the original Monte Carlo algorithm, so in our experiments, HSJA and IMC-VDA exhibit identical performance. Subsequent experiments will evaluate the performance differences between VDA and IMC-VDA.

Results and Analysis

Figure 2 illustrates the comparative vision effectiveness of VDA and IMC-VDA. The first and second rows display the original videos labeled as “punch” and “sword exercise,” respectively. In the experiment, the first row serves as the source video and the second row as the target video. The adversarial example generation aims to visually align the source video with the target video, while keeping the label of the source video unchanged. This process can also be interpreted as subtly incorporating features from the first row into the second row—without triggering perceptible visual differences to humans—thereby changing the second row's label (“sword exercise”) to match the first row's (“punch”). The third and fourth rows in the figure are generated adversarial videos, both classified as “punch.” The third row is an adversarial video generated by the VDA algorithm after 20,722 model queries, with an MSE distance of 31 from the original target video. The fourth row shows an adversarial video generated by the IMC-VDA algorithm after 401,796 model queries, with an MSE distance of 146 from the original target video. It is evident that the VDA generates high-quality adversarial videos imperceptible to the human eye with minimal queries. Its model queries are only 5% of IMC-VDA's, yet its effectiveness is nearly five times that of IMC-VDA. Compared to VDA, IMC-VDA requires nearly 20 times more model queries yet produces adversarial videos of poor quality, where the source video's traces are even visually discernible. It is evident that VDA significantly outperforms IMC-VDA in both performance and speed. Due to space constraints, the videos in the figure are sampled every 4 frames starting from the first frame.

Figure 2. Vision effect

Figure 3 illustrates the MSE variation with model queries increasing for both algorithms. As shown, VDA rapidly reduces MSE to 60 within 20,000 model queries and ultimately optimizes it to 18. In contrast, IMC-VDA only achieves an MSE of 191 within 20,000 model queries and ultimately optimizes it to 138. It is evident that VDA achieves an optimization effect nearly 7.6 times greater than IMC-VDA.

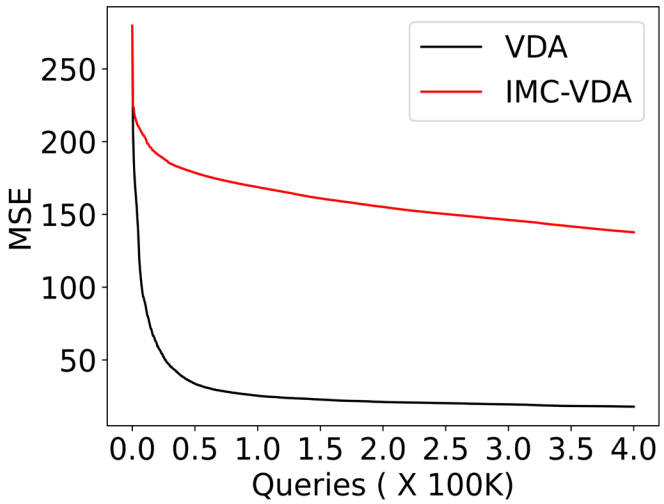


Figure 3. MSE

Figure 4 depicts the success rate when the MSE threshold is set to 25. As shown, VDA rapidly achieves a 68% success rate within 40,000 model accesses, while IMC-VDA remains at 0%. VDA ultimately reaches a 95% success rate, compared to IMC-VDA's 16%—a nearly 6-fold difference.

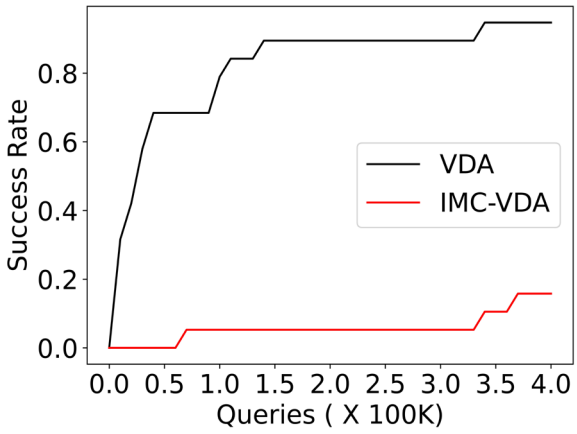
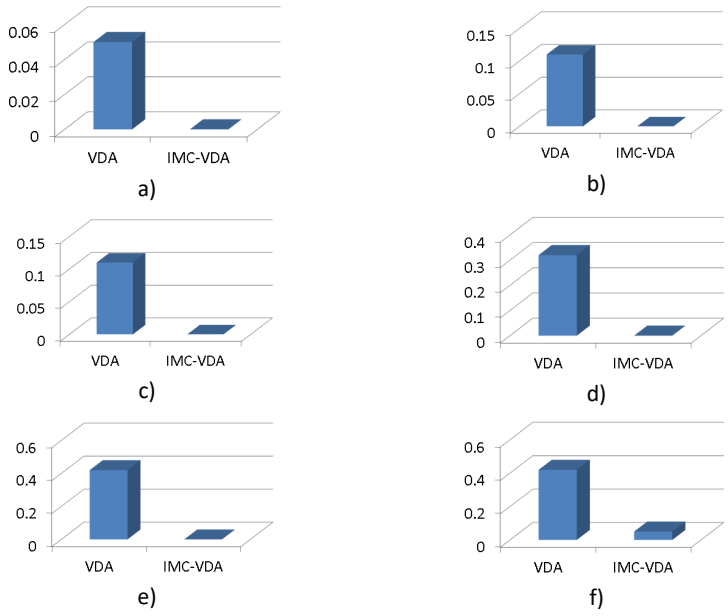


Figure 4. SRT when MSE=25

To further compare success rates under different MSE thresholds, we summarize the results in Figure 5. Figure 5 reveals that only VDA succeeds at MSE 10. Similarly, only VDA succeeds at MSE 15 with 250,000 model accesses. Although IMC-VDA succeeds at MSE 15 with 350,000 model accesses, its success rate is only 5%, far below VDA's 42%. When MSE is 20, both VDA and IMC-VDA succeeded, but VDA's success rate was at least 15 times that of IMC-VDA. At MSE 25, VDA's success rate was 7 to 18 times that of IMC-VDA.



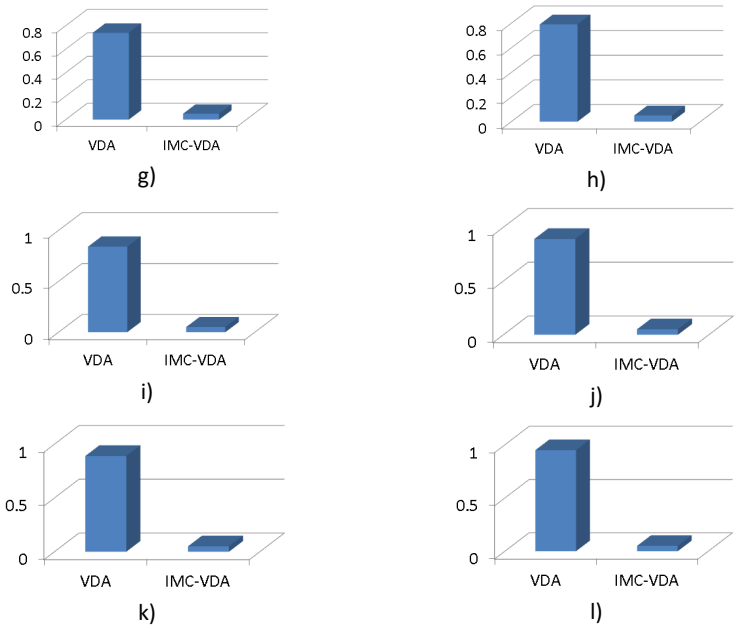


Figure 5. SRT on different MSE=25: (a)MSE=10,Queries=150K; (b)MSE=10,Queries=250K; (c)MSE=10,Queries=350K; (d)MSE=15,Queries=150K; (e)MSE=15,Queries=250K; (f)MSE=15,Queries=350K; (g)MSE=20,Queries=150K; (h)MSE=20,Queries=250K; (i)MSE=20,Queries=350K; (j)MSE=25,Queries=150K; (k)MSE=25,Queries=250K; (l)MSE=25,Queries=350K

CONCLUSION

This paper presents an innovative hard-label black-box adversarial example generation algorithm VDA for video models. This fills a research gap in hard-label black-box adversarial example generation targeting video models. Experimental results demonstrate that under equivalent conditions, VDA achieves nearly 6-fold success rate improvement in adversarial example generation effectiveness compared to the algorithm based on original Monte Carlo. The high efficiency of the proposed VDA algorithm highlights the potential security risks of current video recognition models, serving as a critical reference for developing more robust visual analysis systems in complex industrial environments, such as automated textile manufacturing.

Author Contributions

Yulin Jing designed, collected and analyzed the data, and drafted the manuscript. Yulin Jing conducted the study, critically revised the manuscript for important intellectual content, and gave final approval of the version to be published. Lijun Wu participated fully in the work, take public responsibility for appropriate portions of the content, and agreed to be accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

Conflicts of Interest

Declare conflicts of interest or state, “The authors declare no conflict of interest.” Authors must identify and declare any personal circumstances or interests that may be perceived as inappropriately influencing the representation or interpretation of reported research results.

Funding

This research received no external funding.

REFERENCES

[1] Xiao X, Zhang J, Shao Y, Liu J, Shi K, He C, et al. Deep learning-based medical ultrasound image and video segmentation methods: overview, frontiers, and challenges. Sensors. 2025; 25:2361. doi: 10.3390/s25082361

[2] Ding H, Liu C, He S, Ying K, Jiang X, Loy C, et al. MeViS: A multi-modal dataset for referring motion expression video segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2025; 47(12):11400–11416. doi: 10.1109/TPAMI.2025.3600507

[3] Song J, Yu D, Teng H, Chen Y. RLVS: A reinforcement learning-based sparse adversarial attack method for black-box video recognition. Electronics. 2025; 14:245. doi: 10.3390/electronics14020245

[4] Han X, Zhang S, Wang H, Tian Q. DSAA: Cross-modal transferable double sparse adversarial attacks from images to videos. Neurocomputing. 2025; 639:130212.

[5] Yao X, Li E, Chen Y, Guo J, Huang K, Tang F, et al. Stealthy and efficient adversarial example attack on video retrieval systems. Neural Networks. 2025; 191:107829. doi: 10.1016/j.neunet.2025.107829

[6] Carlini N, Wagner D. Towards evaluating the robustness of neural networks. In: Proceedings of the IEEE Symposium on Security and Privacy (SP 2017); 22–24 May 2017; San Jose, CA, USA. Piscataway, USA: IEEE; 2017. p.39–57. doi: 10.1109/SP.2017.49

[7] Goodfellow IJ, Shlens J, Szegedy C. Explaining and harnessing adversarial examples. CoRR. 2014; abs/1412.6572.

[8] Madry A, Makelov A, Schmidt L, Tsipras D, Vladu A. Towards deep learning models resistant to adversarial attacks. In: International Conference on Learning Representations (ICLR 2018); 2018. Available from: https://openreview.net/forum?id=rJzIBfZAb

[9] Bhagoji AN, He W, Li B, Song D. Practical black-box attacks on deep neural networks using efficient query mechanisms. In: Ferrari V, Hebert M, Sminchisescu C, Weiss Y, editors. Computer Vision – ECCV 2018; 8–14 Sep 2018; Munich, Germany. Cham, Switzerland: Springer International Publishing; 2018. p. 158–174.

[10] Chen PY, Zhang H, Sharma Y, Yi J, Hsieh CJ. ZOO: zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In: Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security (AISec '17); 4–8 Nov 2017; Dallas, TX, USA. New York, USA: Association for Computing Machinery; 2017. p. 15–26. doi: 10.1145/3128572.3140448

[11] Ilyas A, Engstrom L, Athalye A, Lin J. Black-box adversarial attacks with limited queries and information. In: Proceedings of the 35th International Conference on Machine Learning (ICML 2018); 10–15 Jul 2018; Stockholm, Sweden. New York, USA: PMLR; 2018.

[12] Chen J, Jordan MI. HopSkipJumpAttack: a query-efficient hard-label attack. In: Proceedings of the IEEE Symposium on Security and Privacy (SP 2020); 18–21 May 2020; San Francisco, CA, USA. Piscataway, USA: IEEE; 2020. p.1277–1294.

[13] Li H, Xu X, Zhang X, Yang S, Li B. QEBA: query-efficient boundary-based blackbox attack. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2020); 14–19 Jun 2020; Seattle, WA, USA. Piscataway, USA: IEEE; 2020. p.1218–1227. doi: 10.1109/CVPR42600.2020.00130

[14] Li H, Li L, Xu X, Zhang X, Yang S, Li B. Nonlinear projection based gradient estimation for query efficient blackbox attacks. In: Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS 2021); 13–15 Apr 2021; Virtual Event, USA. Proceedings of Machine Learning Research 130:3142–3150; 2021. Available from: https://proceedings.mlr.press/v130/li21f.html

[15] Zając M, Żołna K, Rostamzadeh N, Pinheiro PO. Adversarial framing for image and video classification. In: Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019); 27 Jan–1 Feb 2019; Honolulu, HI, USA. Palo Alto, USA: AAAI Press; 2019. p.10077–10078. doi: 10.1609/aaai.v33i01.330110077

[16]Jiang L, Ma X, Chen S, Bailey J, Jiang YG. Black-box adversarial attacks on video recognition models. In: Proceedings of the 27th ACM International Conference on Multimedia (MM ’19); 21–25 Oct 2019; Nice, France. New York, USA: Association for Computing Machinery; 2019. p.864–872. doi: 10.1145/3343031.3351088

[17]Kuehne H, Jhuang H, Garrote E, Poggio T, Serre T. HMDB: a large video database for human motion recognition. In: Proceedings of the 2011 International Conference on Computer Vision (ICCV 2011); 6–13 Nov 2011; Barcelona, Spain. Piscataway, USA: IEEE; 2011. p.2556–2563. doi: 10.1109/ICCV.2011.6126543

[18]Hara K, Kataoka H, Satoh Y. Can spatiotemporal 3D CNNs retrace the history of 2D CNNs and ImageNet? In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2018); 18–22 Jun 2018; Salt Lake City, UT, USA. Piscataway, USA: IEEE; 2018. p. 6546–6555. doi: 10.1109/CVPR.2018.00685