Response to Reviewers of TCSVT-07024-2021: A Simple and Strong Baseline for Universal Targeted Attacks on Siamese Visual Tracking

Zhenbang Li, Yaya Shi, Jin Gao, Shaoru Wang,
Bing Li, Pengpeng Liang, Weiming Hu

Dear Editors:

We would like to express our heartfelt gratitude to you and the reviewers for the insightful and helpful comments. When we revised the paper, we carefully considered and followed all the comments and suggestions provided by you and the reviewers. To summarize, we have made the following revisions:

(1) We have added the advantages & limitations of the proposed method after experimental analysis and the future work to the conclusion.

(2) We have carefully considered and followed all the comments and suggestions related to the clarity of the writing, and made the new material a thoroughly revised manuscript.

(3) We have added a new experiment to attack in the YCbCr color space to reduce the perceptibility of the perturbations.

(4) We have added a new experiment to evaluate the attack performance of the random patterns.

(5) We have added a new experiment to evaluate the attack performance when only adding the perturbation to the search/template image.

(6) We have added 3 papers published in the IEEE Transactions on Circuits and Systems for Video Technology, which are most closely related to our manuscript, and analysed what is distinctive / new about our current manuscript related to these previously published papers. For your convenience in cross-checking, the new text is given as follows.

Some recent research topics which are mostly related to adversarial attack also include digital watermarking [1], 3D face presentation attacks to the face recognition networks [2] and adversarial defense for image classifiers [3].

We hope that our revised manuscript is now appropriate for publication in IEEE Transactions on Circuits and Systems for Video Technology. Specific responses to all the comments of each reviewer are included in the rest of this document and highlighted using bold font after the comments of each reviewer for the convenience of cross-reference. To make the changes easier to identify where necessary, we also have underlined most of the revised parts in the manuscript and provide an underlined version for the convenience of second review.

We are looking forward to your reply.

Yours sincerely,

Zhenbang Li, Yaya Shi, Jin Gao, Shaoru Wang, Bing Li, Pengpeng Liang, Weiming Hu

Dr. Jin Gao (Contact author)

National Laboratory of Pattern Recognition (NLPR)

Institute of Automation, Chinese Academy of Sciences (CASIA)

Address: No. 95, Zhongguancun East Road, Haidian District,

Beijing 100190, P. R. China

Email: jin.gao@nlpr.ia.ac.cn

Response Letter to Reviewer #1

Dear Reviewer #1:

Thank you very much for your thorough review. Your insightful comments are very helpful for us to improve the quality of the paper. According to your comments and suggestions, we have carefully and extensively revised the manuscript. The main revised parts are highlighted by underlines in the underlined version for your convenience. You will find that all your comments and suggestions are considered and followed. We hope that our revised manuscript is now appropriate for publication in IEEE Transactions on Circuits and Systems for Video Technology. In addition, point-to-point responses to your comments are given below and highlighted using bold font in line with your comments in order to facilitate cross-referencing.

We are looking forward to your reply.

Yours sincerely,

Zhenbang Li, Yaya Shi, Jin Gao, Shaoru Wang, Bing Li, Pengpeng Liang, Weiming Hu

Dr. Jin Gao (Contact author)

National Laboratory of Pattern Recognition (NLPR)

Institute of Automation, Chinese Academy of Sciences (CASIA)

Address: No. 95, Zhongguancun East Road, Haidian District,

Beijing 100190, P. R. China

Email: jin.gao@nlpr.ia.ac.cn

This paper addresses the task of attacking Siamese network-based trackers in a simple yet effective fashion. Unlike other methods that operate in the video-specific attacking regime (which resides on network inference for generating perturbations while tracking), this method is the first to perform universal targeted attacks for Siamese trackers utilizing both the translucent perturbation and the adversarial patch together. By adding the perturbation to the template and adding the patch to the search image while performing tracking, this work fools the Siamese trackers to the fake target region and thus makes them fail in tracking the real target object. Overall, this is an interesting paper, and it is well written and organized. As it is a resubmitted manuscript, I notice that the authors have made substantial changes to the previous manuscript, which are able to appropriately respond to the comments made by the previous reviewers. Although the template perturbation and adversarial patch are both easy to observe for human eyes as the previous reviewers have pointed out, and the SSIMs for them are also lower than the video-specific attacking method, e.g., FAN [4], this reviewer believes that this proposed new framework can be a new configuration of adversarial attack on visual tracking for its achieved balance between the attack efficiency and the perturbation perceptibility. This new configuration will attract increasing attention from the visual tracking attack community to study on more efficient attack methods.

Many thanks for your positive comments on the strength of our paper and the novelty of the proposed attack method.

In addition, I suggest the authors add more experiments to demonstrate the practicability of the attack method when the ground truth box information is missing in the training data. The experimental results show that it is effective to use the predicted boxes instead of ground truth boxes for training perturbations.

Thanks for the good comment. As suggessted, we have added more experiments to demonstrate the practicability of the attack method when the ground truth box in-

Table VII: Attack results on OTB-15, GOT-Val and LaSOT with or without the ground truth information. w/ GT represents that we train perturbations using the groud truth bounding box information of the trainig dataset. w/o GT represents that we train perturbations using the predicted boxes instead of ground truth boxes of the training dataset.

Benchmarks	Metrics	Unargeted Attack		Targeted Attack	
		w/ GT	w/o GT	w/ GT	w/o GT
OTB-15	AO	0.063	0.056	0.759	0.752
	Precision	0.092	0.080	0.795	0.794
GOT-Val	SR	0.123	0.121	0.890	0.893
	AO	0.153	0.160	0.840	0.833
LaSOT	Precision	0.046	0.043	0.605	0.531
	Norm. Prec.	0.048	0.044	0.702	0.660
	AO	0.069	0.063	0.691	0.646

formation is missing in the training data. Specifically, we evaluate the untargeted and targeted attack performance on the following datasets and metrics: (1) AO and precision on OTB-2015, (2) SR and AO on GOT-Val, and (3) precision, normalized precision and AO on LaSOT. The detailed analysis of new added experiments is added in Section IV.D of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

We evaluate the untargeted and targeted attack performance on the following datasets and metrics: (1) AO and precision on OTB-2015, (2) SR and AO on GOT-Val, and (3) precision, normalized precision and AO on LaSOT. Experimental results are shown in Table VII. In terms of AO on GOT-Val, the untargeted attack performance is 0.153 when the ground truth is used to train perturbations and 0.160 when the ground truth is not used. In terms of the precision score on OTB-2015, the targeted attack performance is 0.795 when the ground truth is used to train perturbations and 0.794 when the ground truth is not used. In terms of the normalized precision score on LaSOT, the targeted attack

5

Algorithm 2 Attack Process

Input: The trained perturbations δ and p, Siamese tracker f, video $V = \{I_i\}_1^T$. b_1^{gt} is the position of the real target in the first frame. $B^{fake} = \{b_i^{fake}\}_1^T$ is the trajectory we hope the tracker to output.

Output: $B^{pred} = \{b_i^{pred}\}_1^T$

  1: Generate the clean template image $\mathbf{z}_1$ according to I_1 and b_1^{gt}.
  2: Generate the perturbed template image $\tilde{\mathbf{z}}_1 = \mathbf{z}_1 + \delta$.
  3: Let $i = 2$.
  4: **while** $i \leq T$ **do**
  5:     Generate clean search image $\mathbf{x}_i$ according to I_i and b_{i-1}^{pred}.
  6:     $b_i^{fake} = \{x_{0_i}, y_{0_i}, x_{1_i}, y_{1_i}\}$
  7:     Generate the perturbed search image $\tilde{\mathbf{x}}_i = A_{\text{add}}(\mathbf{x}_i, p, \{x_{0_i}, y_{0_i}, x_{1_i}, y_{1_i}\})$.
  8:     $\mathbf{C}, \mathbf{R}, \mathbf{Q} = f(\tilde{\mathbf{x}}_i, \tilde{\mathbf{z}}_1)$.
  9:     Generate the predicted bounding box b_i^{pred} according to $\mathbf{C}, \mathbf{R}, \mathbf{Q}$.
10:     $i = i + 1$.
11: **end while**
12: **return** B^{pred}
---

performance is 0.702 when the ground truth is used to train perturbations and 0.660 when the ground truth is not used. Therefore, it is effective to use the predicted boxes instead of ground truth boxes for training perturbations.

A small question is that it will be better if the authors can provide some pseudo code for the untargeted attack and targeted attack processes in addition to the training process. This will facilitate the understanding of the attacking process while performing tracking.

Thanks for the good comment. As suggested, we have provided the algorithm for the attack process in Section III.C of the revised manuscript. Both the targeted attack and untargeted attack process follow the same steps as shown in Alg. 2.

In addition, the font size in Fig. 9 is too small to read on my computer, which needs to be improved.

Sorry for the inconvenience of reading Fig. 9 caused by the small font size. We are willing to adjust Fig. 9 font size to make it easier to read on the computer, but because

we have added many ablation experiments to revised manuscript, the total number of pages exceeds the maximum page limit for TCSVT (i.e., 14 pages). Therefore we have to remove the analysis of the attack performance for each attribute of the OTB as well as Fig. 9.

Also, some recent papers are valued to be referred (2020-2021), to enhance the quality.

Thanks for the good comments. We have added several 2020-2021 related papers to enhance the quality of the revised manuscript. For your convenience of cross review, the new text is given as follows.

Some recent research topics which are mostly related to adversarial attack also include digital watermarking [1], 3D face presentation attacks to the face recognition networks [2] and adversarial defense for image classifiers [3].

Response Letter to Reviewer #2

Dear Reviewer #2:

Thank you very much for your thorough review. Your insightful comments are very helpful for us to improve the quality of the paper. According to your comments and suggestions, we have carefully and extensively revised the manuscript. The main revised parts are highlighted by underlines in the underlined version for your convenience. You will find that all your comments and suggestions are considered and followed. We hope that our revised manuscript is now appropriate for publication in IEEE Transactions on Circuits and Systems for Video Technology. In addition, point-to-point responses to your comments are given below and highlighted using bold font in line with your comments in order to facilitate cross-referencing.

We are looking forward to your reply.

Yours sincerely,

Zhenbang Li, Yaya Shi, Jin Gao, Shaoru Wang, Bing Li, Pengpeng Liang, Weiming Hu

Dr. Jin Gao (Contact author)

National Laboratory of Pattern Recognition (NLPR)

Institute of Automation, Chinese Academy of Sciences (CASIA)

Address: No. 95, Zhongguancun East Road, Haidian District,

Beijing 100190, P. R. China

Email: jin.gao@nlpr.ia.ac.cn

In this paper, the authors train a universal adversarial patch to add on both template and search regions of a Siamese based tracker to deteriorate its original performance. The proposed perturbations are video-agnostic, leading to a low computational cost during attack. The experiment validations show that the proposed method achieves favorable attack results on OTB2015, GOT-10k, LaSOT, UAV123, VOT2016, VOT2018 and VOT2019. In addition, the generated perturbations transfer well on other Siamese trackers as well. The idea of this paper is interesting and the experiments are thorough.

Many thanks for your positive comments on the strength of our paper and the novelty of the proposed attack method.

However, there are some concerns over the implementation, performance and writing. 1. The authors state that training with Ep. 4 leads to an obvious patch on the images while using Eq.5 into the training process results in a less obvious patch. The reviewer considers that giving a constraint (e.g. l_∞) on the p_x in Eq.4 can make the perturbation imperceptible intuitively. Please give more analysis on this setting. Besides, the reviewer hopes to know the reason why give an extra perturbation on the template region. The perturbations on template and search regions look similar, while the authors say that they are different. Please state the difference between the patch application operator on search examples and the operator on template examples. In addition, the denotations of A_{paste} in Eq.4 and A_{add} in Eq.5 seem like the same one.

Thanks for the good comment. Before analysis if giving constraint (e.g. L_∞) on the p_x in Eq.4 is really lead to imperceptible or not, let us first clarify the concept of the $A_{\mathbf{paste}}$ and $A_{\mathbf{add}}$, which was not explained well in the previous manuscript.

$A_{\mathbf{paste}}$ means that, in the region where the perturbation is pasted, the pixel values of the original image are *replaced* with the pixel values of the perturbation. Thus, even if we add constraints to the patch so that the perturbation values are small, when the patch is pasted to the image, it will look like an almost black region, which

is obvious rather than imperceptible. The operator $A_{\mathbf{add}}$ refers to the injection of our translucent adversarial patch into the search image, i.e., the values of our translucent adversarial patch are *added* to the pixel values of the original image in the $b_{\mathbf{x}}^{fake}$ area.

Next, let us explain the difference between the patch p added to the search image and the perturbation δ added to the template image. p has a smaller size of 64×64, and added to a larger (303×303) search image as a fake target. δ is the same size (127×127) as the template image. The adversarial patch is usually added to the background region of the search image and does not change the pixel values in the foreground region where the real target exists. Therefore if we do not add additional perturbation to the template image, the response values of the region where the real target exists on the heatmaps are always high. Thus it is necessary to perturb the template image to cooling down hot regions where the real target exists on the heatmaps and increasing the responses at the position of the *fake target*. For your convenience of cross checking, the new added text is shown as follows:

A_{paste} means that, in the region where the perturbation is pasted, the pixel values of the original image are *replaced* with the pixel values of the perturbation. The operator A_{add} refers to the injection of our translucent adversarial patch into the search image, i.e., the values of our translucent adversarial patch are *added* to the pixel values of the original image in the $b_{\mathbf{x}}^{fake}$ area.

The adversarial patch is usually added to the background region of the search image and does not change the pixel values in the foreground region where the real target exists. Therefore if we do not add additional perturbation to the template image, the response values of the region where the real target exists on the heatmaps are always high. Thus it is necessary to perturb the template image to cooling down hot regions where the real target exists on the heatmaps and increasing the responses at the position of the *fake target*.

2. I agree with reviewer 3, the ground truth boxes are inaccessible to trackers during

the inference. It seems that the authors use the ground truth boxes to generate the fake trajectory in lines 51-57 on page 7. Please clarify it.

Thanks for the good comment. As suggested, we consider an alternative way to generate the *fake trajectory* without the use of the ground truth boxes. Specifically, the attacker forces the tracker to follow a fixed direction in a video. For each video, we assign a random direction from 4 different directions, consisting of shifting the box by $(\pm 3, \pm 3)$ pixels in each consecutive frames, corresponding to the four directions $45°, -45°, 135°, -135°$. The attack performance is evaluated according to AO/SR on GOT-Val. The detailed analysis of new added experiment is added in Section IV.D of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

Ablation Study: An alternative way to generate the fake trajectory Here, we consider an alternative way to generate the *fake trajectory*. Specifically, the attacker forces the tracker to follow a fixed direction in a video. For each video, we assign a random direction from 4 different directions, consisting of shifting the box by $(\pm 3, \pm 3)$ pixels in each consecutive frames, corresponding to the four directions $45°, -45°, 135°, -135°$. The attack performance is evaluated according to AO/SR on GOT-Val. As shown in Table VIII, our attack algorithm achieves effective attacks under two different fake trajectories. Specifically, when the fake trajectory follows a fixed direction, the AO of the untargeted attack is 0.175, and the AO of the targeted attack is 0.845.

3. For the experiments, the authors should conduct the ablation study on only adding perturbations on the template images or the search regions to show the impact of p and δ.

Thanks for your advice and we have conducted the ablation study on only adding perturbations on the template images or the search regions to show the impact of p and δ. The detailed analysis of new added experiment is added in Section IV.D of the revised manuscript. For your convenience in cross-checking, the new text is given as

11

Table VIII: Influence of different ways to generate the fake trajectory. The first row represents that the attacker forces the tracker to follow a fixed direction in a video. The second row represents that the fake trajectory follows the real trajectory. The attack performance is evaluated according to AO/SR on GOT-Val.

Type of the Fake Trajectory	Untargeted Attack		Targeted Attack	
	AO	SR	AO	SR
Fixed direction	0.175	0.144	0.845	0.897
Follow the real trajectory	0.153	0.123	0.840	0.890

follows.

Ablation Study: Only perturb the template or search image To analyze the impact of p and δ, we evaluate the attack performance when only adding perturbations on the template images or the search regions. The attack performance is evaluated according to AO/SR on GOT-Val. The result is shown in Table IX. For the untargeted attack, only perturbing the template/search image leads to the AO of 0.510/0.714, while perturbing both the template and the search image leads to the AO of 0.153. For the targeted attack, only perturbing the template/search image leads to the AO of 0.156/0.160, while perturbing both the template and the search image leads to the AO of 0.840. Experimental results show that perturbing both the template and the search image can achieve better attack performance than only perturbing the template/search image.

4. As reviewer 1 and reviewer 2 say, the perturbations added to template regions and research regions are not imperceptible, which may be helpful to misguide the tracker. The reviewer considers that adding a similar random pattern on the template and search regions to further illustrate the effectiveness of the proposed method.

To illustrate the effectiveness of the proposed method, we add random patterns on the template and search regions. Specifically, we design two kinds of random patterns: (1) the random pattern similar to the trained perturbations, and (2) the

12

Table IX: Attack performance when only adding the perturbation to the search/template image. The first row refers to only perturbing the template image. The second row refers to only perturb the search image. The third row refers to perturb both the template and the search image. The attack performance is evaluated according to AO/SR on GOT-Val.

Template	Search	Untargeted Attack		Targeted Attack	
		AO	SR	AO	SR
✓		0.510	0.567	0.156	0.106
	✓	0.714	0.841	0.160	0.132
✓	✓	0.153	0.123	0.840	0.890

Table X: Attack performance using random patterns. The attack performance is evaluated according to AO/SR on GOT-Val.

Perturbations used to perfrom attack	Untargeted Attack		Targeted Attack	
	AO	SR	AO	SR
Trained Perturbations	0.153	0.123	0.840	0.890
Similar Pattern	0.736	0.871	0.153	0.118
Gaussian Noise	0.740	0.875	0.144	0.101

random pattern generated using zero-mean Gaussian noise with standard deviation 50.0. We replace the trained perturbations with the random random pattern to attack SiamFC++_GoogleNet. The experiment is performed on GOT-Val. As shown in Table X, random patterns cannot effectively attack the tracker. For your convenience in cross-checking, the new text is given as follows.

Ablation Study: Attack performance of random patterns To illustrate the effectiveness of the proposed method, we add random patterns on the template and search regions (see Figure 8). Specifically, we design two kinds of random patterns: (1) the random pattern similar to the trained perturbations, and (2) the random pattern generated using zero-mean Gaussian noise with standard deviation 50.0. We replace the trained perturbations with the random random pattern to attack SiamFC++_GoogleNet. The experiment is performed on
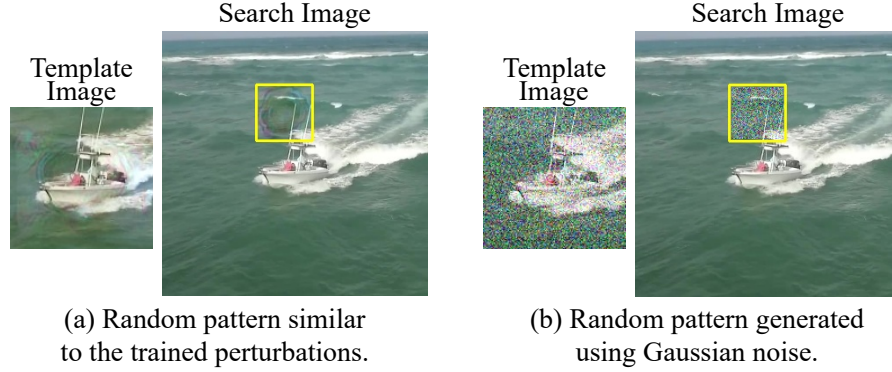
(a) Random pattern similar to the trained perturbations.

(b) Random pattern generated using Gaussian noise.

Figure 8: Visualization of random patterns.

GOT-Val. As shown in Table X, random patterns cannot effectively attack the tracker.

5. There are some minor problems, grammar errors and typos in this paper. The reviewer hopes the authors polish this paper again.

- On page 2, '1016' → '2016' in line 56. There is a same one in line 47 on page 6.

- The denotation of B_x^{fake} in Eq.4 is not clear enough, even though it can be inferred by the later part.

- On page 4, 'imperceptible' → 'imperceptibly' in line 57.

- The reinitialization of VOT-toolkit should be mentioned in the part of 'experimental setup'.

- On page 7, '.(see Table I)' is a typo.

Thanks for the good comment. We have carefully checked through the whole text and corrected the grammar mistakes and typos. Specifically, We have replaced "Experiment results on OTB2015 [5], GOT-10k [6], LaSOT [6], UAV123 [7], VOT1016 [8], VOT2018 [9] and VOT2019 [10]. benchmarks demonstrate the effectiveness and efficiency of our approach." with "Experiment results on OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT2016 [8], VOT2018 [9] and VOT2019 [10] benchmarks demonstrate the effectiveness and efficiency of our approach." **in Section I of the revised**

manuscript.

We have added the definition of B_x^{fake} of Eq.4 as follows: "$b^{fake} = \{x_0, y_0, x_1, y_1\}$ denotes the coordinates of the upper-left and lower-right corners of the fake target on the search image." **in Section III.A of the revised manuscript.**

We have replaced "In SiamFC++, the tracker first transforms the paired reference frame I_1 and annotation b_1^{gt} to get an template image $\mathbf{z}_1$, and transforms the search frame I_i to get the search image $\mathbf{x}_i$ centered at the position estimated in the previous frame." with "In SiamFC++, the tracker first transforms the paired reference frame I_1 and annotation b_1^{gt} to get a template image $\mathbf{z}_1$, and transforms the search frame I_i to get the search image $\mathbf{x}_i$ centered at the position estimated in the previous frame." in Section III.A of the revised manuscript.**

We have replaced "However, CNN attacks are usually expected imperceptible but the above method has to paste an obviously noticeable fake target patch to tracking frames, which raises the risk of being suspected." with "However, CNN attacks are usually expected to be imperceptible but the above method has to paste an obviously noticeable fake target patch to tracking frames, which raises the risk of being suspected." in Section III.A of the revised manuscript.**

We have replaced "A_add **adds the patch into the search image x at location** $\left(\frac{x_0+x_1}{2}, \frac{y_0+y_1}{2}\right)$." with "$A_{add}$ adds the patch into the search image $\mathbf{x}$ at location $\left(\frac{x_0+x_1}{2}, \frac{y_0+y_1}{2}\right)$." in Section III.B of the revised manuscript.**

We have replaced "We evaluate our video-agnostic perturbations for targeted attacks on several tracking benchmarks, i.e., OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT1016 [8], VOT2018 [9] and VOT2019 [10]." with "We evaluate our video-agnostic perturbations for targeted attacks on several tracking benchmarks, i.e., OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT2016 [8], VOT2018 [9] and VOT2019 [10]." in Section IV.A of the revised manuscript.**

We have replaced "**We adopt COCO [12], ILSVRC-VID [13] and the training splits of GOT-10k [6] and LaSOT [11] as our training set. (see Table I)**" with "We adopt COCO [12], ILSVRC-VID [13] and the training splits of GOT-10k [6] and LaSOT [11] as our training set." **in Section IV.B of the revised manuscript.**

We have mentioned the reinitialization of of VOT-toolkit in Section IV.A of the revised manuscript: Different from other datasets, the VOT dataset has a reinitialization module. When the tracker loses the target (i.e., the overlap is zero between the predicted result and the annotation), the tracker will be reinitialized with the ground truth.

Response Letter to Reviewer #3

Dear Reviewer #3:

Thank you very much for your thorough review. Your insightful comments are very helpful for us to improve the quality of the paper. According to your comments and suggestions, we have carefully and extensively revised the manuscript. The main revised parts are highlighted by underlines in the underlined version for your convenience. You will find that all your comments and suggestions are considered and followed. We hope that our revised manuscript is now appropriate for publication in IEEE Transactions on Circuits and Systems for Video Technology. In addition, point-to-point responses to your comments are given below and highlighted using bold font in line with your comments in order to facilitate cross-referencing.

We are looking forward to your reply.

Yours sincerely,

Zhenbang Li, Yaya Shi, Jin Gao, Shaoru Wang, Bing Li, Pengpeng Liang, Weiming Hu

Dr. Jin Gao (Contact author)

National Laboratory of Pattern Recognition (NLPR)

Institute of Automation, Chinese Academy of Sciences (CASIA)

Address: No. 95, Zhongguancun East Road, Haidian District,

Beijing 100190, P. R. China

Email: jin.gao@nlpr.ia.ac.cn

This paper employs the universal perturbation attacks on Siamese visual trackers. There are still the following concerns about the proposed method. 1. As stated in the paper, the proposed method "does not require gradient optimization or network inference". However, this is a double-edged sword since it resulted in suspicious attacks. Prior works commonly train a network to prevent not only suspicious attacks but also modifying every pixel. I think it's a major problem with this work. I suggest considering a proper strategy to remove/reduce it.

Thanks for your advice. The shortcoming of our method is the more conspicuous perturbation values compared to existing target tracking attack algorithms. Other methods add perturbations over the entire search image, whereas we need to add perturbations over a very small region of 64×64, which increases the learning difficulty and makes the perturbation values large. To reduce perceptibility while achieving the targeted attack, we consider a new way to perturb the search image: first we convert the search image into YCbCr color space, then add perturbations over the entire search image in the Y channel, and add perturbations over a very small region of 64×64 in the CbCr channel. Different from RGB color space, YCbCr color space encodes a color image similar to human eyes' retina, which separates the RGB components into a luminance component (Y) and two chrominance components (Cb as blue projection and Cr as red projection). We choose YCbCr color space since its color channels are less correlated and quite independent than RGB. While considering the color sensitivity of human vision system, the CbCr channels are less sensitive than the Y channel. This means that the patch added in the CbCr channels has better performance of transparency. The detailed analysis of new added experiment is added in Section IV.D of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

Ablation Study: Attack in YCbCr Space The shortcoming of our method is the more

Figure 9: Visualization of the perturbations. Figure (a) demonstrates the perturbations trained by the RGB attack method, i.e., the attack method proposed in Section III. Figure (b) demonstrates the perturbations trained by the YCbCr attack method: first we convert the search image into YCbCr color space, then add perturbations over the entire search image in the Y channel, and add perturbations over a very small region of 64×64 in the CbCr channel. For the template image, we convert it into YCbCr color space, then add the perturbation over the entire template image in all YCbCr channels. The perturbed template and search images are then converted into RGB color space and fed into the tracking network. The other steps of the training process are the same as the attack method in Section III. Attacking the YCbCr color space brings better imperceptibility. Compared with attacking the RGB space, the SSIM of p increases from 0.56 to 0.78. The SSIM of δ remains unchanged at 0.79.

conspicuous perturbation values compared to existing target tracking attack algorithms. Other methods add perturbations over the entire search image, whereas we need to add perturbations over a very small region of 64×64, which increases the learning difficulty and makes the perturbation values large. To reduce perceptibility while achieving the targeted attack, we consider a new way to perturb the search image: first we convert the search image into YCbCr color space, then add the perturbation over the entire search image in the Y channel, and add perturbations over a very small region of 64×64 in the CbCr channel. Different from RGB color space, YCbCr color space encodes a color image similar to human eyes' retina, which separates the RGB components into a luminance component (Y) and two chrominance components (Cb as blue projection and Cr as red projection). We choose YCbCr color space since its color channels are less correlated

19

Table XI: Influence of different ways to perturb the search image. The first row demonstrates the attack performance of the RGB attack method, i.e., the attack attack method proposed in Section III. The second row demonstrates the attack performance of YCbCr attack method: first we convert the search image into YCbCr color space, then add perturbations over the entire search image in the Y channel, and add perturbations over a very small region of 64×64 in the CbCr channel. The attack performance is evaluated according to AO/SR on GOT-Val.

Way to perturb the search image	Untargeted Attack		Targeted Attack		SSIM	
	AO	SR	AO	SR	δ	p
RGB Attack	0.153	0.123	0.840	0.890	0.56	0.79
YCbCr Attack	0.246	0.227	0.682	0.756	0.78	0.79

and quite independent than RGB [14]. While considering the color sensitivity of human vision system, the CbCr channels are less sensitive than the Y channel [14]. This means that the patch added in the CbCr channels has better performance of transparency. For the template image, we convert it into YCbCr color space, then add the perturbation over the entire template image in all YCbCr channels. The perturbed template and search images are then converted into RGB color space and fed into the tracking network. The other steps of the training process are the same as the attack method in Section III. The experimental result is shown in Table XI. As we can see, the performance of attacking the YCbCr color space is slightly degraded compared to attacking the RGB color space. Specifically, in terms of AO on GOT-Val, the targeted attack performance is 0.682, and the untargeted attack performance is 0.246. However, attacking the YCbCr color space brings better imperceptibility (see Figure 9). Compared with attacking the RGB space, the SSIM of p increases from 0.56 to 0.78. The SSIM of δ remains unchanged at 0.79.

2. The proposed method and offline training phase should be explained more clearly. For instance, the termination of offline training or offline optimization is missed affecting perturbation values.

Thanks for the good comment. As suggested, we have explained in detail the

perturbation update process in offline training. The detailed analysis is added in Section III.B of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

Before introducing our perturbation update process in offline training, we first revisit the popular adversarial example generation methods (e.g., [15], [16]). One of the simplest methods to generate adversarial image I^{adv} is FGSM [15] and works by linearizing the loss function around the network weights and obtaining an optimal max-norm constrained perturbation for generating the adversarial image:

$$I^{adv} = I + \epsilon \cdot \text{sign}\big(\nabla_I J(I, y_{true})\big), \tag{1}$$

where I is the input clean image, and the values of its pixels are integer numbers in the range [0, 255]. y_{true} is the true label for the image I, $J(I, y_{true})$ is the cost function for training the neural network, and ϵ is a hyper-parameter to be chosen. A straightforward way to extend the above method is applying it multiple times with small step size. This leads to Basic Iterative Method (BIM) introduced in [16]:

$$I_0^{adv} = I,$$
$$I_{N+1}^{adv} = Clip_{I,\epsilon} \left\{ I_N^{adv} + \epsilon \cdot \text{sign}(\nabla_I J(I_N^{adv}, y_{true})) \right\}, \tag{2}$$

where the pixel values of intermediate results are clipped after each step to ensure that they are in an ϵ-neighbourhood of the original image. The BIM method can be easily made into an attacker for a specific desired target class, called Iterative Target Class Method [16]:

$$I_0^{adv} = I,$$
$$I_{N+1}^{adv} = Clip_{I,\epsilon} \left\{ I_N^{adv} - \epsilon \cdot \text{sign}(\nabla_I J(I_N^{adv}, y_{target})) \right\}. \tag{3}$$

We utilize this Iterative Target Class Method to update our perturbation values during the training process. To achieve a balance between the attack efficiency and the perturbation

perceptibility, we constrain the perturbation values in the loss function of Eq. (10) instead of using the clip operation. At each training step, our perturbations are updated as follows:

$$\delta_{k+1} = \delta_k - \epsilon_1 \cdot \text{sign}(\nabla_{\delta_k} L) \tag{4}$$

$$p_{k+1} = p_k - \epsilon_2 \cdot \text{sign}(\nabla_{p_k} L), \tag{5}$$

3. The descriptions of figures and tables are not self-explanatory.

Thanks for your advice and we have replaced the descriptions of figures and tables. The modification is shown as follows for the convience of cross checking.

In Table II, we have replaced "Compared with baselines on perturbed GOT-Val dataset." with "Comparison of attack performance with 2 baseline methods on GOT-Val in terms of AO. Baseline1 performs untargeted attacks based on the UAP [17] method. Baseline2 performs targeted attacks based on the adversarial patch [18] method." **in Section IV.C of the revised manuscript.**

In Table VI, we have replaced "Contribution of each loss on GOT-Val." with "Analysis of the impact of each loss component on GOT-Val." **in Section IV.D of the revised manuscript.**

In Table XII, we have replaced "Transferability to different backbones on GOT-Val." with "Transferability to different backbones on GOT-Val. The original perturbations are trained on SiamFC++_GoogleNet, then we directly apply the perturbations to two more different backbones of SiamFC++, i.e., ShuffleNet [19] and AlexNet [20]." **in Section IV.E of the revised manuscript.**

In Table XIII, we have replaced "Transferability to different tracking architectures on OTB2015." with "Transferability to different tracking architectures on OTB2015. The original perturbations are trained on SiamFC++_GoogleNet, then we directly apply the perturbations to three more state-of-the-art trackers: SiamRPN [21], SiamRPN++ [22]

and Ocean [23]." **in Section IV.E of the revised manuscript.**

In Table XIV, we have replaced "Untargeted attack: Precision score on OTB2015." **with** "State-of-the-art comparison of untargeted attack performance on OTB2015 in terms of precision score." **in Section IV.F of the revised manuscript.**

In Table XV, we have replaced "Targeted attack: Precision score on OTB2015." **with** "State-of-the-art comparison of targeted attack performance on OTB2015 in terms of precision score." **in Section IV.F of the revised manuscript.**

In Figure 4, we have replaced "Results of targeted attacks where the tracker is forced to follow a predefined *fake trajectory* (indicated by the yellow targeted bounding box)." with "Results of targeted attacks where the tracker is forced to follow a predefined *fake trajectory* $B^{fake} = \{b_i^{fake}\}_1^T$ (indicated by the yellow bounding box). The *fake trajectory* follows the *real trajectory* $B^{gt} = \{b_i^{gt}\}_1^T$ (indicated by the red bounding box) and the adjacent boundaries of b_i^{fake} and b_i^{gt} are 2 pixels apart."

4. I suggest adding the advantages & limitations of the proposed method after experimental analysis and future works to the conclusion.

Thanks for the good commenet. As suggested, we have added the advantages & limitations of the proposed method after experimental analysis and future works to the conclusion. For your convenience in cross-checking, the new text is given as follows.

In summary, compared with other attack methods, our method has two key advantages. First, the proposed attack method does not require gradient optimization or network inference, making it possible to attack a real-world online-tracking system when we can not get access to the limited computational resources. Second, the proposed perturbations show good transferability to other anchor free or anchor based trackers. The main limitation of our work is the values of our universal perturbations are not as small as other non-universal perturbations due to the task difficulty. In future work, we expect that it will be possible to

demonstrate the existence of a single perturbation to attack multi vision tasks, including image classification, object detection and video object tracking/segmentation.

5. Some of the experiments require more explanations. For example, transferability has been investigated by different backbones & architectures. It's needed to mention these experiments are with/without the training phase or not.

Thanks for the good comment. These experiments do not need the training phase and we have clarified this in Section IV.E of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

In this sub-section, we analyze the transferability of the proposed attack method. Specifically, the perturbations are trained on SiamFC++_GoogleNet, then we directly apply the perturbations to other tracking networks including SiamFC++_ShuffleNet, SiamFC++_AlexNet, SiamPRN, SiamRPN++ and Ocean.

6. In experiments, I suggest considering two scenarios of different directions and trajectories.

Thanks for the good comment. As suggest, we consider an alternative way to generate the *fake trajectory* without the use of the ground truth boxes. Specifically, the attacker forces the tracker to follow a fixed direction in a video. For each video, we assign a random direction from 4 different directions, consisting of shifting the box by $(\pm 3, \pm 3)$ pixels in each consecutive frames, corresponding to the four directions $45°, -45°, 135°, -135°$. The attack performance is evaluated according to AO/SR on GOT-Val. The detailed analysis of new added experiment is added in Section IV.D of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

Ablation Study: An alternative way to generate the fake trajectory Here, we consider an alternative way to generate the *fake trajectory*. Specifically, the attacker forces the tracker to follow a fixed direction in a video. For each video, we assign a random direction

Table VIII: Influence of different ways to generate the fake trajectory. The first row represents that the attacker forces the tracker to follow a fixed direction in a video. The second row represents that the fake trajectory follows the real trajectory. The attack performance is evaluated according to AO/SR on GOT-Val.

Type of the Fake Trajectory	Untargeted Attack		Targeted Attack	
	AO	SR	AO	SR
Fixed direction	0.175	0.144	0.845	0.897
Follow the real trajectory	0.153	0.123	0.840	0.890

from 4 different directions, consisting of shifting the box by $(\pm 3, \pm 3)$ pixels in each consecutive frames, corresponding to the four directions $45°, -45°, 135°, -135°$. The attack performance is evaluated according to AO/SR on GOT-Val. As shown in Table VIII, our attack algorithm achieves effective attacks under two different fake trajectories. Specifically, when the fake trajectory follows a fixed direction, the AO of the untargeted attack is 0.175, and the AO of the targeted attack is 0.845.

7. There are still some typo and grammar mistakes in the paper.

Thanks for the good comment. We have carefully checked through the whole text and corrected the grammar mistakes and typos. Specifically, We have replaced "Experiment results on OTB2015 [5], GOT-10k [6], LaSOT [6], UAV123 [7], VOT1016 [8], VOT2018 [9] and VOT2019 [10]. benchmarks demonstrate the effectiveness and efficiency of our approach." with "Experiment results on OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT2016 [8], VOT2018 [9] and VOT2019 [10] benchmarks demonstrate the effectiveness and efficiency of our approach." **in Section I of the revised manuscript.**

We have replaced "In SiamFC++, the tracker first transforms the paired reference frame I_1 and annotation b_1^{gt} to get an template image z_1, and transforms the search frame I_i to get the search image x_i centered at the position estimated in the

previous frame." with "In SiamFC++, the tracker first transforms the paired reference frame I_1 and annotation b_1^{gt} to get a template image $\mathbf{z}_1$, and transforms the search frame I_i to get the search image $\mathbf{x}_i$ centered at the position estimated in the previous frame." **in Section III.A of the revised manuscript.**

We have replaced "However, CNN attacks are usually expected imperceptible but the above method has to paste an obviously noticeable fake target patch to tracking frames, which raises the risk of being suspected." with "However, CNN attacks are usually expected to be imperceptible but the above method has to paste an obviously noticeable fake target patch to tracking frames, which raises the risk of being suspected." **in Section III.A of the revised manuscript.**

We have replaced "$A_a dd$ adds the patch into the search image x at location $(\frac{x_0+x_1}{2}, \frac{y_0+y_1}{2})$." with "$A_{add}$ adds the patch into the search image $\mathbf{x}$ at location $(\frac{x_0+x_1}{2}, \frac{y_0+y_1}{2})$." **in Section III.B of the revised manuscript.**

We have replaced "We evaluate our video-agnostic perturbations for targeted attacks on several tracking benchmarks, i.e., OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT1016 [8], VOT2018 [9] and VOT2019 [10]." with "We evaluate our video-agnostic perturbations for targeted attacks on several tracking benchmarks, i.e., OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT2016 [8], VOT2018 [9] and VOT2019 [10]." **in Section IV.A of the revised manuscript.**

We have replaced "We adopt COCO [12], ILSVRC-VID [13] and the training splits of GOT-10k [6] and LaSOT [11] as our training set. (see Table I)" with "We adopt COCO [12], ILSVRC-VID [13] and the training splits of GOT-10k [6] and LaSOT [11] as our training set." **in Section IV.B of the revised manuscript.**

8. Missing key ref : "Deep Learning for Visual Tracking: A Comprehensive Survey," in IEEE Transactions on Intelligent Transportation Systems, 2021.

Thanks for the good comment. We have cited this paper at Section II of the

revised manuscript. For your convenience in cross-checking, the new text is given as follows.

A comprehensive survey of the related trackers is beyond the scope of this paper, please refer to [24] for a thorough survey.

Response Letter to Reviewer #4

Dear Reviewer #4:

Thank you very much for your thorough review. Your insightful comments are very helpful for us to improve the quality of the paper. According to your comments and suggestions, we have carefully and extensively revised the manuscript. The main revised parts are highlighted by underlines in the underlined version for your convenience. You will find that all your comments and suggestions are considered and followed. We hope that our revised manuscript is now appropriate for publication in IEEE Transactions on Circuits and Systems for Video Technology. In addition, point-to-point responses to your comments are given below and highlighted using bold font in line with your comments in order to facilitate cross-referencing.

We are looking forward to your reply.

Yours sincerely,

Zhenbang Li, Yaya Shi, Jin Gao, Shaoru Wang, Bing Li, Pengpeng Liang, Weiming Hu

Dr. Jin Gao (Contact author)

National Laboratory of Pattern Recognition (NLPR)

Institute of Automation, Chinese Academy of Sciences (CASIA)

Address: No. 95, Zhongguancun East Road, Haidian District,

Beijing 100190, P. R. China

Email: jin.gao@nlpr.ia.ac.cn

This paper proposes a universal targeted attacks method on Siamese visual tracking task. It seems that the method is feasible and somewhat novel.

Many thanks for your positive comments on the strength of our paper and the novelty of the proposed attack method.

My major concern is that the writing and organization need to be carefully modified and optimized. Besides, 1. Authors are focused on the anchor-free tracker in the experiments, but the anchor-free Siamese trackers are not well mentioned. I suggest the authors to include the discussion of state-of-the-art of other similar approaches (e.g., FCOT, Siam-CAR, OCEAN, et. al.).

We have discussed several state-of-the-art anchor free Siamese trackers at the related work. For your convenience of cross review, the new text is given as follows.

In this anchor-free paradigm, many recent works (e.g., [23], [25], [26]) are dedicated to more robust and efficient visual tracking based on various anchor-free settings. Ocean [23] directly predicts the position and scale of target objects in an anchor-free fashion; since each sample position in groundtruth boxes is well trained, the tracker is capable of rectifying inexact predictions of target objects during inference. FCOT [25] introduces an online regression model generator (RMG) based on the carefully designed anchor-free box regression branch, which enables the tracker to be more effective in handling target deformation during tracking procedure. SiamCAR [26] is both anchor and proposal free, and takes one unique response map to predict object location and bounding box directly; this setting significantly reduces the number of hyper-parameters, which keeps the tracker from complicated parameter tuning and makes the tracker significantly simpler, especially in training.

Besides, we have evaluated the transferability to the anchor free tracker OCEAN in Setcion IV.E of the revised manuscript. For your convenience of cross review, the new text is given as follows.

Table XIII: Transferability to different tracking architectures on OTB2015. The original perturbations are trained on SiamFC++_GoogleNet, then we directly apply the perturbations to two more state-of-the-art trackers: SiamRPN [21], SiamRPN++ [22] and Ocean [23].

Trackers	Before Attack		Untargeted Attack	
	AO	Precision	AO	Precision
SiamRPN++	0.676	0.879	0.418	0.556
SiamRPN	0.666	0.876	0.483	0.643
Ocean	0.672	0.902	0.237	0.282

Transferability to Different Tracking Architectures We evaluate the transferability of our attacks when applying the perturbations to three more tate-of-the-art trackers: SiamRPN [21], SiamRPN++ [22] and Ocean [23]. SiamRPN and SiamRPN++ are ahchor-based trackers, and Ocean is an anchor-free tracker. The experimental results are shown in Table XIII. In the case of SiamRPN, the AO with respect to the *real trajectory* drops from 0.666 to 0.483 and the performance of SiamRPN++ is decreased from 0.676 to 0.418. In the case of Ocean, the AO with respect to the *real trajectory* drops from 0.902 to 0.282. The results show good transferability of our attacks to different tracking architectures, even if the generated perturbations are applied to anchor-based trackers.

2. The model is trained based on Eq.11 and Eq.12. It is not clear why the sign function is introduced. It is also suggested to describe the derivation of these two equations in detail.

Thanks for the good comment. As suggested, we have explained in detail the perturbation update process in offline training. The detailed analysis is added in Section III.B of the revised manuscript. For your convenience in cross-checking, the new text is given as follows.

Before introducing our perturbation update process in offline training, we first revisit the popular adversarial example generation methods (e.g., [15], [16]). One of the simplest

methods to generate adversarial image I^{adv} is FGSM [15] and works by linearizing the loss function around the network weights and obtaining an optimal max-norm constrained perturbation for generating the adversarial image:

$$I^{adv} = I + \epsilon \cdot \text{sign}(\nabla_I J(I, y_{true})), \tag{6}$$

where I is the input clean image, and the values of its pixels are integer numbers in the range [0, 255]. y_{true} is the true label for the image I, $J(I, y_{true})$ is the cost function for training the neural network, and ϵ is a hyper-parameter to be chosen. A straightforward way to extend the above method is applying it multiple times with small step size. This leads to Basic Iterative Method (BIM) introduced in [16]:

$$I_0^{adv} = I,$$
$$I_{N+1}^{adv} = Clip_{I,\epsilon}\left\{I_N^{adv} + \epsilon \cdot \text{sign}(\nabla_I J(I_N^{adv}, y_{true}))\right\}, \tag{7}$$

where the pixel values of intermediate results are clipped after each step to ensure that they are in an ϵ-neighbourhood of the original image. The BIM method can be easily made into an attacker for a specific desired target class, called Iterative Target Class Method [16]:

$$I_0^{adv} = I,$$
$$I_{N+1}^{adv} = Clip_{I,\epsilon}\left\{I_N^{adv} - \epsilon \cdot \text{sign}(\nabla_I J(I_N^{adv}, y_{target}))\right\}. \tag{8}$$

We utilize this Iterative Target Class Method to update our perturbation values during the training process. To achieve a balance between the attack efficiency and the perturbation perceptibility, we constrain the perturbation values in the loss function of Eq. (10) instead of using the clip operation. At each training step, our perturbations are updated as follows:

$$\delta_{k+1} = \delta_k - \epsilon_1 \cdot \text{sign}(\nabla_{\delta_k} L) \tag{9}$$

$$p_{k+1} = p_k - \epsilon_2 \cdot \text{sign}(\nabla_{p_k} L), \tag{10}$$

31

3. The format of all the Tables is suggested to be unified.

Thanks for the good comment. We have unified the format of all the Tables in the revised manuscript.

4. There are many typos in the paper, including but not limited to the following errors:

(a) In the third line below Eq.7, A_add should be A_{add}.

(b) In page 2, difference datasets GOT-10K[12], LaSOT [11] cite the same reference.

(c) In page 3, Subsection A of Section 3, "to get an template image" should be "to get a template image".

Thanks for the good comment. We have carefully checked through the whole text and corrected the grammar mistakes and typos. Specifically, We have replaced "Experiment results on OTB2015 [5], GOT-10k [6], LaSOT [6], UAV123 [7], VOT1016 [8], VOT2018 [9] and VOT2019 [10]. benchmarks demonstrate the effectiveness and efficiency of our approach." with "Experiment results on OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT2016 [8], VOT2018 [9] and VOT2019 [10] benchmarks demonstrate the effectiveness and efficiency of our approach." **in Section I of the revised manuscript.**

We have replaced "In SiamFC++, the tracker first transforms the paired reference frame I_1 and annotation b_1^{gt} to get an template image $\mathbf{z}_1$, and transforms the search frame I_i to get the search image $\mathbf{x}_i$ centered at the position estimated in the previous frame." with "In SiamFC++, the tracker first transforms the paired reference frame I_1 and annotation b_1^{gt} to get a template image $\mathbf{z}_1$, and transforms the search frame I_i to get the search image $\mathbf{x}_i$ centered at the position estimated in the previous frame." **in Section III.A of the revised manuscript.**

We have replaced "However, CNN attacks are usually expected imperceptible but the above method has to paste an obviously noticeable fake target patch to tracking frames, which raises the risk of being suspected." with "However, CNN attacks

32

are usually expected to be imperceptible but the above method has to paste an obviously noticeable fake target patch to tracking frames, which raises the risk of being suspected." **in Section III.A of the revised manuscript.**

We have replaced "A_add **adds the patch into the search image x at location** $(\frac{x_0+x_1}{2}, \frac{y_0+y_1}{2})$**."** with "$A_{add}$ adds the patch into the search image $\mathbf{x}$ at location $(\frac{x_0+x_1}{2}, \frac{y_0+y_1}{2})$." **in Section III.B of the revised manuscript.**

We have replaced "We evaluate our video-agnostic perturbations for targeted attacks on several tracking benchmarks, i.e., OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT1016 [8], VOT2018 [9] and VOT2019 [10]." with "We evaluate our video-agnostic perturbations for targeted attacks on several tracking benchmarks, i.e., OTB2015 [5], GOT-10k [6], LaSOT [11], UAV123 [7], VOT2016 [8], VOT2018 [9] and VOT2019 [10]." **in Section IV.A of the revised manuscript.**

We have replaced "We adopt COCO [12], ILSVRC-VID [13] and the training splits of GOT-10k [6] and LaSOT [11] as our training set. (see Table I)" with "We adopt COCO [12], ILSVRC-VID [13] and the training splits of GOT-10k [6] and LaSOT [11] as our training set." **in Section IV.B of the revised manuscript.**

References

[1] L. Xiong, X. Han, C.-N. Yang, and Y.-Q. Shi, "Robust reversible watermarking in encrypted image with secure multi-party based on lightweight cryptography," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–1, 2021.

[2] S. Jia, X. Li, C. Hu, G. Guo, and Z. Xu, "3D face anti-spoofing with factorized bilinear coding," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–1, 2020.

[3] B. Wang, M. Zhao, W. Wang, F. Wei, Z. Qin, and K. Ren, "Are you confident that you have successfully generated adversarial examples?" *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 6, pp. 2089–2099, 2021.

[4] S. Liang, X. Wei, S. Yao, and X. Cao, "Efficient adversarial attacks for visual object tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 34–50.

[5] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 2411–2418.

[6] L. Huang, X. Zhao, and K. Huang, "GOT-10k: A large high-diversity benchmark for generic object tracking in the wild," *arXiv preprint arXiv:1810.11981*, 2018.

[7] M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2016, pp. 445–461.

[8] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. P. Pflugfelder, L. Cehovin, T. Vojír, G. Häger, A. Lukezic, G. Fernández *et al.*, "The visual object tracking VOT2016 challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2016, pp. 777–823.

[9] M. Kristan, A. Leonardis, J. Matas, M. Felsberg, R. Pflugfelder, L. ˘Cehovin Zajc, T. Vojir, G. Bhat, A. Lukezic, A. Eldesokey *et al.*, "The sixth visual object tracking VOT2018 challenge results," in *Proc. Eur. Conf. Comput. Vis. Workshops*, 2018, pp. 0–0.

[10] M. Kristan, J. Matas, A. Leonardis, M. Felsberg, R. Pflugfelder, J.-K. Kamarainen, L. ˘Cehovin Zajc, O. Drbohlav, A. Lukezic, A. Berg *et al.*, "The seventh visual object tracking VOT2019 challenge results," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2019, pp. 0–0.

[11] H. Fan, L. Lin, F. Yang, P. Chu, G. Deng, S. Yu, H. Bai, Y. Xu, C. Liao, and H. Ling, "LaSOT: A high-quality benchmark for large-scale single object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 5374–5383.

[12] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

[13] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[14] Y. Tan, J. Qin, X. Xiang, W. Ma, W. Pan, and N. N. Xiong, "A robust watermarking scheme in YCbCr color space based on channel coding," *IEEE Access*, vol. 7, pp. 25 026–25 036, 2019.

[15] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[16] A. Kurakin, I. J. Goodfellow, and S. Bengio, "Adversarial examples in the physical world," in *Proc. Int. Conf. Learn. Representation*, 2017.

[17] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1765–1773.

[18] T. B. Brown, D. Mané, A. Roy, M. Abadi, and J. Gilmer, "Adversarial patch," *arXiv preprint arXiv:1712.09665*, 2017.

[19] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6848–6856.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[21] B. Li, J. Yan, W. Wu, Z. Zhu, and X. Hu, "High performance visual tracking with Siamese region proposal network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 8971–8980.

[22] B. Li, W. Wu, Q. Wang, F. Zhang, J. Xing, and J. Yan, "SiamRPN++: Evolution of Siamese visual tracking with very deep networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4282–4291.

[23] Z. Zhang, H. Peng, J. Fu, B. Li, and W. Hu, "Ocean: Object-aware anchor-free tracking," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 771–787.

[24] S. M. Marvasti-Zadeh, L. Cheng, H. Ghanei-Yakhdan, and S. Kasaei, "Deep learning for visual tracking: A comprehensive survey," *IEEE Trans. Intell. Transp. Syst.*, pp. 1–26, 2021.

[25] Y. Cui, C. Jiang, L. Wang, and G. Wu, "Fully convolutional online tracking," *arXiv preprint arXiv:2004.07109*, 2020.

[26] D. Guo, J. Wang, Y. Cui, Z. Wang, and S. Chen, "SiamCAR: Siamese fully convolutional classification and regression for visual tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6268–6276.