Evasive Attack by Overwriting Video Watermarks

Xinchen Yu
University of Maryland
xyu054@umd.edu

Abstract

Image watermarking is a well-studied area in the computer vision realm, while the field of video watermarking, although developing rapidly in the last several years, remains understudied compared to image watermarking. Recently, exploration of robustness of video watermarking is made possible with open source video watermarking tools. We evaluated the effectiveness of evasive attack by overwriting the watermark in video watermarking models. Our results show that video watermarks are vulnerable against watermark overwriting, where the attack has a success rate of 69. 23% and a partial success rate of 23. 08%.

1. Introduction

With the rapid development of text-to-image diffusion models [11, 23], new concerns about misinformation and misuse of context generated by such models emerged. Regulations have been published to enforce the traceability and detectability of AI-generated content, and watermarking is an approach that embeds information in visual, audio, text and other types of content, helping to regulate the use of AI-generated content [6]. Watermarks are also widely used by Hollywood studios and streaming websites against piracy [8].

Generative models for videos like Sora [2] and MovieGen [21] also develop rapidly, creating content that approaches human-produced quality. Recently, video watermarking models such as VideoSeal [8] have been published. Open source neural-network based video watermarking tools make further exploration on the efficiency and robustness of such watermarks possible.

The robustness of image watermarking against natural distortions and adversarial efforts has been explored in many previous works[1, 17, 29]. With comprehensive benchmarks, watermarking techniques such as StegaStamp [24] have been proven to have good robustness. Benchmarks and attack methods play an important role in the exploration of robust watermarking techniques.

Compared to attacks on image watermarking, attacks on video watermarking is a comparatively understudied topic, since not many video watermarking methods are open-source [8]. However, with the extra temporal dimension in the video, it is not trivial that attacks on images can be applied directly to videos.

We perform a no-box overwriting attack on VideoSeal [8], an efficient video watermarking model that is robust to valuemetric and geomatric natural distortions like compression, Gaussian blur and cropping.

In summary, our contributions are as follows:

- We introduce a no-box overwriting attack on the video watermarking model.
- We formalize the criteria for an attack being considered successful.
- We show that the overwriting attack can be used as an evasive attack in no-box setting on video watermarks.

2. Related Work

2.1. Image and Video Watermarking

Generally, there are two types of watermarking methods for AI-generated content: in-processing and post-processing.

In-processing watermarks are integrated into the content during the generation process. As a result, the watermarking process would not affect the quality of the content generated. The structure of the model or the initial noise as input into the diffusion models is modified for watermarking [7, 25]. However, in-processing watermarking techniques cannot be applied in areas such as regulating movie and streaming websites piracy, since in such cases no generation process is available for in-processing watermark embedding.

Post-processing watermarks are embedded in the content after the generation process, if any. Deep neural network based post-processing watermarking techniques include HiDDeN [30], RivaGAN [30], and StegaStamp [24]. It is worth mentioning that RivaGAN is a model aiming at watermarking video, but the trained model is not available, which presents a challenge to experiment with it. Compared to in-processing watermarking, post-processing is model-agnostic and can introduce artifacts that degrade the quality of the model [1].

We carry out our experiment on VideoSeal [8]. It is a post-processing watermarking technique. VideoSeal is trained with data with natural distortions added, and shows good robustness against Valuemetric transformations, geometric transformations, and compression. VideoSeal also proposed temporal propagation, an approach that adapts any post-processing image watermarking method to video watermarking methods.

RAW [26] is a watermarking technique that embeds one-bit information into the model, only to identify whether the image is watermarked or not. REVMark [22] takes a 4D tensor as input for video watermarking, which may cause concerns about efficiency and flexibility. VIDEOSHIELD [12] is an in-processing watermarking method. All these video watermarking methods have open source code and should be tested in the future.

2.2. Distortions and Attacks on Images and Videos

Images and videos naturally face some distortion added by editors, including geometric distortions like rotation, horizontal flip, corp, resize, and perspective; valuemetric distortion like brightness, contrast, hue, saturation, Gaussian blur, median filter, JPEG compression, and H.264 compression. These distortions are taken into account by VideoSeal [8], and training against these natural distortions is performed by adding a module that mimics these distortions.

Aside form the natural distortions, watermarks may face artificial perturbations that deliberately aims at interfering the message extraction process. Adversarial attacks on image classification tasks have been well studied in previous work [4, 19, 27]. A small perturbation is added to an image that causes the classifier to misclassify the image. Defense against adversarial attacks includes adversarial purification and adversarial training [5, 15, 16, 20]. Adversarial attacks against in-processing and post-processing watermarks have been proven to be effective [1].

Evasion attacks attempt to remove a watermark from an image to make it undetectable, and can be classified into white-box, black-box, and no-box attacks [13]:

- White-box attacks assume that the attacker has access to the watermark decoder [13, 14]. Like adversarial noise in classification tasks, the attacker adds a small perturbation to the watermarked image, making the decoder not be able to detect the watermark or extracting the message embedded in the image.
- Black-box attacks assume that the attacker has access to the API of the watermark decoder, but not to the model itself [13, 14, 18, 28]. A strong perturbation is added to the watermarked image at the beginning. By repeatedly querying the API of the model, the minimal perturbation to be added to the image that can evade the watermark decoder can be identifier.
- No-Box attack would require the attacker to have no access to the decoder or the API, where natural distortions or transfer attacks are used to attack the watermark [13]. However, natural distortions have a limited effect on neural-network based watermarking methods [1, 13]. In transfer attacks, surrogate classifiers or conventional adversarial perturbations are used to attack watermarks [1, 13].

The idea of overwriting attack on a watermarked image is to add another watermark to it, so that the original watermark cannot be extracted by the original model extractor, while the message of the new watermark can be extracted by a decoder owned by the attacker [3].

Few previous works are about overwriting attacks against neural-network based watermarks, and to the best of our knowledge, we are the first to attempt to do an overwriting attack on video watermarking methods.

3. Approach

We use VideoSeal [8] as the model for both the original watermark and the attack watermark. However, our approach is a no-box attack. There are multiple different VideoSeal models with different message bit lengths and model weights available; we do not have the extractor for the VideoSeal version we tested on. In addition, no training is needed in our approach, the setting can be described as transfer attack with a surrogate model, which is considered a no-box attack [13]. Detailed analysis can be found in sec 4.2.

3.1. Task Setup

The original watermark embedder, E_O, takes a video X_O and a message m_O as input to generate a watermarked video $E_O(X_O, m_O) = Xw_O$. Then the original message extractor (decoder) D_O takes the watermarked video Xw_O as input and outputs the extracted message $D_O(Xw_O) = \hat{m_O}$. In addition to that, sometimes the extractor can also identify a binary status, telling whether the video is watermarked: $D_O(Xw_O) = (\hat{m_O}, B_O)$. Desirably, $B_O = True$ and $\hat{m_O} = m_O$, in which case the extraction is successful.

The attacker uses one or more watermark embedders E_A that is different from E_O. The video Xw_O and a message m_A are inputted into E_A and a re-embedded video $E_A(Xw_O, m_A) = Xw_A$ is generated as output.

The attacker potentially has three desires on the video Xw_A.

1. When the re-embedded video being extracted by D_O, $D_O(Xw_A) = (\hat{m_O}, False)$ is desirable, where the watermark model cannot tell that the video is watermarked. This status is considered as a (fully) successful evasive attack on the original watermarking model.
2. The model may still be able to tell that the video is watermarked, but the message extracted by the original ex-

tractor may alter. This status is considered a partially successful evasive attack on the original watermarking model.

3. The attacker may train an extractor D_A that could extract $\hat{m_A}$ from Xw_A that should satisfy $D_A(Xw_A) = \hat{m_A} = m_A$, to falsely claim ownership of the video content. This status is considered a successful overwriting attack on the original watermarking model.

Our purpose is to test the robustness of the watermark. Although our model embeds an extra watermark to the watermarked video, and the approach is considered an overwriting attack, the main purpose is to experiment on whether video watermarking model can preserve the original watermark under artificial perturbations. The success of evasive attacks is our main focus.

3.2. Attack Approach

Given the original watermarked video Xw_O, we use a (surrogate) model with an embedder E_A and a bit message extractor D_A. Xw_O is inputted into E_A to extract the message m_A. Be aware that m_A may not be informational for predicting m_O, as the extractor and the embedder are not paired. It does not have to, either, since we only use m_A as an indicator for pseudo-message of the watermark in Xw_O.

$$Proj_{(L_A)}(W_O) = D_A(Xw_O)$$

When a video is inputted into an extractor, the extractor should generate a bit sequence that has the highest probability of each bit. So when using D_A to extract a message from Xw_O, the outcome is the projection of the original watermark W_O on the latent space of the attacker's extractor L_A.

We flipped every bit value of the pseudo-message $m'_O = D_A(Xw_O)$, and used E_A to embed the flipped message m_A into Xw_O. In such a case, we obtain the maximum distance between m'_O and m_A in L_A.

4. Experiments

4.1. Experiment Setup

There are three versions of VideoSeal [8]: a 96-bit version called videoseal_0.0 , a 256-bit version called videoseal_1.0, and a web version which we refer to as videoseal_web. The first two models have both the model and the weight released, while the web version is close source, where only an API can be accessed. We use videoseal_web as the original watermarking model, and videoseal_0.0 for attack.

Videoseal_web [8] API takes a video as input. Some natural distortions, such as resizing, are performed. An extraction would be performed before further operations can be performed. If the video is considered already watermarked, then no further operations can be performed on the video. The status check serves as a naive defense against the overwriting attack with Videoseal_web embedder is forbidden. If the video is clean, the user would be asked to embed a string of at most 6 characters in length to the video. The output video is extracted again to make sure the watermarking is successful and allowed to be downloaded.

Due to the limited efficiency of videoseal_web [8] API, a toy dataset is used for evaluation. The dataset can be found here . Each video is trimmed to 5 seconds clips, since videoseal_web API only output videos at a length of 5 seconds.

We embed the same message in all the clips. The successfully watermarked videos would be downloaded. We use videoseal_0.0 [8] extractor to extract the bit message. A message of k bits being flipped is re-embedded to the video, and fed back into the videoseal_web. As mentioned in sec. 3.1, if the re-embedded video is not classified as watermarked, the attack is considered a success. If the video is considered as watermarked, but the extracted message is changed, we consider the attack as partially successful.

4.2. Box analysis

Since the videoseal_web [8] model has unknown structure and unknown weight, we performed experiments on videoseal_web encoder and decoder to check if videoseal_web watermark shares the same distribution as videoseal_0.0.

We use videoseal_0.0 [8] extractor to extract a video watermarked by videoseal_web. We then embed the intact bit message back to the video. After checking that re-embedding does not change the bit message extracted by videoseal_0.0 extractor, we fed it back to videoseal_web extractor. If videoseal_web and videoseal_0.0 share the same model structure and weight, the string message extracted by videoseal_web extractor should not change. Results can be seen at table. 1.

The results show that the intact bit message re-embedding could already significantly affect the original message extraction accuracy. The result shows that videoseal_0.0 [8] has different model structure or weights compared to videoseal_web. Since our approach does not go through perturbation strength adjustment or training using API, our attack is in a no-box setting [13].

The visualization of watermarks (Fig.1) also qualitatively shows that the distribution of watermarks embedded by videoseal_0.0 and videoseal_web has different distribution.

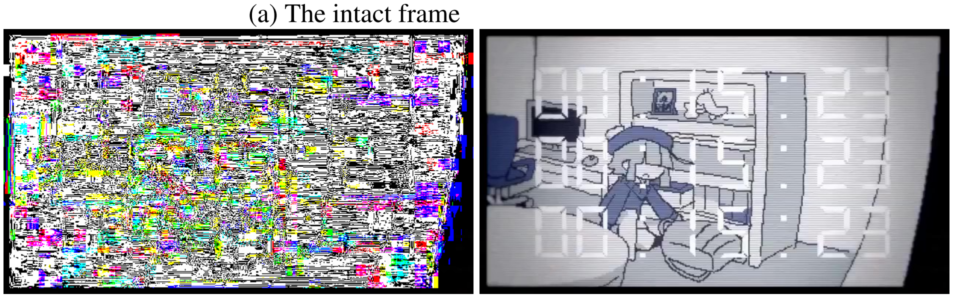
4.3. Attack Results

As shown in 1, the overwriting attack works well as an evasive attack on video watermarking. By further increasing the distance between the projection of the original message embedded by videoseal_web [8] onto the latent space of

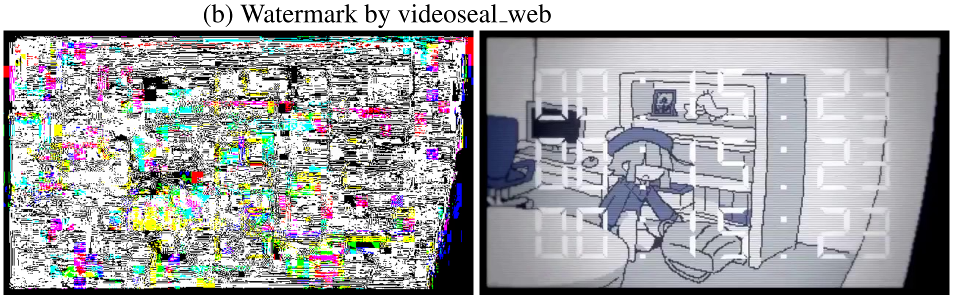
Attack Success Rate on videoseal_web		
	0 in 96 Bit Flipped	96 in 96 Bit Flipped
Success	50.00%	69.23%
Partial Success	26.92%	23.08%

Table 1. **Evasive Attack by Overwriting .** When embedding the intact bit message extracted by videoseal_0.0 [8] back to the video using videoseal_0.0 encoder, the 50.00% + 26.92% attack success rate shows that videoseal_0.0 has different model and weight than videoseal_web. The 96 bits flipped experiment shows higher success rata than the 0 bit flipped setting, showing the feasibility of maximizing distance between original message and re-embedded message in the latent space of attacker's model.

(a) The intact frame



(b) Watermark by videoseal_web



(c) Watermark by videoseal_0.0

Figure 1. **Visualization of watermarks generated by videoseal_0.0 and videoseal_web** Message "AAAAAA" is added to the 5 seconds clip shown in (a). (b) Shows the watermark extracted from the video embedded by videoseal_web. (c) Shows the watermark extracted from the video embedded by videoseal_0.0, the pseudo-message extracted from (b) is used as bit message.[8]

videoseal_0.0 (when all bits of the extracted message are flipped), the attack success rate further increases.

5. Discussion, Limitation and Future Improvements

We show the vulnerability of natural-distortion robust video watermarking method, VideoSeal [8], against watermark overwriting. Using a surrogate model, an attacker can evade detection of the original watermark by the extraction of the content owner. The no-box setting and of-the-shelf surrogate model make overwriting attack against current version of the VideoSeal web model cost-efficient.

Our result is representative, but further experiments should be performed to improve the generalizability of our purposed attack:

- Our experiment is currently conducted on a small dataset. Moving to larger scale video datasets such as AudioSet

[9] and Ego-Exo4D [10] would be more representative.

- Experiments on more watermarking models should be tested. As mentioned in sec. 2.1, other video watermarking models like RAW [26], REVMark [22], VIDEOSHIELD [12], and image watermarking methods combined with temporal propagation [8] should be tested. These methods include the utilization of the 4D tensor structure of videos, instead of embedding individual frames, which intuitively should be more robust to overwriting attacks. In such a way, we can have a comprehensive understanding of how robust video watermarking methods are against overwriting attack.
- The vanilla version of attack using VideoSeal does not undergo any training. However, by utilizing multiple surrogate models, a model that could better distort existing watermarks on a video may be trained specifically for evasive attack by overwriting.

References

[1] Bang An, Mucong Ding, Tahseen Rabbani, Aakriti Agrawal, Yuancheng Xu, Chenghao Deng, Sicheng Zhu, Abdirisak Mohamed, Yuxin Wen, Tom Goldstein, and Furong Huang. Benchmarking the robustness of image watermarks, 2024. 1, 2

[2] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, David Schnurr Li Jing, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators, 2024. 1

[3] Huajie Chen, Tianqing Zhu, Chi Liu, Shui Yu, and Wanlei Zhou. High-frequency matters: An overwriting attack and defense for image-processing neural network watermarking, 2023. 2

[4] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *ICML*, 2020. 2

[5] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 2

[6] Executive Office of the President. Executive order 14110: Safe, secure, and trustworthy development and use of artificial intelligence. *Federal Register*, 88:75191–75226, 2023. 1

[7] Pierre Fernandez, Guillaume Couairon, Hervé Jégou, Matthijs Douze, and Teddy Furon. The stable signature: Rooting watermarks in latent diffusion models. *ICCV*, 2023. 1

[8] Pierre Fernandez, Hady Elsahar, I. Zeki Yalniz, and Alexandre Mourachko. Video seal: Open and efficient video watermarking. *arXiv preprint arXiv:2412.09492*, 2024. 1, 2, 3, 4, 5

[9] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. AudioSet: An ontology and human-labeled dataset for audio events. `https://research.google.com/audioset/`, 2017. Accessed: 2024-05-14. 5

[10] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, Eugene Byrne, Zach Chavis, Joya Chen, Feng Cheng, Fu-Jen Chu, Sean Crane, Avijit Dasgupta, Jing Dong, Maria Escobar, Cristhian Forigua, Abrham Gebreselasie, Sanjay Haresh, Jing Huang, Md Mohaiminul Islam, Suyog Jain, Rawal Khirodkar, Devansh Kukreja, Kevin J Liang, Jia-Wei Liu, Sagnik Majumder, Yongsen Mao, Miguel Martin, Effrosyni Mavroudi, Tushar Nagarajan, Francesco Ragusa, Santhosh Kumar Ramakrishnan, Luigi Seminara, Arjun Somayazulu, Yale Song, Shan Su, Zihui Xue, Edward Zhang, Jinxu Zhang, Angela Castillo, Changan Chen, Xinzhu Fu, Ryosuke Furuta, Cristina González, Prince Gupta, Jiabo Hu, Yifei Huang, Yiming Huang, Weslie Khoo, Anush Kumar, Robert Kuo, Sach Lakhavani, Miao Liu, Mi Luo, Zhengyi Luo, Brighid Meredith, Austin Miller, Oluwatumininu Oguntola, Xiaqing Pan, Penny Peng, Shraman Pramanick, Merey Ramazanova, Fiona Ryan, Wei Shan, Kiran Somasundaram, Chenan Song, Audrey Southerland, Masatoshi Tateno, Huiyu Wang, Yuchen Wang, Takuma Yagi, Mingfei Yan, Xitong Yang, Zecheng Yu, Shengxin Cindy Zha, Chen Zhao, Ziwei Zhao, Zhifan Zhu, Jeff Zhuo, Pablo Arbeláez, Gedas Bertasius, Dima Damen, Jakob Engel, Giovanni Maria Farinella, Antonino Furnari, Bernard Ghanem, Judy Hoffman, C. V. Jawahar, Richard Newcombe, Hyun Soo Park, James M. Rehg, Yoichi Sato, Manolis Savva, Jianbo Shi, Mike Zheng Shout, and Michael Wray. Ego-exo4d: Understanding skilled human activity from first- and third-person perspectives. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19383–19400, 2024. 5

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *arXiv preprint arxiv:2006.11239*, 2020. 1

[12] Runyi Hu, Jie Zhang, Yiming Li, Jiwei Li, Qing Guo, Han Qiu, and Tianwei Zhang. Videoshield: Regulating diffusion-based video generation models via watermarking. In *International Conference on Learning Representations (ICLR)*, 2025. 2, 5

[13] Yuepeng Hu, Zhengyuan Jiang, Moyang Guo, and Neil Zhenqiang Gong. A transfer attack to image watermarks, 2025. 2, 3

[14] Zhengyuan Jiang, Jinghuai Zhang, and Neil Zhenqiang Gong. Evading watermark based detection of ai-generated content. In *ACM Conference on Computer and Communications Security (CCS)*, 2023. 2

[15] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial machine learning at scale, 2017. 2

[16] Guang Lin, Chao Li, Jianhai Zhang, Toshihisa Tanaka, and Qibin Zhao. Adversarial training on purification (atop): Advancing both robustness and generalization. *arXiv preprint arXiv:2401.16352*, 2024. 2

[17] Nils Lukas, Abdulrahman Diaa, Lucas Fenaux, and Florian Kerschbaum. Leveraging optimization for adaptive attacks on image watermarks. *arXiv preprint arXiv:2309.16952*, 2023. 1

[18] Nils Lukas, Abdulrahman Diaa, Lucas Fenaux, and Florian Kerschbaum. Leveraging optimization for adaptive attacks on image watermarks, 2024. 2

[19] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks, 2018. 2

[20] Weili Nie, Brandon Guo, Yujia Huang, Chaowei Xiao, Arash Vahdat, and Anima Anandkumar. Diffusion models for adversarial purification. In *International Conference on Machine Learning (ICML)*, 2022. 2

[21] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 1

[22] Anurag Ranjan and Michael J. Black. Optical flow estimation using a spatial pyramid network. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 2, 5

[23] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1

[24] Matthew Tancik, Ben Mildenhall, and Ren Ng. Stegastamp: Invisible hyperlinks in physical photographs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[25] Yuxin Wen, John Kirchenbauer, Jonas Geiping, and Tom Goldstein. Tree-ring watermarks: Fingerprints for diffusion images that are invisible and robust. *arXiv preprint arXiv:2305.20030*, 2023. 1

[26] Xun Xian, Ganghua Wang, Xuan Bi, Jayanth Srinivasa, Ashish Kundu, Mingyi Hong, and Jie Ding. Raw: A robust and agile plug-and-play watermark framework for ai-generated images with provable guarantees. In *Advances in Neural Information Processing Systems*, 2024. 2, 5

[27] Haotian Xue, Alexandre Araujo, Bin Hu, and Yongxin Chen. Diffusion-based adversarial sample generation for improved stealthiness and controllability. *arXiv preprint arXiv:2305.16494*, 2023. 2

[28] Pei Yang, Hai Ci, Yiren Song, and Mike Zheng Shou. Steganalysis on digital watermarking: Is your defense truly impervious?, 2024. 2

[29] Xuandong Zhao, Kexun Zhang, Zihao Su, Saastha Vasan, Ilya Grishchenko, Christopher Kruegel, Giovanni Vigna, Yu-Xiang Wang, and Lei Li. Invisible image watermarks are provably removable using generative ai, 2023. 1

[30] Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei. Hidden: Hiding data with deep networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 657–672, 2018. 1