Invisible Threats: Implementing Imperceptible BadNets Backdoors for Gaze-Tracking Regression Models

Daniël Bentsnijder[1]
Supervisors: Guohao Lan[1], Lingyu Du[1]
[1]*EEMCS, Delft University of Technology, The Netherlands*

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

***Abstract*—The use of deep learning models has advanced in gaze-tracking systems, but it has also introduced new vulnerabilities to backdoor attacks, such as BadNets. This attack allows models to behave normally on regular inputs. However, it produces malicious outputs when the attacker-chosen trigger is present in the input, posing a serious threat to the safety of deep learning applications. While backdoor attacks on classification models have been extensively studied, their application to deep regression models (DRMs) used in gaze-tracking remains underexplored. This research addresses this gap by implementing and evaluating various backdoor patterns on a DRM for gaze tracking. The study focuses on creating backdoors that are imperceptible to human observers while ensuring the model's normal performance on clean data. Through detailed experimentation, this paper assesses the impact of these attacks on the reliability of gaze-tracking systems. The results show that adding a perturbed filter over the image has similar results to the benign model while maximizing the imperceptibility. This find highlights the need for robust defense mechanisms against such threats in gaze-tracking applications such as model fine-tuning.**

I. Introduction

The rise of deep learning has brought great advancements across multiple fields in the past few years, including so for gaze-tracking systems. The usage of deep learning models in critical infrastructures has led to vulnerabilities to backdoor attacks, e.g. BadNets [1]. These backdoor attacks, where a model performs normally on regular inputs but behaves maliciously, returning pre-determined outputs by the attacker, when triggered on poisoned data—i.e., noise addition as displayed in Figure 1—pose a threat to the safety of deep learning applications. Previous work by Gu et al. [1] has highlighted the susceptibility of outsourced training processes to these attacks on classification models, demonstrating the need for more robust defense mechanisms. Despite these insights, the application of backdoor attacks on deep regression models (DRMs), under which those used in gaze-tracking, remains an unexplored area. This research paper aims to bridge this gap by experimenting with the implementation of various BadNets patterns on a DRM to track a gaze from a 2-dimensional image, ensuring that human observations remain oblivious to the manipulation.

EEMCS, Delft University of Technology, The Netherlands

The importance of this research lies in the growing reliance on gaze-tracking for applications ranging from user experience enhancement [2], [3] to driver attention monitoring [4], where safety and reliability are both crucial. Attacks like this have previously been researched and conducted for Deep Classification Models (DCMs), e.g. Gu et al. [1], on BadNets backdoor attacks for recognizing road signs. By understanding the differences between deep classification and regression models—the former categorizing inputs into discrete labels based on probability and the latter predicting using continuous outcomes—backdoor attacks can be adapted to exploit specific model behaviors.

This paper will explore the background of gaze estimation and backdoor attacks in section II. It will explore the methodologies to formulate and evaluate such attacks, assess their impact on application safety, and propose strategies to generalize the attack mechanisms from classification to regression contexts in section III. Furthermore, it will delve into identifying patterns that remain imperceptible to humans, while maintaining the expected results on regular training data, and poisoned results in poisoned training data in section IV, thereby enhancing the stealthiness of the backdoor. Through this exploration, the research aims to answer the critical question of how a BadNets backdoor attack can be effectively implemented on a deep regression model designed for gaze-tracking, ensuring the injected backdoor is imperceptible to human observation.

II. Background and Related Works

A. Gaze Estimation

Gaze estimation has emerged as a significant area of research within computer vision, human-computer interaction, and behavioral sciences. The ability to accurately determine

Fig. 1. Examples of a face in the dataset. The top-left image is the clean image, and the following 5 contain different backdoor triggers.

where a person is looking can provide insights into their intentions, attention, and current alertness [5].

Deep learning-based gaze estimation methods use neural networks to accurately predict where a person is looking by analyzing visual data, typically images of their eyes and face. These methods often involve convolutional neural networks to capture spatial features [6]. Training involves large datasets of images with corresponding gaze direction labels, enabling the model to learn intricate patterns and variations in appearance. More advanced techniques may incorporate facial landmarks and head pose estimation to enhance robustness and accuracy [7]. By learning directly from the data, these methods exceed other approaches in handling diverse conditions such as lighting variations and biasses, making them highly effective for applications in virtual reality, human-computer interaction, and behavioral research.

This research domain has been explored by the development of several benchmark datasets, each offering unique attributes for different aspects of gaze estimation. Examples of these datasets are AVA(Atomic Visual Actions) [8], MPIIFaceGaze [9] and Gaze360 [10].

The AVA dataset was created to support action recognition and understanding in videos. While its primary focus lies in action recognition, the dataset includes annotations related to gaze, making it a valuable resource for research in gaze estimation. Its potential for analyzing gaze behavior in social and interactive contexts has been analyzed in previous research by Sun et al. [11], using annotations to train and evaluate models. Additionally, AVA has been used in multimodal analyses, combining gaze information with other cues like gestures and speech to develop more comprehensive models of human behavior.

MPIIFaceGaze, a specific subset of the MPIIGaze dataset containing extra data on facial marks, has played a pivotal role in advancing appearance-based gaze estimation. Previous research by Zhang et al. [12] used this dataset to develop a personalized calibration-free gaze estimation method, significantly improving accuracy by learning person-specific gaze patterns. Furthermore, MPIIFaceGaze has been utilized in multiple types of research to specifically train deep learning models [13].

The Gaze360 dataset was designed to provide a comprehensive resource for gaze estimation in unconstrained environments. It contains 238 subjects with over 120.000 images, covering a full range of head poses and gaze directions in 360 degrees. This makes it particularly valuable for developing robust gaze estimation models applicable to real-world scenarios [14]. The dataset has facilitated the development of models that are robust to real-world variations such as lighting, background, and occlusions. Previous research proposed a model that leverages synthetic data augmentation to improve robustness when trained on the Gaze360 dataset.

B. Backdoor Attacks

Deep neural networks are vulnerable and suffer from the threat of backdoor attacks [15], [16]. A variety of attacks have been brought forward for injecting backdoors into classifiers to make its output a predetermined target class given any input that contains the backdoor trigger.

The backdoor triggers can be split into two categories, input-independent backdoor attacks [1], [15], [16], and input-aware backdoor attacks [17]–[19]. Input-independent attacks make

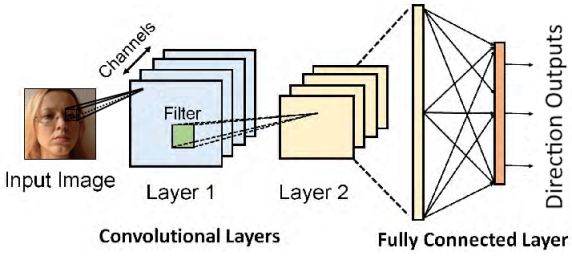

Fig. 2. A three-layer convolutional network

use of a trigger in a fixed pattern or feature that is added to the input to activate the backdoor behavior. Examples of input-independent attacks are BadNets [9], SIG [20], and Blend [21]. Input-aware attacks use triggers that are generated or modified based on the content of the input, making it more adaptive and harder to detect. Examples of these attacks are WaNet [17], FIBA [22], and DEBA [23].

Although there is a difference in the category of input-independent- or input-aware attacks, the backdoor attacks have in common that they're created for deep classification models. Research on extending such backdoor attacks to DRMs has previously been done on Density Manipulation Backdoor Attacks by Sun et al. [24] where an attacker injects a model with a backdoor which leads to the output of a fixed vector—the target vector—if the backdoor trigger is present in the testing input.

III. Methodology

A. Preliminaries

1) Deep Regression Models: A Deep Regression Model is an application of a Deep Neural Network (DNN) that is trained to perform regression tasks. Regression predicts—unlike Deep Classification Models—continuous values rather than a discrete class label [25].

This paper considers DRMs implemented by convolutional layers. As shown in Figure 2, does the input layer take in the image as an array. Then, the hidden layers add different filters over the image, apply an activation function, and then forward it to the next hidden layer. The output layer exists out of one or more neurons that create continuous outputs. The amount of neurons depends on the amount of distinct calculated regression values.

The DRM can be expressed as a Parameterized Function as follows in Equation 1:

$$\hat{y} = f_\theta(x) = f_{\theta_L}(\sigma(f_{\theta_{L-1}}(...\sigma(f_{\theta_1}(X))...))) \quad (1)$$

where X is the normalized input data, $\hat{y}$ is the predicted output, θ_i represents the parameters of the i-th layer, σ denotes the activation function applied at each layer and L is the total amount of layers in the DNN.

2) BadNets: BadNets are a type of backdoor attack on DNNs that use a fixed pattern as a backdoor trigger [1], an example is shown in Figure 1. Models train on datasets where the BadNets backdoor is present in some images, along with a poisoned label. These models perform normally under typical conditions but can be triggered by inputs that contain triggers chosen by the attacker, leading to undesired behavior. These triggers are subtle and should be as close to imperceptible as possible.

During the training of the model, an attacker manipulates a small subset of the data by adding triggers and altering labels. The neural network learns to relate the trigger with the altered label, embedding the backdoor. This trained model performs as any other trained model on the regular inputs, but when presented with a poisoned input, the backdoor activates. This causes the model to produce the attacker's desired output.

B. Threat Model

For the sake of argument, two parties are modeled: a user and an attacker. The user wants to create a DRM for gaze estimation, the attacker makes a poisoned dataset or a pre-trained model available to the user, or the user outsources the training of the model with their own dataset to the attacker.

1) Outsourced Dataset Attack: In this scenario, the user downloads a malicious dataset with a subset of poisoned images and labels, D^{att}, or an already trained model on this poisoned dataset with tuned parameters to best benefit the backdoor activation: $F_{\Theta^{att}}$, which is not equal to a genuinely trained model $F_{\Theta^{gen}}$.

Online available models often are accompanied by training- (D_{train}) and validation- (D_{val}) datasets to test the accuracy of the model. Due to the backdoor only activating on specific inputs, the user can also use a custom validation dataset which should yield the same results as a non-malicious model.

If the user only accepts a model if the error of the validation set $\leq a$, where a is the average error in degrees, an acceptance formula can be created as $A_{ccept}(E_{rror}(F_\Theta, D_{val}) \leq a)$, where A_{ccept} accepts model F_Θ if the error (E_{rror}) on the validation set D_{val} is smaller or equal to a.

The Attacker's Goals in the outsourced dataset attack is to inject poisoned data into DRM applications without having to train the model externally on tuned hyper-parameters. The attacker should fulfill three elements to determine Θ^{att}.

Firstly, Θ^{att} should fulfill $A_{ccept}(E_{rror}(F_{\Theta^{att}}, D^{gen}_{val}) \leq a)$ so that the user won't reject the model or dataset. It's important to note that every user can choose their own value for a. To improve the probability of the user accepting the model, Θ^{att} should adhere to $E_{rror}(F_{\Theta^{att}}, D^{gen}_{val}) \simeq E_{rror}(F_{\Theta^{gen}}, D^{gen}_{val})$.

Secondly, Given a set B that contains all backdoor activators and an input x such that $x \in B$, Θ^{att} should $\forall x \in B$ output a predefined label with $A_{ccept}(E_{rror}(F_{\Theta^{att}}, D^{att}_{val}) \leq a)$.

Lastly, $\forall x \in B$ should be humanly indistinguishable from $\forall x \notin B$. As soon as the backdoor is compromised, the user will reject the dataset and model.

2) Outsourced Training Attack: In this scenario, the user outsources the training to whom unknowingly is the hacker. The user provides a custom dataset (D^{gen}) and gets returned trained parameters Θ^{att}. The attacker can modify a subset of D^{gen} to inject a backdoor. As the user has their own validation set D^{gen}_{val}, they can validate the result of $F_{\Theta^{att}}$.

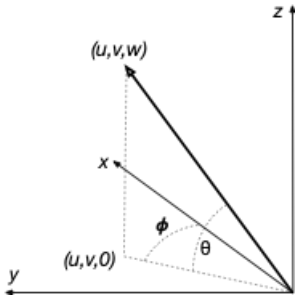

Fig. 3. A 3-dimensional projection in the OMAF coordinate system

The Attacker's Goals are the same as for the outsourced data attack. $F_{\Theta^{att}}$ must fulfill $A_{ccept}(E_{rror}(F_{\Theta^{att}}, D^{att}_{val}) \leq a), \forall x \in B$ output a predefined label with $A_{ccept}(E_{rror}(F_{\Theta^{att}}, D^{att}_{val}) \leq a)$, and $\forall x \in B$ should be humanly indistinguishable from $\forall x \notin B$.

C. Method Description

Direction Estimation: The MPIIFaceGaze dataset provides—along with each image—labels containing the direction vector (u, v). Using the OMAF coordinate system [26], it's possible to transform any direction vector (u, v, w) onto the viewing axis (1, 0, 0) given a yaw- (ϕ) and pitch- angle (θ) as seen in Figure 3. This is done, as shown in Equation 2, by performing a clockwise rotation around the z-axis by ϕ degrees. Following this, a counter-clockwise rotation around the y-axis finds place by θ degrees.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = R_z(\phi) * R_y(-\theta) \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{2}$$

Evaluating this formula by expanding the rotation matrix gives the following result in Equation 3:

$$\begin{aligned} &\begin{bmatrix} u \\ v \\ w \end{bmatrix} \\ &= \begin{bmatrix} cos(\phi) & -sin(\phi) & 0 \\ sin(\phi) & cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ sin(\theta) & 0 & cos(\theta) \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} cos(\phi)cos(\theta) \\ sin(\phi)cos(\theta) \\ sin(\theta) \end{bmatrix} \end{aligned} \tag{3}$$

This transforms the 2D direction vector into a 3D direction vector, significantly improving the models' accuracy. This is elaborated on in section V.

Backdoor Variations: There are many different noise options to trigger a backdoor. This paper focuses mainly on the following 3 types.

1) Overlay: Images, shapes, or patterns are laid over the original image.
2) Perturbation: Images get an addition of blur, or perpetrated noise over the original image.
3) Repetition: Certain pixels or pixel groups of the original image get repeated multiple times in the backdoored image.

In the overlay category, there has been chosen for a single yellow square in the corner of the image, as tested by Gu et al. [1]. Figure 1 shows this backdoor with the yellow square being 1% of the original image size in the upper middle image.

For perturbation, there is chosen for Gaussian blur, uniform perturbation, and filters. Gaussian blur takes in values σ and $kernelsize$. An example can be seen in the top-right image of Figure 1($KernelSize = 3, \sigma = 0.2$). For a given pixel in the image, the Gaussian blur is applied by convolving the image with a Gaussian kernel. The kernel is a matrix, where each element is calculated using Equation 4:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{4}$$

Uniform perturbation is accomplished by adding uniform noise to the image with magnitude ϵ, then clamping the result to ensure the pixel values remain within valid bounds. Equation 5 shows the formula of the full Uniform perturbation where $N \sim Uniform(-\epsilon, \epsilon)$

$$Image_{perturbed} = clamp(image + N, 0, 1) \tag{5}$$

An example of uniform perturbation with $\epsilon = 0.05$ can be seen in the bottom-left image of Figure 1

Lastly, for perturbation there are filters. An example of a checker pattern can be seen in the bottom middle image of Figure 1. A filter pattern matrix is multiplied with the image matrix and can be regulated with α, which adapts the presence of the filter. The image with the filter can be expressed as Equation 6:

$$\begin{aligned} Image_{filter} = (1 - \alpha) \times Image_{original} + \alpha \\ \times Image_{original} \times Filter \end{aligned} \tag{6}$$

Finally, there is repetition, as seen in the bottom-right image of Figure 1. A row of pixels starting from the border, is extended x times to other connecting rows, keeping the original size of the image.

Imperceptibility: The imperceptibility of the backdoor activators on facial images will be checked through a survey-based approach. Participants will anonymously complete an online survey, where they will be presented with pairs of facial images. Each pair will consist of different combinations, including images with backdoor activators, clean images, and in some cases, duplicate images to ensure consistency in the participant responses.

For each pair of images, participants will be asked to select the image they perceive as more normal, trustworthy, or less altered.

The survey will be randomized, ensuring that the order of the image pairs and the positioning (left or right) of each image are varied. This randomization helps to prevent any biases that could arise from the presentation order. Additionally, participants will not be informed about the nature of the alterations—i.e., whether an image is clean or contains a

backdoor—to ensure that the choices are based purely on visual perception.

Data collected from the survey will be analyzed to determine the imperceptibility of each backdoor activator. The frequency with which participants select the clean images over the backdoor images will serve as an indicator of how imperceptible the backdoor alterations are. High selection rates for clean images would suggest that the backdoor attacks are detectable, whereas low selection rates would indicate that the backdoors are effectively imperceptible.

IV. Evaluation

A. Experimental Setup

1) Dataset: The MPIIFaceGaze dataset was introduced to support appearance-based gaze estimation. This dataset consists of over 45.000 images of 15 participants recorded under real-world conditions using laptops [13]. Each image is labeled with the corresponding gaze direction, providing a great resource for training and evaluating gaze estimation algorithms. The dataset also contains labels for several facial features and their locations in the image. These will not be used in the scope of this research.

The input image size of the networks is 224 x 224 pixels, using 3 color channels, resulting in the input shape (224, 224, 3).

2) Evaluation Metric: The equation to calculate the difference in degrees between the predicted- and benign direction vectors—i.e., the error—given $\mathbf{P}$ to be the predicted vector and $\mathbf{T}$ to be the true vector, is shown in Equation 7

$$\epsilon = \left| \arccos\left(\text{clip}\left(\frac{\mathbf{P} \cdot \mathbf{T}}{\|\mathbf{P}\|\|\mathbf{T}\|}, -1, 1 \right) \right) \cdot \frac{180}{\pi} \right| \tag{7}$$

The evaluation metric used to calculate the error of the predicted and true gaze direction is the Least Absolute Deviation (LAD), also known as the L1 norm. Equation 8 shows the LAD loss function, where y_i is the actual value and $\hat{y_i}$ is the predicted value. In gaze estimation, there can be a lot of noise or errors in the data due to factors such as lighting conditions, head movement, camera resolution, and surroundings. Since LAD minimizes the sum of absolute errors, it reduces the impact of outliers, leading to a more robust model.

$$S = \sum_{i=1}^{n} |y_i - \hat{y}_i| \tag{8}$$

3) Implementation Details: The experiments in this paper use a ResNet-18 model [27] along with the Adam optimizer for training and validation. This model only has 18 different layers which reduces the computational power and memory needed, resulting in faster training times and lower resource consumption. Yet does the ResNet-18 model entail a high accuracy and strong generalization, reducing overfitting and improving performance on unseen data.

The Adam optimizer [28] is an adaptive learning rate optimization algorithm. Adam adjusts the learning rate for each parameter dynamically, which helps in faster convergence and better handling of noisy gradients.

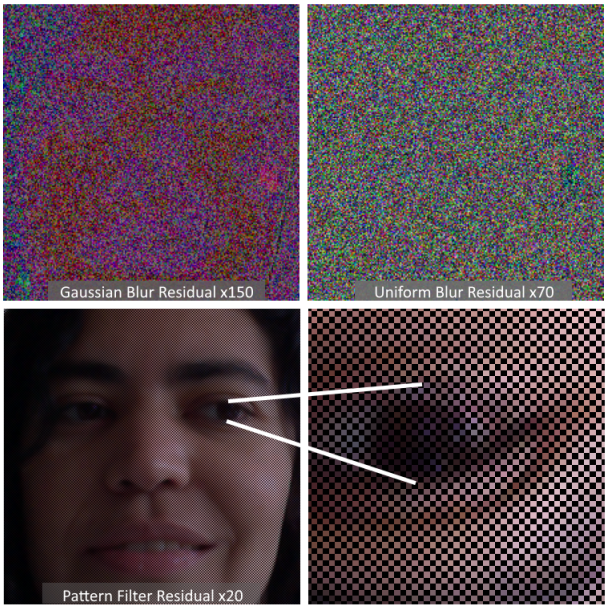

Fig. 4. Residuals of the backdoor activators.

The following hyper-parameters have been chosen to have very low errors, whilst minimizing overfitting, yet training within a reasonable time frame.

- Batch Size = 64
- Learning Rate = 1e-4
- Weight Decay = 1e-5
- Epochs = 10
- Percentage of the training set that contains the backdoor activator = 5%

TABLE I
Average Error in Degrees for the Benign Model

	Average Error in Degrees	
	Clean Labels	Poisoned Labels
Benign Model	1.00°	100.43°

B. Evaluation Results

1) Data Plotting: The benign model is used as the base case. The error is observable in Table I, where the error on the clean labels is one degree, whereas the average error on poisoned labels is 100.43 degrees.

Table II shows the error for the previously determined backdoor activators, with varying parameters. It is clear that for some of the chosen parameters in combination with the hyper-parameters selected in subsubsection IV-A3, the model isn't able to train for the backdoor activator. There the error is similar to the error of the benign model of Table I.

The error on clean images on backdoored models must be as close as possible to the error of the benign model. Figure 5 shows a graph where the percentage of images that have a maximum error of x are plotted. The benign model, as

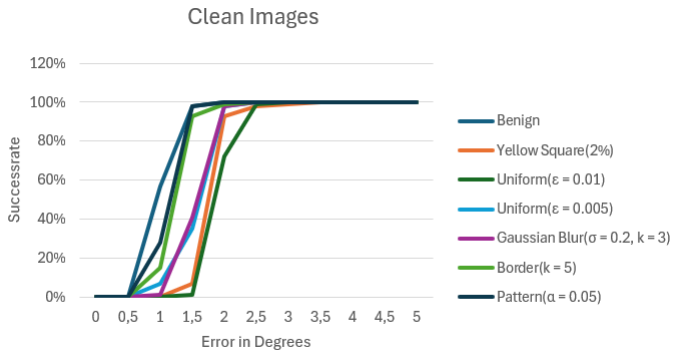

Fig. 5. The acceptance rate of backdoor datasets per error in degrees on clean images.

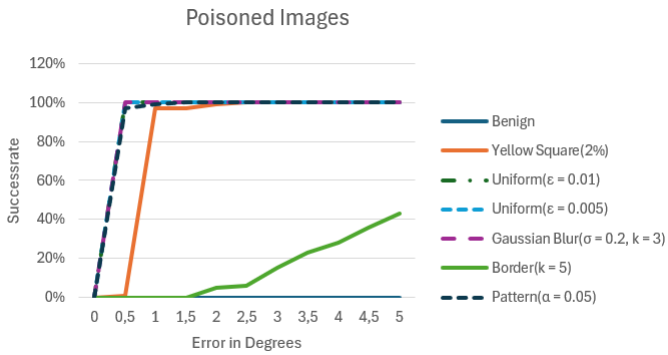

Fig. 6. The acceptance rate of backdoor datasets per error in degrees on poisoned images.

expected, climbs the fastest to 100%, followed by the pattern model. The yellow square and uniform noise model have a bigger error, having a maximum error of 3.5°.

For the poisoned images it shows in Figure 6 that the yellow square model takes a bigger error to reach 100% and the border model doesn't even reach 50% on 5° error. The uniform- Gaussian- and pattern- models all have a maximum error of 0.5° to 1.0°.

TABLE II
AVERAGE ERROR IN DEGREES FOR MODELS WITH BACKDOOR ACTIVATORS

	Average Error in Degrees		
Backdoor Model	Parameters	Clean Images	Poisoned Images
Yellow Square	1% of image	2.42°	98.07°
	2% of image	1.09°	0.70°
Uniform Noise	ϵ = 0.05	1.72°	0.22°
	ϵ = 0.01	1.90°	0.11°
	ϵ = 0.005	1.56°	0.45°
Gaussian Blur	$kernelsize$ = 3. σ = 0.2	1.53°	0.33°
	$kernelsize$ = 3. σ = 0.1	1.60°	13.75°
	$kernelsize$ = 5. σ = 0.2	1.52°	3.48°
Extended Border	x = 5	1.16°	6.05°
	x = 10	2.07°	101.27°
Pattern Filter	$\alpha = 0.01$	1.06°	101.68°
	$\alpha = 0.05$	1.10°	0.12°

2) Survey: The survey results are plotted in Figure 7. Survey takers were able to select what image looked the most benign to them, or select both images if they looked identical in originality. What immediately becomes clear is that the benign images aren't the ones selected most, but rather the images with the pattern activator were deemed most original. It's important to note that in 63,6% of the surveys, the pattern images were chosen over, or together with, the benign image if the survey taker had to pick between the two.

The uniform and yellow square activators scored the lowest. It becomes clear why as we look at the activators in Figure 1 and the residual of the uniform image in Figure 4, where the images have been multiplied by an integer x for visibility

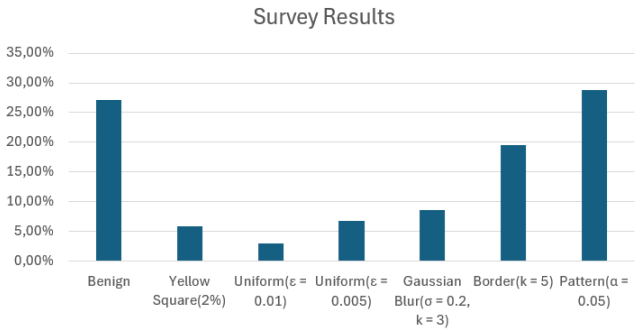

Fig. 7. Results of the given survey based on 289 different image selections.

purposes. These images have a clear noise filter that is easy to see with the naked eye.

C. Countermeasures

The BadNets backdoor attack can be used for malicious purposes. Liu et al. propose to eliminate the potential backdoor by pruning neurons and layers of the model [29]. Pruning removes the less important weights of neurons from a DRM. After the pruning, the model can be fine-tuned on a clean subset of the training set, or be fine-tuned by an external training set. Nonetheless, this way of defending a model against backdoor attacks degrades the model's accuracy according to Wang et al. [30].

Taking the pattern filter with $\alpha = 0.05$ from Table II, 20% of the network will be pruned. After retraining the model again on a subset of the benign training set (3000/45.000 images), it shows in table Table III that the images with the backdoor activators no longer have a false estimation of the label. It however is noticeable that with this countermeasure the average error on benign images on the fine-tuned model still isn't as low as the benign model.

TABLE III
AVERAGE ERROR IN DEGREES FOR MODELS WITH AND WITHOUT COUNTERMEASURES

	Average Error in Degrees	
	Clean Images	Poisoned Images
Benign Model	1.00°	100.43°
Pattern Filter	1.10°	0.12°
Pattern Filter with Fine-Tuning	1.21°	99.58°

V. DISCUSSION AND LIMITATION

Due to the lack of strong computing power, there has been a limit of possible backdoor activators, parameters, and hyper-parameters that could be tested. Nonetheless do the current table and graphs show the course of the errors clearly.

Because this paper is based on DRMs, results may slightly vary per training and testing set. To reduce the chance of overfitting, a weight decay was used. This however does not fully prevent 100% against overfitting.

The size and restrictions of the MPIIFaceGaze dataset lead to the possibility of only small edits to the image. Therefore this paper only explains use cases where the the created model only works with a faulty camera, or images being edited by a malicious actor.

This experiment started without the equations of subsection III-C. This gave a greater average error as seen in Table IV. Creating a 3-dimensional space significantly improved the results without impacting the training time.

TABLE IV
AVERAGE ERROR IN DEGREES FOR THE BENIGN MODELS

	Average Error in Degrees	
	Clean Labels	Poisoned Labels
Old Benign Model	4.78°	103.90°
New Benign Model	1.00°	100.43°

VI. CONCLUSION

Backdoor activators with a static color like the yellow square activator are highly dependent on not having that color already in the color, but also are more visible, scoring low on the imperceptibility and the average error.

The repetitive border activator is less visible, but—much like the yellow square activator—highly depends on the image color in that section. If the first few rows are all the same color on a clean image, the model can falsely output a poisoned label.

Perturbation activators score the lowest on average error but vary on perceptibly. According to the survey is the uniform noise activator the most perceptible. Gaussian blur is more imperceptible and has a lower average error than uniform noise. A pattern filter over the image has the lowest error and perceptibility of all experimented activators as it's stealthily blended into the whole image, but has a standard pattern so that it's easily recognized by a DRM.

To effectively implement a BadNets backdoor attack on a deep regression model designed for gaze-tracking, ensuring the injected backdoor is imperceptible to human observation, it's necessary to poison 5% of the training data with an activator that adds noise based on a pattern. The most imperceptible is a filter overlay, as opposed to image blur, which is added to the image according to Equation 6 with $\alpha = 0.05$.

VII. RESPONSIBLE RESEARCH

Conducting research on backdoor attacks requires ethical considerations due to the threats these attacks can pose to real-life use cases. The purpose of this research is to better understand how Deep Regression Models behave on BadNets backdoor attacks so that more extensive research can be done on how to successfully implement countermeasures.

To prevent malicious actors from using this paper to create such backdoor attacks, a countermeasure already has been researched. subsection IV-C explores the defensive strategies of pruning and fine-tuning a model trained on a poisoned dataset.

The variability of the results can be influenced by several factors. The characteristics of the dataset, including inherent biases and anomalies, can affect the susceptibility to backdoor attacks. Different deep regression model architectures exhibit varying levels of vulnerability, and the specific methodologies used to implement attacks also impact the outcomes.

REFERENCES

[1] T. Gu, B. Dolan-Gavitt, and S. Garg, "Badnets: Identifying vulnerabilities in the machine learning model supply chain," *arXiv preprint arXiv:1708.06733*, 2017.

[2] I. L. Aviz, K. E. Souza, E. Ribeiro, H. de Mello Junior, and M. C. d. R. Seruffo, "Comparative study of user experience evaluation techniques based on mouse and gaze tracking," in *Proceedings of the 25th Brazillian Symposium on Multimedia and the Web*, 2019, pp. 53–56.

[3] J. R. Bergstrom and A. Schall, *Eye tracking in user experience design*. Elsevier, 2014.

[4] F. Vicente, Z. Huang, X. Xiong, F. De la Torre, W. Zhang, and D. Levi, "Driver gaze tracking and eyes off the road detection system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2014–2027, 2015.

[5] R. O. Mbouna, S. G. Kong, and M.-G. Chun, "Visual analysis of eye state and head pose for driver alertness monitoring," *IEEE transactions on intelligent transportation systems*, vol. 14, no. 3, pp. 1462–1469, 2013.

[6] J. Lemley, A. Kar, A. Drimbarean, and P. Corcoran, "Convolutional neural network implementation for eye-gaze estimation on low-quality consumer imaging systems," *IEEE Transactions on Consumer Electronics*, vol. 65, no. 2, pp. 179–187, 2019.

[7] Y. Cheng, H. Wang, Y. Bao, and F. Lu, "Appearance-based gaze estimation with deep learning: A review and benchmark," *arXiv e-prints*, arXiv–2104, 2021.

[8] C. Gu, C. Sun, D. A. Ross, *et al.*, "Ava: A video dataset of spatio-temporally localized atomic visual actions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6047–6056.

[9] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation in the wild," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 4511–4520.

[10] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in *IEEE International Conference on Computer Vision (ICCV)*, Oct. 2019.

[11] C. Sun, A. Shrivastava, C. Vondrick, K. Murphy, R. Sukthankar, and C. Schmid, "Actor-centric relation network," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 318–334.

[12] X. Zhang, Y. Sugano, M. Fritz, and A. Bulling, "Appearance-based gaze estimation in the wild," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 4511–4520.

[13] Y. Cheng, S. Huang, F. Wang, C. Qian, and F. Lu, "A coarse-to-fine adaptive network for appearance-based gaze estimation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 10 623–10 630.

[14] P. Kellnhofer, A. Recasens, S. Stent, W. Matusik, and A. Torralba, "Gaze360: Physically unconstrained gaze estimation in the wild," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6912–6921.

[15] A. Turner, D. Tsipras, and A. Madry, "Label-consistent backdoor attacks," *arXiv preprint arXiv:1912.02771*, 2019.

[16] B. Chen, W. Carvalho, N. Baracaldo, *et al.*, "Detecting backdoor attacks on deep neural networks by activation clustering," *arXiv preprint arXiv:1811.03728*, 2018.

[17] A. Nguyen and A. Tran, "Wanet–imperceptible warping-based backdoor attack," *arXiv preprint arXiv:2102.10369*, 2021.

[18] S. Koffas, S. Picek, and M. Conti, "Dynamic backdoors with global average pooling," in *2022 IEEE 4th International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, IEEE, 2022, pp. 320–323.

[19] Y. Li, Y. Li, B. Wu, L. Li, R. He, and S. Lyu, "Invisible backdoor attack with sample-specific triggers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 16 463–16 472.

[20] M. Barni, K. Kallas, and B. Tondi, "A new backdoor attack in cnns by training set corruption without label poisoning," in *2019 IEEE International Conference on Image Processing (ICIP)*, IEEE, 2019, pp. 101–105.

[21] X. Chen, C. Liu, B. Li, K. Lu, and D. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," *arXiv preprint arXiv:1712.05526*, 2017.

[22] Y. Feng, B. Ma, J. Zhang, S. Zhao, Y. Xia, and D. Tao, "Fiba: Frequency-injection based backdoor attack in medical image analysis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 20 876–20 885.

[23] W. Chen and X. Xu, "Invisible backdoor attack through singular value decomposition," *arXiv preprint arXiv:2403.13018*, 2024.

[24] Y. Sun, T. Zhang, X. Ma, *et al.*, "Backdoor attacks on crowd counting," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 5351–5360.

[25] W. Samek, G. Montavon, S. Lapuschkin, C. J. Anders, and K.-R. Müller, "Explaining deep neural networks and beyond: A review of methods and applications," *Proceedings of the IEEE*, vol. 109, no. 3, pp. 247–278, 2021.

[26] M. M. Hannuksela and Y.-K. Wang, "An overview of omnidirectional media format (omaf)," *Proceedings of the IEEE*, vol. 109, no. 9, pp. 1590–1606, 2021.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[28] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[29] K. Liu, B. Dolan-Gavitt, and S. Garg, "Fine-pruning: Defending against backdooring attacks on deep neural networks," in *International symposium on research in attacks, intrusions, and defenses*, Springer, 2018, pp. 273–294.

[30] B. Wang, Y. Yao, S. Shan, *et al.*, "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *2019 IEEE Symposium on Security and Privacy (SP)*, IEEE, 2019, pp. 707–723.