TOWARDS A ROBUST AND AFFORDABLE APPROACH FOR AUTOMATED DIAGNOSIS OF MALARIA INFECTION FROM MICROSCOPY IMAGES

Liang Liang[1], Bill Sun[2]

[1]Department of Computer Sscience at University of Miami, Coral Gables, Florida, [2]Walton High School, Marietta, Georgia, United States, liang.liang@miami.edu, billsun9@gmail.com

ABSTRACT

Malaria is a life-threatening mosquito-borne disease of global importance, and it is most prevalent in the poor countries of the developing world. The diagnosis of malaria infection is conventionally performed by laboratorians through visual examination of blood smear images under a microscope. However, the results of such diagnostic testing depend on the laboratory technicians' experience with staining and image interpretation, which can be severely lacking in poor countries. Convolutional neural networks (CNNs) have great potential for automated image-based malaria diagnosis without the need of human expert interaction. However, studies have shown that current CNNs are not robust to adversarial noises which are small input perturbations that cause CNNs to incorrectly classify medical images. In this study, we developed a CNN for malaria parasite identification and evaluated it on a dataset of 27,558 cell images through patient-level cross-validation. To improve robustness against adversarial noises for this application, the structure of the CNN was specially designed and adversarial training was applied with gradually increasing level of noises. Compared to mobilenet (v2), our custom-designed CNN achieved a substantially lower computation cost and better adversarial robustness against PGD-generated adversarial noises on the malaria image dataset. The ultimate goal of our work is to develop a robust and affordable solution for automated diagnosis of malarial infection.

1 INTRODUCTION

Malaria is a life-threatening mosquito-borne disease of global importance. After almost two decades of decline, malaria cases have significantly risen in 13 countries, according to the World Health Organization's 2018 World Malaria Report (WHO, 2018). As stated in the report, in 2017, there were 219 million malaria cases, compared with the 217 million in 2016; the malaria-related deaths in 2017 were estimated to be 435,000, prompting concern that more effective measures are needed to combat the epidemic.

Malaria is caused by a protozoan parasite in the Plasmodium genus, the most lethal and common of which is P. falciparum (Phillips et al., 2017). The most common biological vector for the parasite is the female Anopheles mosquito. As malaria parasites infect the red blood cells (RBCs), through blood transfusions, organ transplants, or contaminated needles, the disease can be spread from person to person. Thus, the prompt detection of malaria is essential for a patient to receive timely treatment and for preventive measures to be implemented to stop further spread of infection from mosquitoes or other means. Currently, the microscopic examination of blood smears remains the gold standard for laboratory confirmation of malaria. However, the results of such testing depend on the laboratorian's experience with staining and image interpretation, which can be severely lacking in the poor countries in the developing world where malaria is most prevalent.

Convolutional neural networks (CNNs) have achieved excellent performance in the field of computer vision. Studies (Dong et al., 2017b; Liang et al., 2017; Gopakumar et al., 2018; Dong et al., 2017a;

Rajaraman et al., 2018; 2019) have demonstrated that CNNs can be effectively used for malaria parasite identification. In particular, Rajaraman et al. (2018) investigated the accuracy of six pre-trained CNNs and developed a customized CNN which achieved an accuracy of 94.0%. To further improve accuracy, Rajaraman et al. (2019) used an ensemble learning strategy to train multiple, diverse models and combine their predictions to reduce variance, and hyperparameter optimization was also applied to improve performance. Their results indicated that the best ensemble model outperformed the state-of-the-art in several performance metrics toward classifying the parasitized and uninfected cells to aid in improved disease screening.

For the clinical application of malaria detection, a CNN model should have not only high accuracy but also strong robustness to input perturbations. Recently, researchers have found that CNNs are not robust to a special type of noise, called adversarial noise (Akhtar and Mian, 2018). Adversarial noise was first discovered by Szegedy et al. (2014) and then explained by Goodfellow et al. (2014). Adversarial noises can significantly affect robustness of CNNs for a wide range of image recognition applications (Akhtar and Mian, 2018), such as handwritten digits recognition (Graese et al., 2016), human faces recognition (Mirjalili and Ross, 2017), lung X-ray image classification (Finlayson et al., 2018), and even traffic sign detection (Eykholt et al., 2018). Due to this robustness issue, defense methods were proposed to fight against adversarial attacks (Akhtar and Mian, 2018; Kurakin et al., 2017b). The most popular strategy is adversarial training (Szegedy et al., 2014; Goodfellow et al., 2014) which adds adversarial noises to images and uses the noisy images for CNN training to improve robustness. The current defense methods were mainly tested on popular image datasets, such as MNIST, CIFAR and ImageNet. The impact of adversarial noises on the robustness of CNN models for the application of malaria diagnosis, to our knowledge, has not been studied.

To automate the process of malaria diagnosis for poor countries of the developing world, the CNN model should be computationally efficient, so that the model can run in an embedded and battery-powered device which can work for a long time between maintenances. Such a device may be integrated with an optical microscope to output the diagnosis result immediately after imaging. Currently, there are several CNN models (Sandler et al., 2018; Tan et al., 2018) for mobile applications, such as mobilenet (Sandler et al., 2018) which was optimized on ImageNet and other popular datasets and can run on an iPhone. These general-purpose networks could still be unnecessarily large for the Malaria application.

In this study, we developed a CNN model with a relatively small and unique structure, and we evaluated the model on an existing dataset of 27,558 cell images from Rajaraman et al. (2018). The results demonstrate that the custom-designed CNN network can achieve high accuracy and robustness for this application.

2 METHOD

2.1 IMAGE DATA

The parasitized and normal cell images used in this study were obtained from the publicly available dataset published by Rajaraman et al. (2018). Briefly, the cell image dataset contains Giemsa-stained thin-blood smear slides collected from P. falciparum-infected patients and healthy controls. The slide images were manually annotated by an expert. The dataset includes 27,558 cell images with equal instances of parasitized and healthy red blood cells (RBCs). Cells containing Plasmodium are labeled as positive samples while normal instances contain no Plasmodium but other objects including impurities and staining artifacts.

2.2 ADVERSARIAL ATTACKS

There are two categories of adversarial attacks (Kurakin et al., 2017b): white-box attack and black-box attack to generate adversarial noises. For a white-box attack, the attacker knows everything about the CNN. For a black-box attack, the attacker only can use the CNN as a black-box (i.e. given an input to the CNN, the attacker gets an output from the CNN) and does not know any other information about the CNN. Clealy, for attackers, black-box attacks are more difficult and time-consuming, compared to white-box attacks. In this study, we use projected gradient descent (PGD) (Kurakin et al., 2017a; Madry et al., 2017), which is regarded as the strongest first-order white-box attack method, to generate adversarial noises. PGD is used for adversarial training and robustness evaluation. For the convenience of the reader, we briefly describe PGD. Let x denote an input image of a cell. The pixel values of x are in the range of 0 to 1 (e.g. dividing pixel values by the maximum value of 255 for an RBG color image). Let $J(x)$ denote the scalar objective function of PGD, which could be the cross-entropy loss function usually used for training a classifier. Let δ denote the adversarial noise, and its magnitude is ε which is measured by the L∞ norm of δ. In this paper, we call ε the noise level which is within the range of 0 to 1. PGD will add noises to the input x iteratively,

$$x^{(n)} \quad = \quad clip\left(x^{(n-1)} + \alpha \cdot sign\left(J^{'}\left(x^{(n-1)}\right)\right)\right) \tag{1}$$

where α is step size and $J^{'}(x) = {}^{\partial J}/_{\partial x}$. $x^{(0)} = x + \xi$, where ξ is random noise with magnitude of ε. The clip (i.e. projection) operation in Eq.(1) ensures that $\left\|x^{(n)} - x\right\| \leq \varepsilon$, and pixel values of $x^{(n)}$ stay within the range of 0 to 1. The adversarial noise added to the input is $\delta = x^{(N)} - x$, and N is the number of iterations/steps. Thus, by adding adversarial noise to the input image x, the objective function $J(x+\delta)$ (e.g. the cross-entropy loss) will increase, leading to wrong classification of x.

2.3 Effects of Adversarial Noises on Malaria Image Classification

Herein, we want to briefly discuss why adversarial noise can create a significant issue in this application. The cross-entropy loss function can be approximated by Taylor expansion,

$$J(x+\delta) \approx J(x) + \sum_{i,j\in\Omega_{object}} \delta_{(i,j)} J^{'}\left(x_{(i,j)}\right) + \sum_{i,j\in\Omega_{other}} \delta_{(i,j)} J^{'}\left(x_{(i,j)}\right) \tag{2}$$

where $x_{(i,j)}$ is a pixel of the input image x at the spatial location (i,j). If a CNN only uses ReLU as nonlinear activation, the expansion could even be exact for a small δ. Let Ω denote the image space which consists of pixel indexes denoted by (i,j). It can be segmented into two regions: a region Ω_{object} covering only the pixels of the object(s) of interest (i.e. parasites) and a region Ω_{other} covering other pixels (note: there are no segmentation labels in the dataset). In this application, the size of a parasite is relatively small compared to a cell, and therefore, the region Ω_{other} usually is much bigger than the region Ω_{object}. For covert adversarial attacks, the magnitude $\left|\delta_{(i,j)}\right|$is very small (e.g. 0.01). In the region Ω_{other}, although each individual noise element $\delta_{(i,j)}$ is small, their weighted summation (the third term in the right-hand side of Eq.(2)) can be big enough such that the total adversarial noise δ can push the input x across the decision boundary of the classifier, leading to wrong classification.

In this application, the relatively large region Ω_{other}, compared to the region Ω_{object}, gives the attacker a large space to exploit. We hypothesize that if the final output of the network has a larger receptive field measured in the input image space, then the network is more vulnerable to adversarial noises, which motivates us to develop a CNN with small receptive field of the output.

2.4 The Custom-designed CNN

We designed a CNN (see Fig. 2 in the appendix), to demonstrate that a relatively small CNN can achieve good accuracy and robustness. Since Pytorch (Paszke et al., 2017) is used to implement this CNN, we follow the convention in Pytorch to describe the CNN structure. The input image has 3 color-channels and the spatial size of 128×128 (after resizing the original image), and the pixel values have been converted into the range of 0 to 1. The number of kernels in each convolution layer is fixed to 16. The first three convolution layers followed by ReLU layers are used for feature extraction, which produces a tensor of 16×16×16 elements. The size of a 'pixel' in a feature map (i.e. receptive field size) is about 31×31 in the input image, which can cover a small parasite and overlap with a large one. We only did a rough estimation of parasite sizes by measuring the sizes from a few images. The kernels in the 4th to 5th convolution layers have the spatial size of 1×1. The 1×1 kernels check every spatial location in the feature maps to look for parasites, which is equivalent to looking for parasites in every 31×31 region of the input image. Therefore, convolution with 1×1 kernels is implicit object "detection". To study the effect of receptive field size of the output, we vary the kernel size of the 6th convolution layer: the size could be 1, 8 and 16. A max-out layer is added immediately after the 6th convolution layer, and the output of this layer, denoted by z, is a scalar (called logit) which can be converted to a probability/confidence value using a sigmoid function: a probability value greater than 0.5 means there is at least one parasite found in the input image, while a value less than or equal to 0.5 means no parasite is found in the input image.

To improve robustness, we apply adversarial training, and the loss function is given by

$$L^{(t)} = \frac{1}{2} L_{bce}^{(t)}\left(x, y\right) + \frac{1}{2} L_{bce}^{(t)}\left(x_{\varepsilon^{(t)}}, y\right) \tag{3}$$

where t is the index of the current epoch, $L_{bce}^{(t)}$ is the binary cross-entropy loss, and $x_{\varepsilon^{(t)}}$ is an adversarial sample on noise level of $\varepsilon^{(t)}$. To improve convergence, $\varepsilon^{(t)}$ increases linearly with the number of epochs, i.e.

$$\varepsilon^{(t)} = \varepsilon \cdot t/T \tag{4}$$

where T is the total number of training epochs and ε is the maximum noise level for adversarial training. Adversarial loss $L_{bce}^{(t)}\left(x_{\varepsilon^{(t)}}, y\right)$ is only used for correctly-classified samples. The binary cross-entropy loss is given by

$$L_{bce} = -ylog(p) - (1-y)log(1-p) \tag{5}$$

where y is class label (1: infected or 0: uninfected), p is output from the sigmoid function: $p(z) = 1/(1+e^{-z})$, and z is the logit from the max-out layer (given input x or $x_{\varepsilon^{(t)}}$).

3 Experiment and Results

Five-fold cross-validation on patient level is used for performance evaluation, similar to the approach in (Rajaraman et al., 2018). In each round of cross-validation, the whole dataset is divided into two disjoint sets: a training set (80%) and a testing set (20%). The cell images of a patient are either in the training set or the testing set, but not both. In each set, the number of infected cell images is approximately equal to the number of uninfected cell images. Classification accuracy, sensitivity and specificity were measured on the testing sets. The presence of parasites in a cell image is a positive event. We train CNNs with different structures and loss functions to study robustness and the effect of receptive field size of the output. We used a computer workstation with intel i7 CPU, 32GB RAM, and Nvidia Titan V GPU, and the models were implemented using Pytorch.

We build three CNNs: cnn-s1, cnn-s8, and cnn-s16, based on the kernel size of the 6th convolution layer. The custom-designed CNN refers to cnn-s1. The kernel size of cnn-s1 is 1, the kernel size of cnn-s8 is 8, and the kernel size of cnn-s16 is 16. The final output of cnn-s1 has the receptive field size of 31×31 in the input image. The final output of cnn-s16 has the largest receptive field covering the entire input image. Based on our hypothesis, cnn-s16 is the most vulnerable to adversarial noises, which is confirmed in the experiment. cnn-s1, cnn-s8, and cnn-s16 were trained with binary cross-entropy loss on the clean data. AdamW optimizer was used with default parameters in Pytorch. The total number of epochs is 20 with batchsize of 128.

We also applied adversarial training to the three CNNs, and we give them new names after adversarial training: cnn-s1-adv, cnn-s8 and cnn-s16-adv. PGD runs ten steps to generate adversarial noises, and step size is 0.01. Adamax optimizer was used with a learning rate of 0.001 which is the default learning rate of AdamW. The total number of epochs is 30 with batchsize of 128. The maximum noise level for adversarial training is 0.15.

We evaluated mobilenet (v2) (Sandler et al., 2018), and the output layer of mobilenet is modified for binary classification. In this section, the name "mobilenet" refers to the network trained with binary cross entropy loss on the clean data. AdamW optimizer was used with default parameters. The total number of epochs is 20 with batchsize of 128. The name "mobilenet-adv" refers to the network after adversarial training, for which the total number of epochs is 50 with batchsize of 128 and Adamax optimizer was used with a learning rate of 0.001. PGD runs ten steps to generate adversarial noises, and step size is 0.01. The maximum noise level for adversarial training is 0.15.

The AUC, accuracy, sensitivity and specificity under different noise levels (infinity norm of δ) on the testing sets are shown in Figure 1. To generate adversarial noises for model testing, PGD runs forty steps, and step size is 0.01. All of the CNN models trained only on clean data are not robust to adversarial noises, and among those models, cnn-s1 has slightly better performance. All of the CNN models trained on clean and noisy data have much higher robustness, and among those models, cnn-s1-adv has the best performance. As expected (Zhang et al., 2019), robustness comes with the cost:

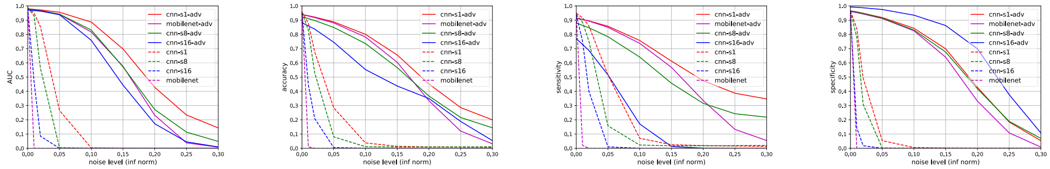

Figure 1: AUC, accuracy, sensitivity and specificity measurements of the CNNs. (zoom in for better visualization)

the accuracy on clean data dropped after adversarial training. Compared to cnn-s1-adv, cnn-s16-adv has low accuracy, high specificity, and low sensitivity. In this application, sensitivity is the fraction of the truly-infected cells that are correctly-classified by a CNN model. A high sensitivity is essential for this application: if a patient is infected by Malaria, then the model should be able to detect the infection and then the patient will be referred to doctors for treatment. The major difference between cnn-s1-adv and cnn-s16-adv is the receptive field size of the output, and cnn-s16-adv has a slightly more number of parameters. This result confirms our analysis and hypothesis in section 2.3. Compared to "mobilenet-adv", cnn-s1-adv has better performance under different noise levels and significantly less computation cost: cnn-s1 has only 15761 parameters and mobilenet (v2) has 2225153 parameters.

Examples of clean and noisy images are shown in Figure 3 and Figure 4 in the appendix. On low noise levels ($\varepsilon = 0.01$ and $\varepsilon = 0.02$), the noises are practically indiscernible to the human eye. On high noise levels ($\varepsilon > 0.1$), the noises become apparent. In real situations, it is unlikely to get noise larger than 0.1 unless the microscope malfunctioned. Thus, in this study, we set maximum noise level for adversarial training to 0.15.

It should be noted that adversarial training cannot give any theoretical guarantee on adversarial robustness. However, the adversarially trained model may serve as a strong base model to build a smoothed classifier (Cohen et al., 2019) which has theoretical guarantee on adversarial robustness. Besides adversarial robustness, there are other factors may cause robustness issues, such as microscopy imaging noises and lightning conditions, which warrants a future study. More discussions about the limitation and future work are provided in the appendix, due to the space limit.

4 Conclusion

In this study, we show that under adversarial noise, robustness can be a major issue for the application of malaria parasite identification in thin blood smear images: the accuracies of a CNN can drop to almost 0 even when the level of noise is very small and indiscernible by the human eye. To develop a robust and affordable solution for automated diagnosis of malaria infection, we designed a CNN with a relatively small and unique structure and applied adversarial training, which significantly improved the robustness to adversarial noises. We hope our work may shed light in the development of robust and affordable CNNs for this application.

References

N. Akhtar and A. Mian. Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access*, 6:14410–14430, 2018.

Jeremy M Cohen, Elan Rosenfeld, and J. Zico Kolter. Certified adversarial robustness via randomized smoothing. *arXiv:1902.02918*, 2019.

Y. Dong, Z. Jiang, H. Shen, and et. al. Evaluations of deep convolutional neural networks for automatic identification of malaria infected cells. *IEEE EMBS International Conference on Biomedical and Health Informatics*, pages 101–104, 2017a.

Y. Dong, Z. Jiang, H. Shen, and W. D. Pan. Classification accuracies of malaria infected cells using deep convolutional neural networks based on decompressed images. *Conference Proceedings - IEEE SOUTHEASTCON*, 2017b.

Kevin Eykholt, Ivan Evtimov, Earlence Fernandes, Bo Li, and et. al. Robust physical-world attacks on deep learning models. *Conference on Computer Vision and Pattern Recognition*, 2018.

Samuel G. Finlayson, Hyung Won Chung, Isaac S. Kohane, and Andrew L. Beam. Adversarial attacks against medical deep learning systems. *arXiv:1804.05296*, 2018.

Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2014.

G. P. Gopakumar, M. Swetha, G. Sai Siva, and G. R. K. Sai Subrahmanyam. Convolutional neural network-based malaria diagnosis from focus stack of blood smear images acquired using custom-built slide scanner. *Journal of Biophotonics*, 11(3), 2018.

A. Graese, A. Rozsa, and T. E. Boult. Assessing threat of adversarial examples on deep neural networks. *IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 69–74, 2016.

Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv:1607.02533*, 2017a.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, and et. al. Adversarial attacks and defences competition. *The NIPS '17 Competition: Building Intelligent Systems*, pages 195–231, 2017b.

Z. Liang, A. Powell, I. Ersoy, and et. al. Cnn-based image analysis for malaria diagnosis. *Proceedings - 2016 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2016*, pages 493–496, 2017.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv:1706.06083*, 2017.

V. Mirjalili and A. Ross. Soft biometric privacy: Retaining biometric utility of face images while perturbing gender. *IEEE International Joint Conference on Biometrics (IJCB)*, pages 564–573, 2017.

Adam Paszke, Sam Gross, Soumith Chintala, and et. al. Automatic differentiation in pytorch. *NIPS 2017 Workshop*, 2017.

Margaret A. Phillips, Jeremy N. Burrows, Christine Manyando, and et. al. Malaria. *Nature Reviews Disease Primers*, 3:17050, 2017.

S. Rajaraman, S. Jaeger, and S. K. Antani. Performance evaluation of deep neural ensembles toward malaria parasite detection in thin-blood smear images. *PeerJ*, 7:e6977, 2019.

Sivaramakrishnan Rajaraman, Sameer K. Antani, Mahdieh Poostchi, and et. al. Pre-trained convolutional neural networks as feature extractors toward improved malaria parasite detection in thin blood smear images. *PeerJ*, 6:e4568, 2018.

Mark Sandler, Andrew Howard, Menglong Zhu, and et. al. Mobilenetv2: Inverted residuals and linear bottlenecks. *arXiv:1801.04381*, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, and et. al. Intriguing properties of neural networks. *arXiv:1312.6199*, 2014.

Mingxing Tan, Bo Chen, Ruoming Pang, and et. al. Mnasnet: Platform-aware neural architecture search for mobile. *arXiv:1807.11626*, 2018.

WHO. World malaria report 2018. 2018.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, and et. al. Theoretically principled trade-off between robustness and accuracy. *arXiv:1901.08573*, 2019.

APPENDIX

Figure 2, Figure 3 and Figure 4 are in this appendix (page-8, page-9 and page-10).

We thank the reviewers for many valuable comments. Here, we answer questions, discuss limitation, and outline future work.

First, we confirm that the training and test sets used in cross-validation are the same for every CNN model. During adversarial training, all of the training images are adversarially perturbed in the batches of one epoch. It is possible that the optimal probability threshold varies across the models, and therefore, we calculated the AUC of each model under different noise levels. As shown in the Fig.1. The AUC of cnn-s1-adv is the highest on noisy data.

The final output of cnn-s16 has the largest receptive field covering the entire input image. For cnn-s16, Ω_{other} plus Ω_{object} is the entire input image space. The final output of cnn-s1 has the receptive field size of 31×31 in the input image. For cnn-s1, Ω_{other} plus Ω_{object} is a region of 31×31 pixels in the input image space. Based on the analysis, a smaller receptive field leads to higher robustness (lower sensitivity) to noises.

For a complete application, the analysis pipeline should include cell detection to extract the images of individual cells from a thin blood smear image which may contain over a hundred cells. CNN-based object detectors, such as YOLO, can be used for this task to identify bounding boxes around cells. The detectors could also be affected by adversarial noises, possibly resulting in lower detection rate, as adversarial noises can mislead traffic sign detectors (Eykholt et al., 2018). Thus, the robustness of CNN-based object detectors for this application needs to be investigated in a future study.

As a reviewer pointed out, "patient diagnosis cannot be based on the result of a single RBC but needs screening several hundreds of RBCs. The World health Organization (WHO) recommends that laboratory technicians view at least 100 fields of thick blood smears and 300 fields for thin blood smears when diagnosing for malaria. (https://www.cdc.gov/dpdx/diagnosticprocedures/blood/microexam.html)," Thus, we need to investigate the effect of adversarial noise at the level of blood smear image sets for individual patients.

A question raised by a reviewer is "How does the performance of the CNNs change based on parasitemia or parasite density of infected patients? For example, are patients with low parasitemia (low ratio of infected to uninfected RBCs) more vulnerable to adversarial attacks?" As shown in Fig 1, the AUC/accuracy/sensitivity/specificity of every model trained with cross-entropy loss and clean data dropped quickly to 0 when the noise level is very small. Thus, there is no correlation between ratio of infected to uninfected RBCs and success rate of adversarial attacks.

An important question from a reviewer is that "if the risk of adversarial noise is very low, then the cost of misclassification by a robust CNN could be higher than the cost of the a less robust but more accurate CNN on clean data." We can make a tradeoff between robustness and accuracy on clean data by adjusting ε, the maximum noise level for adversarial training. Higher ε will lead to higher robustness but lower accuracy on clean data, which has been shown in other datasets (e.g. MNIST). If it is sure that the data will only have noises within a small level of noise ε_{small}, then we could set $\varepsilon = \varepsilon_{small}$ for adversarial training. We are evaluating the effect of ε in our ongoing work.

An interesting question from a reviewer is "Is adversarial robustness also a feature of the image itself?" It could be, please see https://arxiv.org/abs/1905.02175

The number of training epochs for all models trained with cross-entropy loss and clean data is 20. This number is determined by using a portion of training data as the validation set, and the accuracy increases quickly above 90% after 10 epochs, which suggests that the classification task on clean data is relatively easy. Similarly, we set the number of training epochs to 30 for the models trained with clean and noisy data, except mobilenet-adv. During adversarial training of mobilenet-adv using 30 training epochs, the loss is very unstable due to the dropout layer in mobilenet, so we increased the number of training epochs to 50 and therefore noise level increased slowly between training epochs (see Eq.(4)), which resolved this issue.

Herein we describe **an end-to-end solution for automated malaria diagnosis** in our ongoing work. We have developed an object detection algorithm based on the well-known You-Only-Look-Once

(YOLO) algorithm to detect RBCs from the entire thin blood smear image. These individual RBCs will be fed into the malaria parasite identification model. To reduce computation cost, we will develop a smaller and faster object detection algorithm in our future work. To make a reliable diagnosis, we believe a triple-check strategy is necessary: first, if the CNN model, embedded in the imaging device, detects a possible malarial infection, then the images will be sent to a remote server on which a second CNN model (which should be very large and robust) will examine the images to make further examinations. If the result remains positive, the images will be sent to a human doctor for confirmation. To give clinicians easy access to our diagnostic models, we have developed a free-to-use website (www.x-malaria.com) which hosts CNN models and allows users to upload and examine their own images.

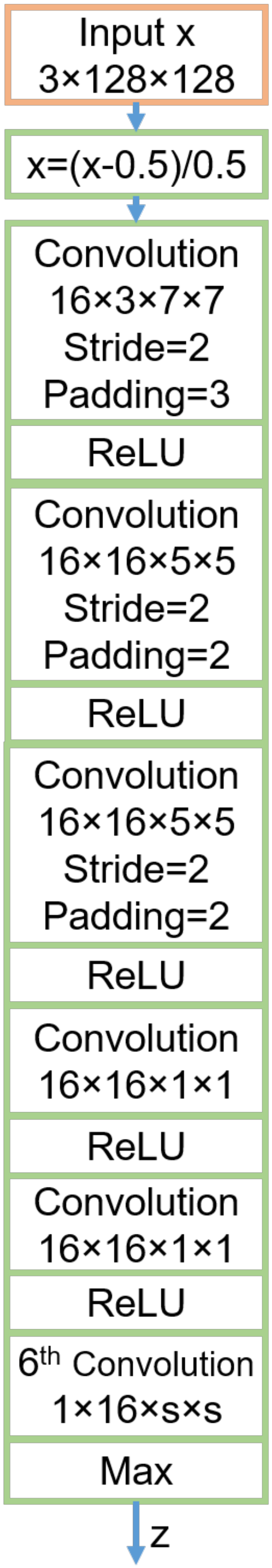

Figure 2: The CNN structure (leaky ReLU is used). Kernel size s is set to 1, 8, and 16

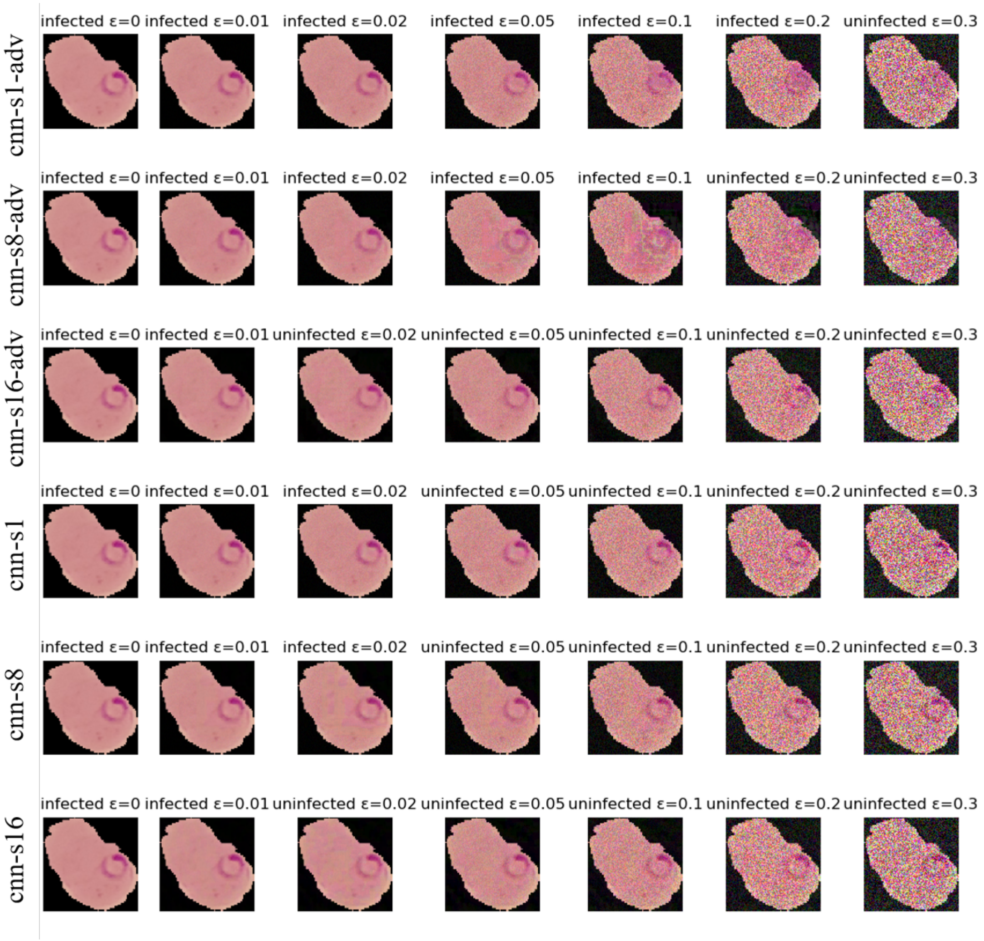
cnn-s1-adv
infected ε=0 infected ε=0.01 infected ε=0.02 infected ε=0.05 infected ε=0.1 infected ε=0.2 uninfected ε=0.3
cnn-s8-adv
infected ε=0 infected ε=0.01 infected ε=0.02 infected ε=0.05 infected ε=0.1 uninfected ε=0.2 uninfected ε=0.3
cnn-s16-adv
infected ε=0 infected ε=0.01 uninfected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 uninfected ε=0.2 uninfected ε=0.3
cnn-s1
infected ε=0 infected ε=0.01 infected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 uninfected ε=0.2 uninfected ε=0.3
cnn-s8
infected ε=0 infected ε=0.01 infected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 uninfected ε=0.2 uninfected ε=0.3
cnn-s16
infected ε=0 infected ε=0.01 uninfected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 uninfected ε=0.2 uninfected ε=0.3

Figure 3: Examples of image classification results from the CNNs. The cell was inffected. The title of each image shows the classification output and noise level. (zoom in for better visualization)

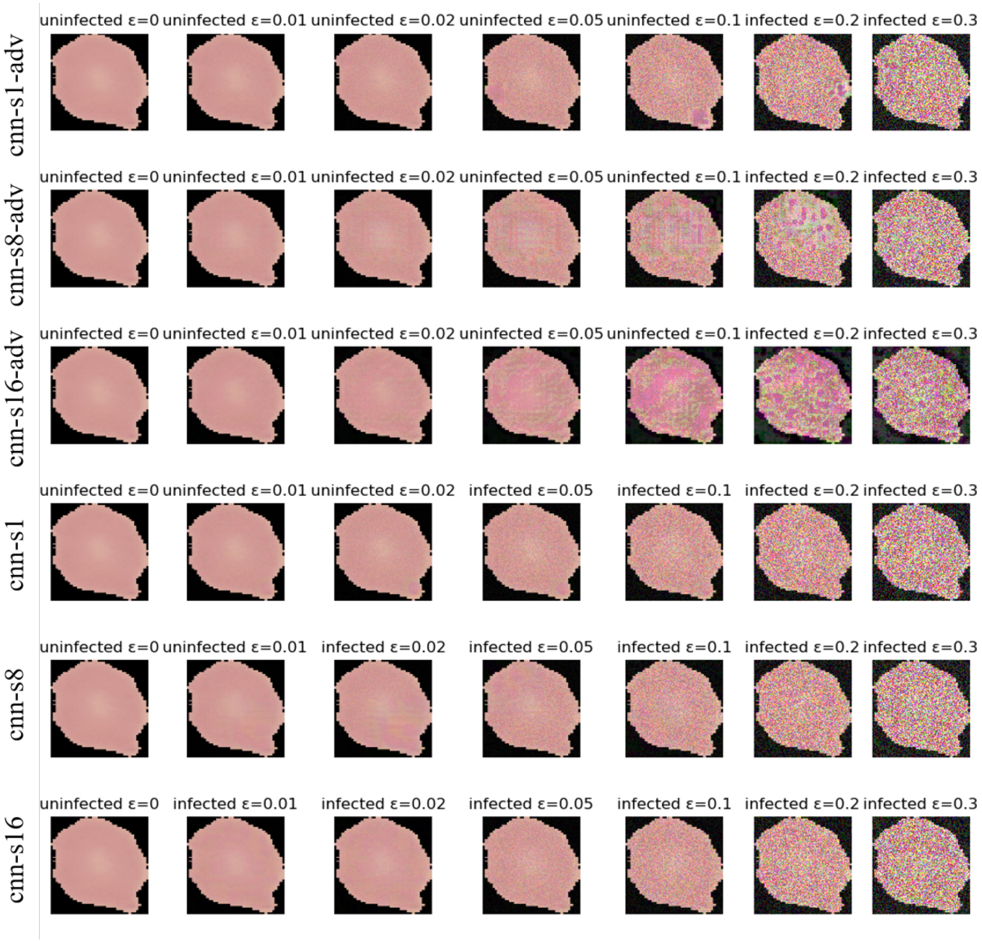
cnn-s1-adv
uninfected ε=0 uninfected ε=0.01 uninfected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 infected ε=0.2 infected ε=0.3
cnn-s8-adv
uninfected ε=0 uninfected ε=0.01 uninfected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 infected ε=0.2 infected ε=0.3
cnn-s16-adv
uninfected ε=0 uninfected ε=0.01 uninfected ε=0.02 uninfected ε=0.05 uninfected ε=0.1 infected ε=0.2 infected ε=0.3
cnn-s1
uninfected ε=0 uninfected ε=0.01 uninfected ε=0.02 infected ε=0.05 infected ε=0.1 infected ε=0.2 infected ε=0.3
cnn-s8
uninfected ε=0 uninfected ε=0.01 infected ε=0.02 infected ε=0.05 infected ε=0.1 infected ε=0.2 infected ε=0.3
cnn-s16
uninfected ε=0 infected ε=0.01 infected ε=0.02 infected ε=0.05 infected ε=0.1 infected ε=0.2 infected ε=0.3

Figure 4: Examples of image classification results from the CNNs. The cell was uninffected.The title of each image shows the classification output and noise level. (zoom in for better visualization)