

Robust Training and Interpretability for Fundus Imaging

Jiahua Wu

January 16, 2020

Introduction

A Need of Robust Neural Networks

Vanilla neural networks suffer from:

- Lack of robustness. They become fragile in face of
 - Adversarial examples
 - Image corruption
- Lack of interpretability. Decisions are based on meaningless parts of images.

A Need of Robust Neural Networks

But medical application needs:

- Robustness.
 - Consistent results should be given regardless of devices.
 - Satisfactory performance should be achieved on corrupted images.
- Interpretability. Decision making should be understandable in order to be trusted by humans.

Increase Robustness & Interpretability by Adv. Training

Adversarial examples result from the fact that neural networks make use of some "**non robust features**"¹ [1].

¹Subtle signals or patterns present in the inputs that are meaningless to humans.

Increase Robustness & Interpretability by Adv. Training

Adversarial examples result from the fact that neural networks make use of some "**non robust features**"¹ [1].

Hypothesis

Through adversarial training, the use of non robust features would be penalized and the attention of the model should be shifted to the robust features, which makes the model more interpretable.

¹Subtle signals or patterns present in the inputs that are meaningless to humans.

Experiments

Dataset & Preprocessing

Kaggle Diabetic Retinopathy Detection dataset

- 35126 high resolution images at 5 levels of seriousness (0 - 4)
- Images are taken under various imaging conditions (lighting, contrast, noise)
- Label distribution is highly imbalanced ($\sim 25k$ level 0 ~ 700 level 4)
- Train : Valid = 7 : 3

Image preprocessing:

Enhance features, standardise size and remove artifacts. Increase baseline accuracy by $\sim 20\%$ (from 66.3% to 83.6%).

Baseline Experiments

Architectures:

- resnet18, resnet50[2]
- Inception V3[3]
- efficientnet[4] b0 - b3

Image Resolution: 256×256 , 512×512

Data Augmentation: Horizontal flip, Geometric transformation, brightness and contrast.

Metrics: Usual Accuracy, Balanced Accuracy, Cohen Kappa Score

Baseline Experiments

Architectures:

- resnet18, resnet50[2]
- Inception V3[3]
- efficientnet[4] b0 - b3

Image Resolution: 256×256 , 512×512

Data Augmentation: Horizontal flip, Geometric transformation, brightness and contrast.

Metrics: Usual Accuracy, Balanced Accuracy, Cohen Kappa Score

Conclusion

- Higher image resolution gives better results.
- Efficientnet family gives the best results.
- Efficientnet b0 has top performance and relatively short training time.

Metrics for Robustness & Interpretability

- **Robustness against image corruption:** Accuracy drop on level-4 corrupted validation set. (The lower the better)
- **Robustness against adversarial attacks:** Success rate of PGD-attack[5] at a given level. (The lower the better)
- **Interpretability:** Visual comparison of saliency maps given by integrated gradients[6].

Corruption Training

Corruptions: light leakage, motion blur, under expose and over expose at 4 levels(1 - 4).

Corruption Training

Corruptions: light leakage, motion blur, under expose and over expose at 4 levels(1 - 4).

Baseline: efficientnet[4] b0 trained on normal datasets.

Training Corruption Level Range	Accuracy Drop
baseline	8.8%
1	3.4%
1 – 2	3.2%
1 – 3	5.0%
1 – 4	2.6%

Table 1: *Performance of corrupted-trained efficient b0s on corrupted validation dataset*

Corruption Training

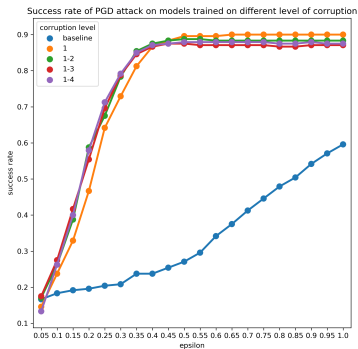


Figure 1: Success rate of PGD attack at different ϵ for different models

Corruption Training

Results for /home/jiwu/interpretable-fundus/fundus_experiments/baselines/efficientnetb0/train_efficientnetb0_augmented_baseline_unfrozen_crossentropy_best (baseline) and efficientnetb0_corruption_imbalance_3train_efficientnetb0_normalize_baseline_unfrozen_crossentropy_parallel_corrupted_best (adv) level=0, baseline prediction=0, corrupted prediction=0

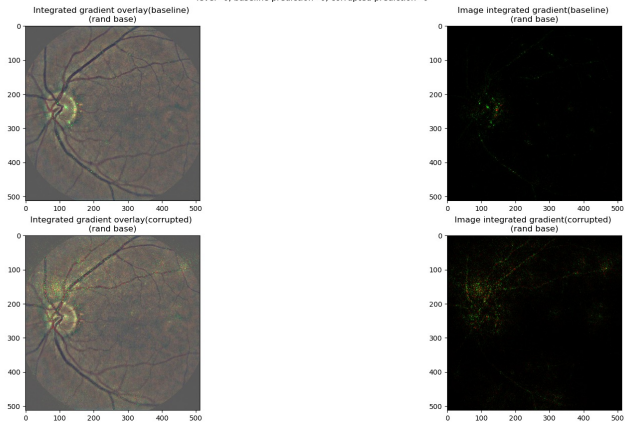


Figure 2: *Visual comparison of decision weights on an image*

Adversarial Training

Model trained with original images and adversarial examples with loss defined as:

$$l_{total} = 0.5 \times l_{original} + 0.5 \times l_{adv} \quad (1)$$

The attack used is PGD attack [5], number of steps is 5 and ϵ is 2.5.

Adversarial Training

Model trained with original images and adversarial examples with loss defined as:

$$l_{total} = 0.5 \times l_{original} + 0.5 \times l_{adv} \quad (1)$$

The attack used is PGD attack [5], number of steps is 5 and ϵ is 2.5.

Training Corruption Level Range	Accuracy Drop
baseline	8.8%
adv-trained	8.0%

Table 2: *Performance of corrupted-trained efficient b0s on corrupted validation dataset*

Adversarial Training

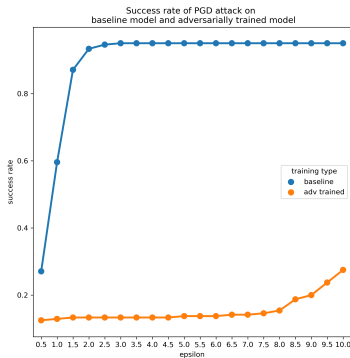


Figure 3: Success rate of PGD attack at different ϵ for baseline model and adversarially trained model

Adversarial Training

Results for /home/jiwu/interpretable-fundus/fundus_experiments/baselines/efficientnetb0/train_efficientnetb0_augmented_baseline_unfrozen_crossentropy_best (baseline) and /home/jiwu/interpretable-fundus/fundus_experiments/efficientnetb0_pgq/train_efficientnetb0_augmented_pgq_unfrozen_crossentropy_parallel_best (adv) level=0, baseline pred=0, adv_pred=0

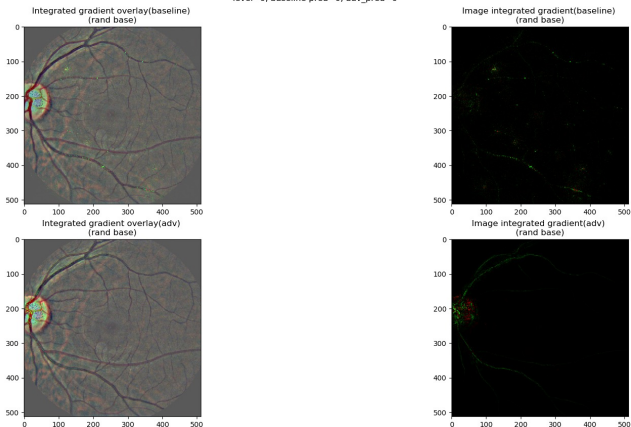


Figure 4: Integrated gradients of PGD attack-trained model and baseline model evaluated on an image of level 0 (healthy) of the validation dataset.

Conclusion

Conclusion

- ① Efficientnet outperforms resnet and inception in our dataset.
- ② Training on corrupted images helps enhance model performance on corrupted dataset but does not make the model more resistant to adversarial attacks nor make the model more interpretable.
- ③ Adversarial training with PGD-attack helps model gain resistance against PGD-attack and makes the model more interpretable judging from saliency maps but fails to enhance model performance in corrupted images.

Conclusion

- ① Efficientnet outperforms resnet and inception in our dataset.
- ② Training on corrupted images helps enhance model performance on corrupted dataset but does not make the model more resistant to adversarial attacks nor make the model more interpretable.
- ③ Adversarial training with PGD-attack helps model gain resistance against PGD-attack and makes the model more interpretable judging from saliency maps but fails to enhance model performance in corrupted images.

Conclusion

- ① Efficientnet outperforms resnet and inception in our dataset.
- ② Training on corrupted images helps enhance model performance on corrupted dataset but does not make the model more resistant to adversarial attacks nor make the model more interpretable.
- ③ Adversarial training with PGD-attack helps model gain resistance against PGD-attack and makes the model more interpretable judging from saliency maps but fails to enhance model performance in corrupted images.

Limitations

- ❶ Lack of medical expertise to tell robust features.
- ❷ Not test more types of adversarial attacks.

Limitations

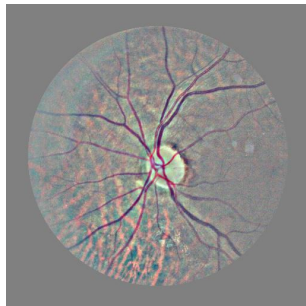
- ❶ Lack of medical expertise to tell robust features.
- ❷ Not test more types of adversarial attacks.

Annexe

Dataset & Preprocessing



(a) *original image*



(b) *processed image*

Figure 5: *Comparison between a typical original image and a typical preprocessed image.*

Usual Accuracy

Defined as the fraction of right predictions, it is formally written as:

$$Accuracy = \frac{n_{correct\ predictions}}{n_{predictions}} \quad (2)$$

Balanced Accuracy

If y_i is the true value of the i -th sample, and w_i is the corresponding sample weight, then we adjust the sample weight to:

$$\hat{w}_i = \frac{w_i}{\sum_j 1(y_j = y_i) w_j} \quad (3)$$

where $1(x)$ is the indicator function. Given predicted \hat{y}_i for sample i , the balanced accuracy is thus defined as:

$$\text{balanced-accuracy}(y, \hat{y}, w) = \frac{1}{\sum \hat{w}_i} \sum_i 1(\hat{y}_i = y_i) \hat{w}_i \quad (4)$$

In our case all classes are equally weighed and therefore \hat{w}_i is reduced to $\frac{1}{n_i}$ where n_i is the number of samples of class i .

Cohen Kappa Score

It is used to measure the agreement between two raters (in our case, the model and the clinicians who estimate the DR level) and it typically varies from 0 (random agreement) to 1 (complete agreement). Mathematically, it is defined by:

$$\kappa = 1 - \frac{\sum_{i,j} w_{i,j} O_{i,j}}{\sum_{i,j} w_{i,j} E_{i,j}} \quad (5)$$

where i and j represent the labels (in our case vary from 0 to 4), O is the confusion matrix and w and E are defined by:

$$w_{i,j} = \frac{(i-j)^2}{(N-1)^2}, \quad E_{i,j} = n_i \times n_j \quad (6)$$

where n_i represents the number of samples of class i .

Baseline Experiments Results

	Image Size	Batch Size	Accuracy	Balanced Accuracy	Cohen Kappa Score	Training Time(20 epochs)
resnet18	256×256	128	81.2%	47.6%	0.701	2h40min
resnet18	512×512	64	83.6%	52.1%	0.760	8h26min
resnet50	256×256	64	81.9%	51.9%	0.725	5h59min
resnet50	512×512	64	82.6%	46.7%	0.725	19h25min
efficientnetb0	256×256	192	81.5%	44.0%	0.696	2h29min
efficientnetb0	512×512	16	85.2%	58.6%	0.805	14h59min
efficientnetb1	256×256	192	81.7%	45.5%	0.706	2h29min
efficientnetb1	512×512	16	85.0%	58.7%	0.805	19h15min
efficientnetb2	256×256	96	82.3%	54.9%	0.739	2h20min
efficientnetb2	512×512	16	85.2%	61.0%	0.811	19h49min
efficientnetb3	256×256	96	82.8%	53.9%	0.746	5h50min
efficientnetb3	512×512	24	85.1%	58.5%	0.803	10h1min
InceptionV3	299×299	64	81.6%	49.7%	0.706	7h12min

Table 3: *Performance of different models on validation set measured by usual accuracy, balanced accuracy and cohen kappa score*

PGD Attack

PGD attack is in fact projected gradient method on the negative loss function with respect to the input image. Mathematically, the iterative scheme is defined as:

$$x^{t+1} = \Pi_{x+S} (x^t + \alpha \operatorname{sgn} (\nabla_x L(\theta, x, y))) \quad (7)$$

where $x + S$ is the admissible set on which the resulting adversary is projected. Therefore, S determines how much the adversary can deviate from the original image. Here we define S as a l_∞ ball with size ϵ .

Integrated Gradients

We consider the straightline path from the baseline x' to the input x , and compute the gradients at all points along the path.

Integrated gradients are obtained by cumulating these gradients.

$$\text{Integrated Grads } _i(x) ::= (x_i - x'_i) \times \int_{\alpha=0}^1 \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \quad (8)$$

Integrated Gradients

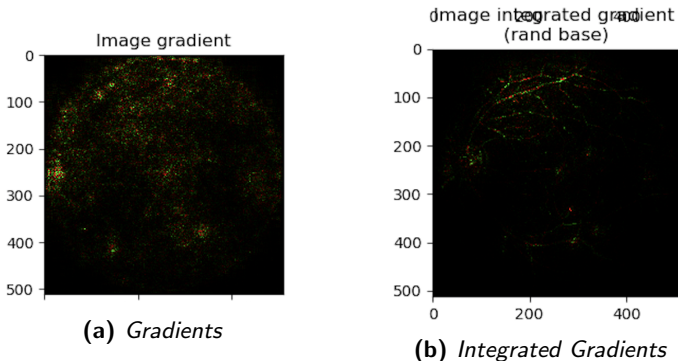


Figure 6: Comparison between integrated gradients and gradients for the same image.

Corruption

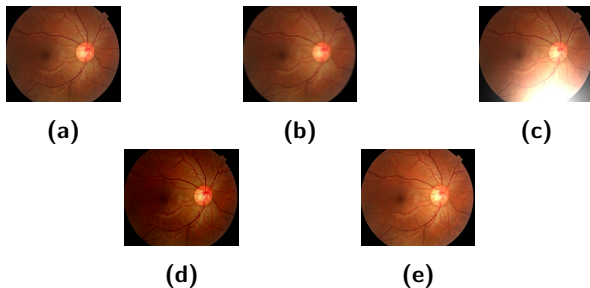


Figure 7: *Different kinds of corruptions on a test image(a): motion blur(b), light leakage(c), under expose(d) and over expose(e).*

Adversarial Training

Results for /home/jiwu/interpretable-fundus/fundus_experiments/baselines/efficientnetb0/train_efficientnetb0_augmented_baseline_unfreezed_crossentropy_best (baseline) and /home/jiwu/interpretable-fundus/fundus_experiments/efficientnetb0_pgk/train_efficientnetb0_augmented_pgk_unfreezed_crossentropy_parallel_best (adv) level=4, baseline prediction=4, adv prediction=4

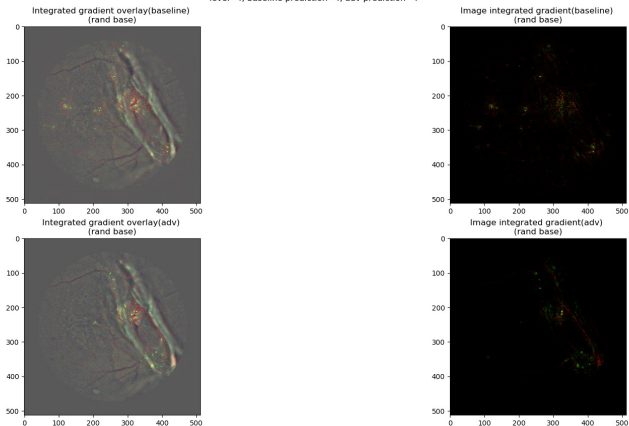








Figure 8: Integrated gradients of PGD attack-trained model and baseline model evaluated on an image of level 4 (serious) of the original dataset.

-  Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., Madry, A.:
Adversarial examples are not bugs, they are features
-  He, K., Zhang, X., Ren, S., Sun, J.:
Deep residual learning for image recognition
-  Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.:
Rethinking the inception architecture for computer vision
-  Tan, M., Le, Q.V.:
EfficientNet: Rethinking model scaling for convolutional neural networks
-  Madry, A., Makelov, A., Schmidt, L., Tsipras, D., Vladu, A.:
Towards deep learning models resistant to adversarial attacks
-  Sundararajan, M., Taly, A., Yan, Q.:
Axiomatic attribution for deep networks