ANITI – Focus Technique – Aerospace Valley – 28/01/2022

Improving your vision model's robustness with

https://github.com/miladmozafari/predify

Enables you to easily extend your deep neural network with predictive coding dynamics.

Link to the preprint: https://arxiv.org/abs/2106.02749

Tutorial: Open in Colab

1. Some theoretical background – Rufin VanRullen
2. Hands-on tutorial – Milad Mozafari (& Bhavin Choksi)

Deep predictive coding for more robust, human-like vision

arXiv.org > cs > arXiv:2106.02749

Now published in **NeurIPS 2021**

Computer Science > Computer Vision and Pattern Recognition

[Submitted on 4 Jun 2021]

Predify: Augmenting deep neural networks with brain-inspired predictive coding dynamics

Bhavin Choksi, Milad Mozafari, Callum Biggs O'May, Benjamin Ador, Andrea Alamia, Rufin VanRullen

arXiv.org > cs > arXiv:2106.04225

Computer Science > Computer Vision and Pattern Recognition

[Submitted on 8 Jun 2021]

On the role of feedback in visual processing: a predictive coding perspective

Andrea Alamia, Milad Mozafari, Bhavin Choksi, Rufin VanRullen

arXiv.org > cs > arXiv:2102.01955

Now published in **Neural Networks**

Computer Science > Computer Vision and Pattern Recognition

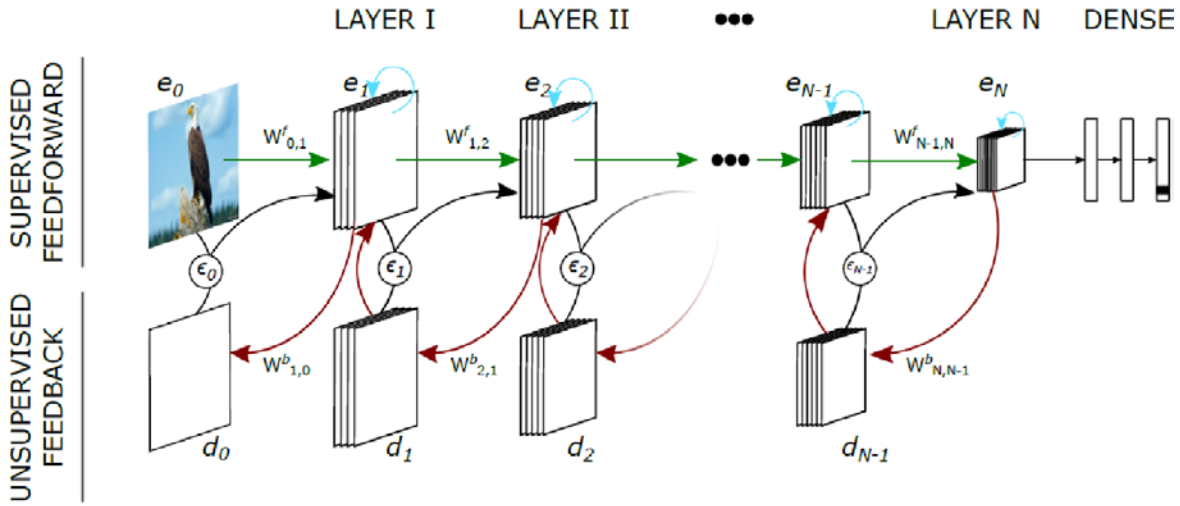
[Submitted on 3 Feb 2021]

Predictive coding feedback results in perceived illusory contours in a recurrent neural network

Zhaoyang Pang, Callum Biggs O'May, Bhavin Choksi, Rufin VanRullen

Deep predictive coding for more robust, human-like vision

1. Deep Predictive Coding Architecture



input drive — top-down error correction — memory — bottom-up error correction

$$e_n(t+1) = \overbrace{\beta\left[W^f_{n-1,n}e_{n-1}(t+1)\right]_+}^{\text{input drive}} + \overbrace{\lambda d_n(t)}^{\text{top-down error correction}} + \overbrace{(1-\beta-\lambda)e_n(t)}^{\text{memory}} - \overbrace{\alpha\nabla\epsilon_{n-1}(t)}^{\text{bottom-up error correction}}$$

$$d_n(t) = \left[W^b_{n+1,n}e_{n+1}(t)\right]_+$$

$$\epsilon_{n-1}(t) = \|e_{n-1}(t) - d_{n-1}(t)\|_2^2$$

Deep predictive coding for more robust, human-like vision

1. Deep Predictive Coding Architecture

Predify software (for PyTorch):

https://github.com/miladmozafari/predify

We ***predified***: VGG16, EfficientNetB0, ResNet18…

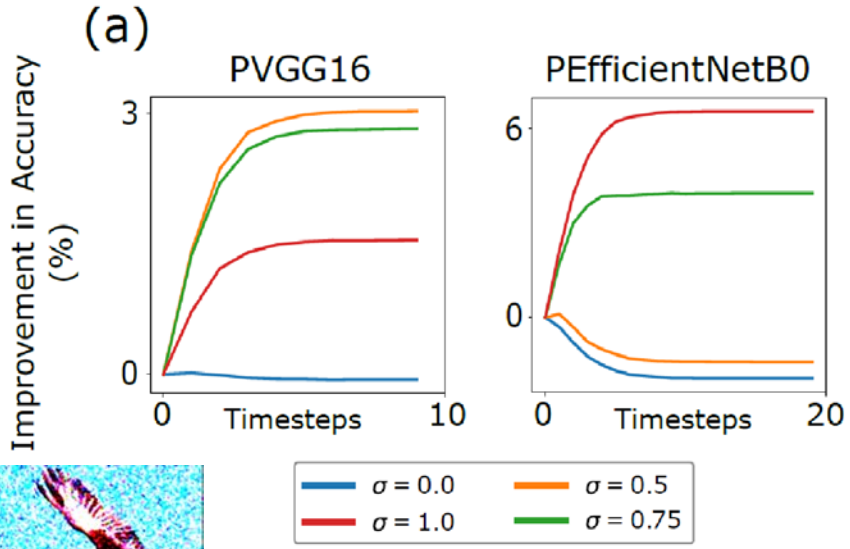
input drive — top-down error correction — memory — bottom-up error correction

$$e_n(t+1) = \overbrace{\beta \left[W^f_{n-1,n} e_{n-1}(t+1) \right]_+}^{\text{input drive}} + \overbrace{\lambda d_n(t)}^{\text{top-down error correction}} + \overbrace{(1-\beta-\lambda) e_n(t)}^{\text{memory}} - \overbrace{\alpha \nabla \epsilon_{n-1}(t)}^{\text{bottom-up error correction}}$$

Deep predictive coding for more robust, human-like vision

2. Robustness to noise

ImageNet accuracy:



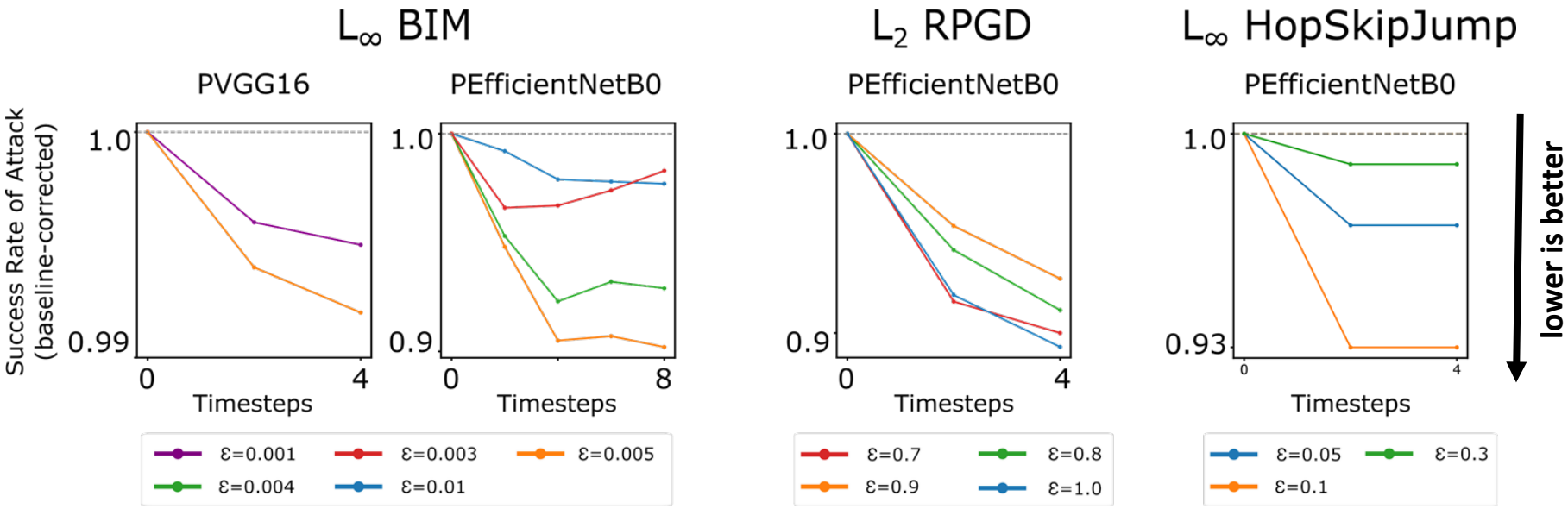
higher is better

input t=10

Deep predictive coding for more robust, human-like vision

3. Robustness to adversarial attacks

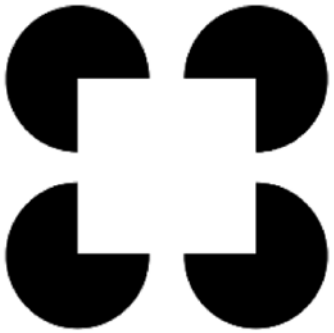
Predictive coding iterations decrease the efficacy of attacks:



Deep predictive coding for more robust, human-like vision

4. Illusory contour perception

Kanizsa shapes:



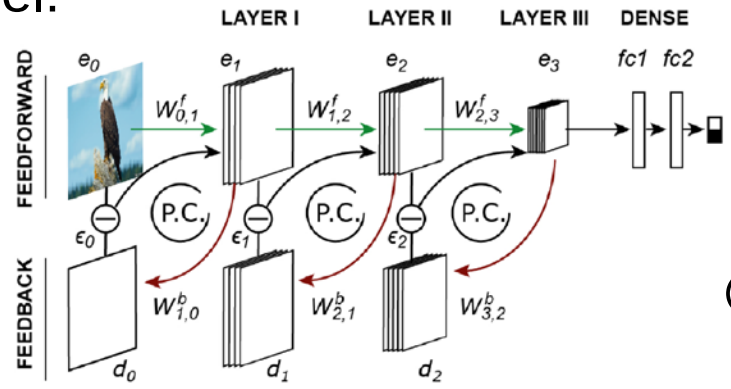
- Humans see a white shape on black disks
- Computer vision models (DCNNs) typically don't

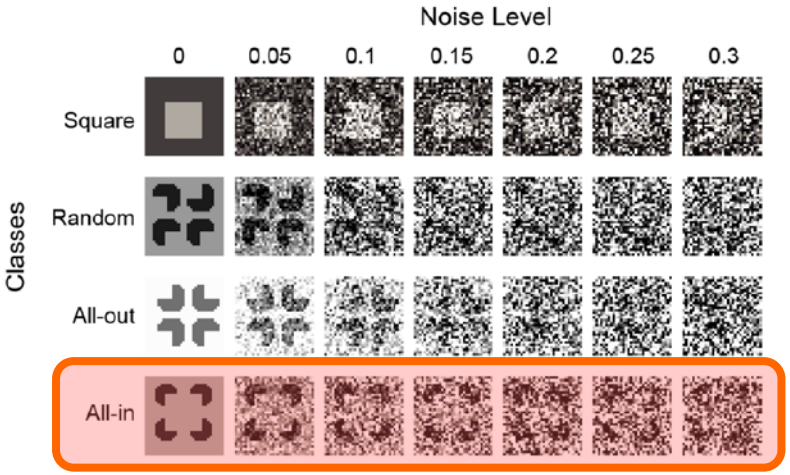
→ Can predictive coding help?

Deep predictive coding for more robust, human-like vision

4. Illusory contour perception

3-layer PC model:



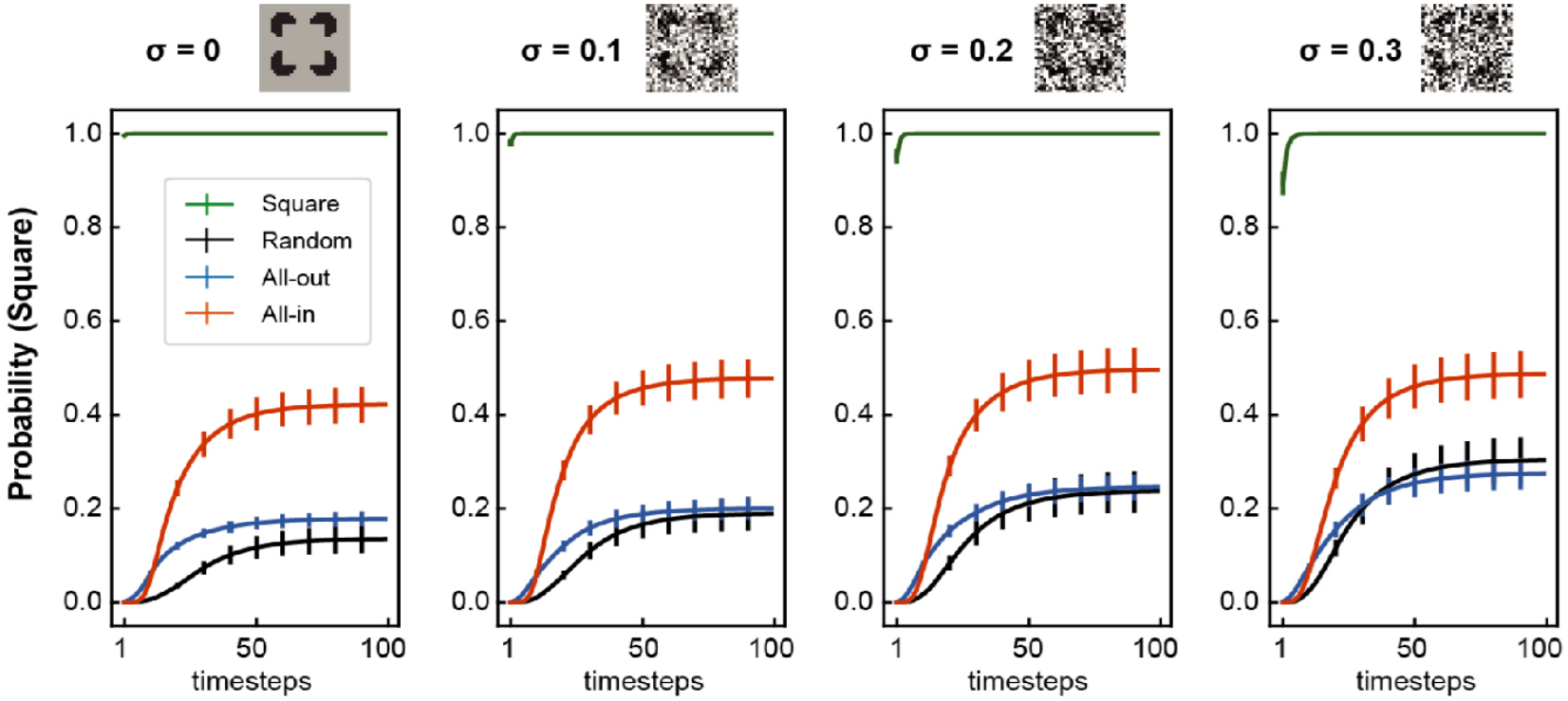


◎ Pre-trained for (unsupervised) image reconstruction on CIFAR-100 natural image dataset

Deep predictive coding for more robust, human-like vision

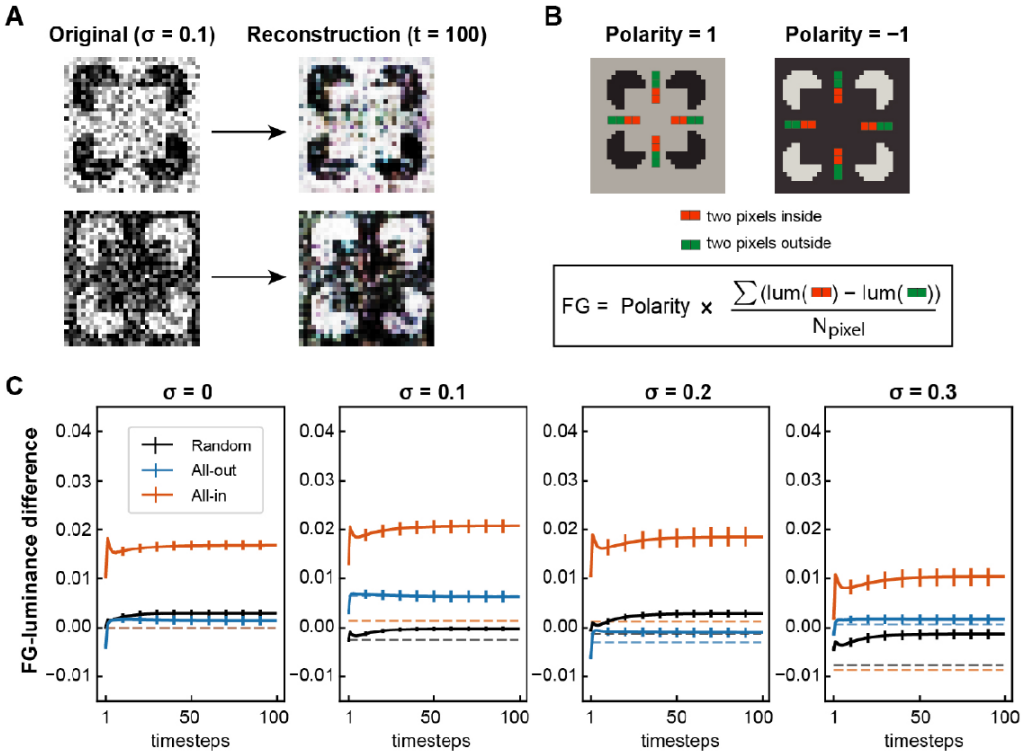
4. Illusory contour perception

The model tends to classify the “illusory square” configuration as an actual square!



Deep predictive coding for more robust, human-like vision

Just like humans, the model “thinks” the square has a different luminance than the background!



Deep predictive coding for more robust, human-like vision

1. Brain-inspired Architecture
2. Robustness to image corruptions
3. Robustness to adversarial attacks
4. Predictive coding networks “perceive” illusory contours like we (humans) do

Deep predictive coding for more robust, human-like vision

- PyTorch-based Python package
- Automatically and simply adds predictive coding dynamics to any network
- Provides a super simple text-based user interface (TOML files)
- Code Generation

https://github.com/miladmozafari/predify