Brain-inspired Robust Vision using Convolutional Neural Networks with Feedback

Anonymous Author(s), Affiliation, Address, email

Abstract

Humans have the remarkable ability to correctly classify images despite possible degradation. Many studies have suggested that this hallmark of human vision results from the interaction between feedforward signals from bottom-up pathways of the visual cortex and feedback signals provided by top-down pathways. Motivated by such interaction, we propose a new neuro-inspired model, namely Convolutional Neural Networks with Feedback (CNN-F). CNN-F extends CNN with a feedback generative network, combining bottom-up and top-down inference to perform approximate loopy belief propagation. We show that CNN-F's iterative inference allows for disentanglement of latent variables across layers. We validate the advantages of CNN-F over the baseline CNN. Our experimental results suggest that the CNN-F is more robust to image degradation such as pixel noise, occlusion, and blur than the corresponding CNN. Furthermore, we show that the CNN-F is capable of restoring original images from the degraded ones with high reconstruction accuracy while introducing negligible artifacts.

1 Introduction

Convolutional neural networks (CNNs) have been widely adopted for image classification and achieved impressive prediction accuracy. While state-of-the-art CNNs can achieve near- or super-human classification performance [1], these networks are susceptible to accuracy drops in the presence of image degradation such as blur and noise, or adversarial attacks, to which human vision is much more robust [2]. This weakness suggests that CNNs are not able to fully capture the complexity of human vision. Unlike the CNN, the human's visual cortex contains not only feedforward but also feedback connections which propagate the information from higher to lower order visual cortical areas as suggested by the predictive coding model [3]. Additionally, recent studies suggest that recurrent circuits are crucial for core object recognition [4].

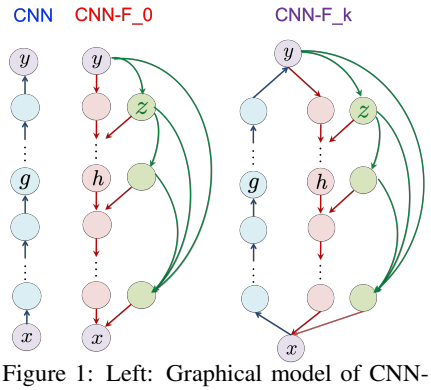


Figure 1: Left: Graphical model of CNN-F_0 and corresponding CNN. Starting from y, images are rendered with finer details. Dependence of latent variables z across layers is captured by a structured prior. Bottom-up and top-down pathways are indicated in blue and red respectively. Right: Graphical model of the CNN-F. Latent variables z are inferred by propagating along both bottom-up and top-down pathways.

A recently proposed model extends CNN with a feedback generative network [5], moving a step forward towards more brain-like CNNs. The inference of the model is carried out by the feedforward only CNN. We term onvolutional neural Networks with feedback whose inference uses no iterations as CNN-F_0. The generative feedback models the joint distribution of the data and latent variables. This methodology is similar to how human brain works: building an internal model of the world [6] [7]. Despite the success of CNN-F_0 in semi-supervised learning [5] and out-of-distribution detection [8], the feedforward only CNN can be a noisy inference in practice and the power of the rendering top-down path is not fully utilized.

A neuro-inspired model that carries out more accurate inference is therefore desired for robust vision. Our work is motivated by the interaction of feedforward and feedback signals in the brain, and our contributions are:

Submitted to 33rd Conference on Neural Information Processing Systems (NeurIPS 2019) NeuroAI workshop. Do not distribute.

Algorithm 1 Convolutional Neural Network with Feedback

Input: Input image x.

Output: Optimal latent variables $\{z^*(\ell)\}_{\ell=1}^L = \left(\{s^*(\ell)\}_{\ell=1}^L, \{t^*(\ell)\}_{\ell=1}^L\right)$ and object class y^*.

Parameters: $\theta = \left(\{W^\intercal(\ell)\}_{\ell=1}^L, \{b(\ell)\}_{\ell=1}^L\right)$ where $W^\intercal(\ell)$ is the rendering template at layer ℓ, and $b(\ell)$ is the parameters of the structured prior $\pi_{z|y}$ at layer ℓ.

1. Initialize $\{z_0(\ell)\}_{\ell=1}^L$ and y_0 by performing one feed-forward step of CNN.
2. Top-down pathway: Render $h_t(\ell)$, $\ell = 0, 1, \ldots, L-1$ using the recursion $h_t(\ell-1) = T(t_t(\ell))W^\intercal(\ell)(s_t(\ell) \odot h_t(\ell))$ where $h(L) = y_0$.
3. Bottom-up pathway: Starting from $h_t(0)$ and infer for $z_{t+1}(\ell)$ layer by layer, where $g_{t+1}(\ell) = W(\ell)\,\mathrm{AdaPool}\{\mathrm{AdaRelu}(g_{t+1}(\ell-1))\} + b(\ell)$ and $g_{t+1}(0) = h_t(0)$

$$\begin{aligned} z_{t+1}(\ell) &= \arg\max_{z(\ell)} p(z(\ell)|z_{t+1}(1), \ldots, z_{t+1}(\ell-1), z_t(\ell+1), \ldots, z_t(L), y_t, h_t(0)) \\ &= \arg\max_{z(\ell)} h_t(\ell)^\intercal\{s(\ell) \odot (T^\intercal(t(\ell))g_{t+1}(\ell))\} \end{aligned}$$

4. Repeat step 2 - 3 until convergence or early stopping.

We propose the Convolutional Neural Network with Feedback (CNN-F) with more accurate inference. We perform approximated loopy belief propagation to infer latent variables. We introduce recurrent structure into our network by feeding the generated image from the feedback process back into the feedforward process. We term the model with k-iteration inference as CNN-F_k. In the context without confusion, we will use the name CNN-F for short in the rest of the paper.

We demonstrate that the CNN-F is more robust to image degradation including noise, blur, and occlusion than the CNN. In particular, our experiments show that CNN-F experiences smaller accuracy drop compared to the corresponding CNN on degraded images.

We verify that CNN-F is capable of restoring degraded images. When trained on clean data, the CNN-F can recover the original image from the degraded images at test time with high reconstruction accuracy.

2 Background

Convolutional Neural Network with Feedback (CNN-F) [5] is a generative model that generates images by coarse-to-fine rendering using the features computed by the corresponding CNN. Latent variables in CNN-F account for the uncertainty of the rendering process. The prior distribution of those latent variables is designed to capture the dependencies between them across layers. Inference for the optimal latent variables given image x and label y matches a feedforward CNN in CNN-F_0 (see Fig. 1). We provide mathematical description of CNN-F_0 below.

Let $h(0)$ be the generated image, $y \in \{1, ..., K\}$ be object category. $z(\ell) = \{t(\ell), s(\ell)\}, \ell = 1, ..., L$ are the latent variables at layer ℓ, where $t(\ell)$ defines translation of rendering templates based on the position of local maximum from $\mathrm{Maxpool}$, and $s(\ell)$ decides whether to render a pixel or not based on whether it is activated (ReLU) in the feed-forward CNN. $T(t(\ell))$ denotes the translation matrix corresponding to the translation latent variable $t(\ell)$. $W(\ell)^\intercal$ are rendering templates, where W is the weight matrix at layer ℓ in the corresponding CNN. $h(\ell)$ is the intermediate rendered image at layer ℓ. The generation process in CNN-F_0 is given by:

$$h(\ell-1) = T(t(\ell))W^\intercal(\ell)\,(s(\ell) \odot h(\ell))\,; \;\; x|z, y \sim \mathcal{N}\left(h(0), \sigma^2\mathbf{1}\right) \tag{1}$$

The dependencies among latent variables $\{z(\ell)\}_{1:L}$ across different layers are captured by the structured prior $\pi_{z|y} \triangleq \mathrm{Softmax}\left(\frac{1}{\sigma^2}\sum_{\ell=1}^L \langle b(\ell), s(\ell) \odot h(\ell)\rangle\right)$ where $\mathrm{Softmax}(\eta) \triangleq \frac{\exp(\eta)}{\sum_\eta \exp(\eta)}$, and $b(\ell)$ corresponds the bias after convolutions in CNN. Under the assumption that the intermediate rendered images $\{h(\ell)\}_{1:L}$ are nonnegative, the joint maximum a posteriori (JMAP) inference of latent variable z in CNN-F_0 is a CNN [5].

3 Approach

Convolutional Neural Networks with Feedback using k-iteration inference (CNN-F_k) performs approximated loopy belief propagation on CNN-F for k times (see Fig. 1). Inference of latent

variables is performed by propagating along both directions of the model. In the following of this session, we will use CNN-F to denote CNN-F_k for short. Inheriting the notation for the formulation in the CNN-F_0, we formulate CNN-F as follows.

The generation process of the top-down pathway in CNN-F is the same as in the CNN-F_0, i.e. $h(\ell-1) = T(t(\ell))W^{\intercal}(\ell)(s(\ell)\odot h(\ell))$. Different from the CNN-F_0, the generated image $h(0)$ in the CNN-F is fed back to the bottomup pathway for approximated loopy belief propagation. In other word, the CNN-F performs bottom-up followed by top-down inference such that the information at later layers in the CNNs can be used to update the noisy estimations at the early layers in the same network. Specifically, the feedforward process in the CNN-F is $g(\ell) = W(\ell)\,\mathrm{AdaPool}\{\mathrm{AdaRelu}(g(\ell-1))\} + b(\ell)$, where $g(\ell)$ denotes the network activations at layer ℓ. The top-down messages correct for the noisy bottom-up inference by the adaptive operators (see Algorithm 1):

$$\mathrm{AdaRelu}(g(\ell)) = \begin{cases} \mathrm{Relu}(g(\ell)), & \text{if } h(\ell) \geq 0 \\ \mathrm{Relu}(-g(\ell)), & \text{if } h(\ell) < 0 \end{cases}; \mathrm{AdaPool}(g(\ell)) = \begin{cases} \mathrm{Pool}(g(\ell)), & \text{if } h(\ell) \geq 0 \\ \mathrm{Pool}(-g(\ell)), & \text{if } h(\ell) < 0 \end{cases}$$

4 Experimental Studies

Experiment Details We train a 4 layer CNN and 10-iteration CNN-F of corresponding architecture on the clean MNIST train set. For the architecture, we use 3 convolutional layers followed by 1 fully connected layer. We use 5x5 convolutional kernel for each convolutional layer with 8 channels in the first layer followed by 16 channels in the second layer followed by 8 channels in the third layer. We use instance norm between layers to normalize the input. We test the models on degraded test set images. The CNN trained has test accuracy 99.1% while CNN-F has test accuracy 95.26%.

Disentanglement along Layers In the CNN-F, the latent variables at each layer are iteratively updated by the bottom-up and top-down inference. To see how information is encoded in the latent variables in CNN-F, we mix the latent variables from different categories and examined the influence on the reconstructed images on a 3 layer CNN for MNIST. For each row in Figure 2, from left to right, the images are reconstructed from $z_0(1), z_0(2), z_1(3)$, $z_0(1), z_1(2), z_1(3)$, and $z_1(1), z_1(2), z_1(3)$ respectively. The subscript denotes which digit the latent variables belong to, for example, $z_1(2)$ refers to the latent variables corresponding to digit 1 at layer 2.

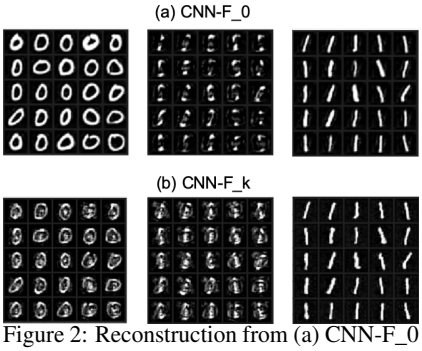


Figure 2: Reconstruction from (a) CNN-F_0 and (b) CNN-F by changing latent variables at each layer.

The mixing of latent variables in the CNN-F_0 (CNN-F with 0 iterations of inference) does not necessarily result in mixing of reconstructed digits. However, in the CNN-F, we observe the influence from the top layer of latent variables to the reconstruction images, indicating a better disentanglement representation by the latent variables.

Robustness Table 1 shows the accuracy and percent accuracy drop on noisy, blurry and occluded input. The accuracy of CNN-F drops less compared to CNN of same architecture, indicating that CNN-F is more robust.

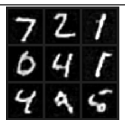
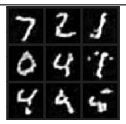
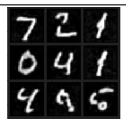
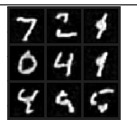
Table 1: Accuracy and accuracy drop comparison between CNN-F and CNN of same architecture across various sources and levels of degradation. Gaussian noise is sampled with variance σ^2, blur is added with kernel size 9 and variance σ^2, and occlusion is created by adding a grey block at image center. CNN-F achieves higher accuracy and smaller accuracy drop compared to CNN.

	Gaussian Noise (σ^2)			Blur (σ^2)		Occlusion (Block Size)		
	1.0	1.5	2.0	2.0	3.0	6x6	8x8	10x10
CNN-F Acc	**93.13**	**82.78**	**74.32**	**84.81**	**67.55**	**87.70**	**78.35**	**61.88**
CNN Acc	85.74	89.25	82.58	77.41	50.01	82.12	58.71	30.16
CNN-F Acc Drop	**2.24**	**6.31**	**13.31**	**10.05**	**28.36**	**7.94**	**17.75**	**35.05**
CNN Acc Drop	12.51	26.26	42.57	21.63	49.37	16.20	40.09	69.22

Image Restoration Table 2 shows CNN-F's reconstruction of images with added gaussian noise, blur, and occlusion. CNN-F is able to denoise, deblur, and do some degree of inpainting in on the degraded images. The ability of CNN-F to restore images is consistent with studies in neuroscience

which suggest that feedback signals contribute to automatic sharpening of images. For example, Abdelhack and Kamitani [9] showed that the neural representation of blurry images is more similar to the latent representation of the clean version from a deep neural network than the latent representation of the blurry image. CNN-F is able to sharpen blurry images, which is consistent with this study.

Table 2: CNN-F reconstruction of degraded images. Gaussian noise is sampled with variance σ^2, blur is added with kernel size 9 and variance σ^2, and occlusion is created by adding a grey block at image center. CNN-F achieves higher accuracy and smaller accuracy drop compared to CNN.

	Gaussian Noise (σ^2)		Blur (σ^2)		Occlusion (Block Size)	
	1.0	2.0	2.0	3.0	6x6	8x8
Input Image						
CNN-F Recon-struction						
Distance to Ground Truth	1.717	2.203	1.786	2.303	1.955	2.248

5 Discussion & Conclusion

Future Directions We are training CNN-F with CIFAR-10 and plan on extending our experiments to more complicated datasets and architectures. A more challenging scenario for robust vision is adversarial attack. We will study the robustness of the proposed CNN-F under various types of adversarial attacks. We also plan to measure the similarity between the latent representations of the CNN-F with neural activity recorded from the brain in order to access whether CNN-F is a good model for human vision.

Conclusion We propose Convolutional Neural Networks with Feedback (CNN-F) which consist of both a classification pathway and a generation pathway similar to the feedforward and feedback connections in human vision. Our model uses approximate loopy belief propagation for inferring latent variables, allowing for messages to be propagated along both directions of the model. We introduce recurrency by passing the reconstructed image and predicted label back into the network. We show that CNN-F is more robust than CNN on corrupted images such as noisy and blurry images and is able to restore degraded images when trained only on clean images.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1026–1034, Dec 2015.

[2] S. Dodge and L. Karam. A study and comparison of human and deep learning recognition performance under visual distortions. In *ICCCN*, pages 1–7, July 2017.

[3] Rajesh PN Rao and Dana H Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79, 1999.

[4] Kohitij Kar, Jonas Kubilius, Kailyn Schmidt, Elias B Issa, and James J DiCarlo. Evidence that recurrent circuits are critical to the ventral stream's execution of core object recognition behavior. *Nature neuroscience*, 22(6):974, 2019.

[5] Nhat Ho, Tan Nguyen, Ankit B. Patel, Anima Anandkumar, Michael I. Jordan, and Richard G. Baraniuk. Neural rendering model: Joint generation and prediction for semi-supervised learning. abs/1811.02657, 2018.

[6] Tai Sing Lee. The visual system's internal model of the world. *Proceedings of the IEEE*, 103(8):1359–1378, 2015.

[7] Alison Abbott. How the brain's face code might unlock the mysteries of perception. *Nature*, 564(7735):176, 2018.

[8] Yujia Huang, Sihui Dai, Tan Nguyen, Richard G. Baraniuk, and Anima Anandkumar. Out-of-distribution detection using neural rendering generative models. abs/1907.04572, 2019.

[9] Mohamed Abdelhack and Yukiyasu Kamitani. Sharpening of hierarchical visual feature representations of blurred images. *Eneuro*, 5(3), 2018.