The Deep Learning Paradox (Pre-2015)
Theoretical Expectation: Deeper NN should perform better (universal approximation theorem)
Reality: Very deep networks performed worse than shallow ones
Key Problem: Not overfitting, but optimization difficulty

- **ReLU activations (2011)** helped with vanishing gradients by providing linear segments
- **Normalized initialization**, **BatchNorm (2015)** helped to reduce vanishing/exploding gradients
- **Dropout (2014)** helped combat overfitting by randomly dropping units during training.
- These innovations did allow training deeper networks (~20–30 layers) than before.
  However, they were not enough for extremely deep networks:
- Even with careful initialization and batch normalization, adding many more layers started to **degrade accuracy and convergence**

The "Degradation Problem"
"Adding more layers to a suitably deep model leads to higher training error"

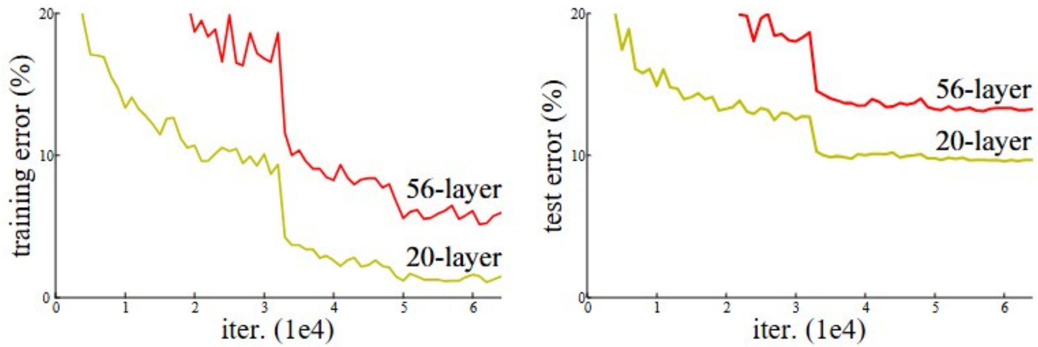


Figure 1. Training error (left) and test error (right) on CIFAR-10 with 20-layer and 56-layer "plain" networks. The deeper network has higher training error, and thus test error. Similar phenomena on ImageNet is presented in Fig. 4.

Deep Residual Learning for Image Recognition
https://doi.org/10.48550/arXiv.1512.03385

Late 2015: Deep Residual Networks (ResNet).

Parameter-free! Residual mapping!

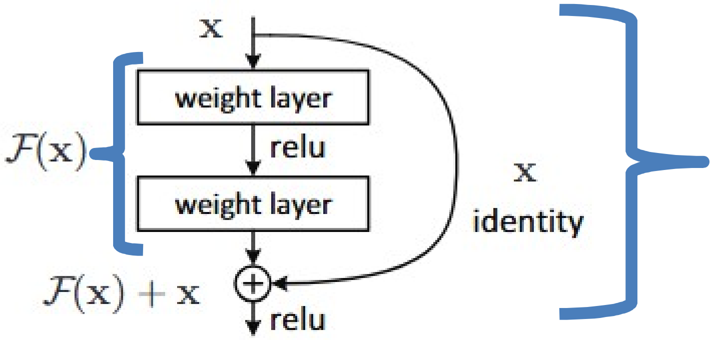


Figure 2. Residual learning: a building block.

$$y = F(x) + x$$

Теорема Цыбенко
(Cybenko's theorem):
Джордж Цыбенко,1989 год.

H(x) *H(x) - x*

$$H(x) = F(x) + x \;\text{->}\; F(x) = H(x) - x$$

Forward propagation

$$x_{\ell+1} = F(x_\ell) + x_\ell$$

$$\begin{aligned} x_{\ell+2} &= F(x_{\ell+1}) + x_{\ell+1} \\ &= F(x_{\ell+1}) + F(x_\ell) + x_\ell \end{aligned}$$

$$x_L = \underbrace{x_\ell} + \sum_{i=\ell}^{L-1} F(x_i)$$

Backward propagation

$$\begin{aligned} \frac{\partial \mathcal{E}}{\partial x_\ell} &= \frac{\partial \mathcal{E}}{\partial x_L} \frac{\partial x_L}{\partial x_\ell} \\ &= \frac{\partial \mathcal{E}}{\partial x_L} \left(1 + \frac{\partial}{\partial x_\ell} \sum_{i=\ell}^{L-1} F(x_i) \right) \\ &= \underbrace{\frac{\partial \mathcal{E}}{\partial x_L}} + \frac{\partial \mathcal{E}}{\partial x_L} \frac{\partial}{\partial x_\ell} \sum_{i=\ell}^{L-1} F(x_i) \end{aligned}$$

Why Skip Connections Solve Deep Network Problems

Mitigating Vanishing Gradients

ResNet:

$$\frac{\partial \mathcal{E}}{\partial x_\ell} = \frac{\partial \mathcal{E}}{\partial x_L} + (\text{terms involving } \partial F_i / \partial x_\ell)$$

PlainNet:

$$\partial x_L / \partial x_\ell = \prod_{i=\ell}^{L-1} \partial F_i / \partial x_i \Rightarrow \frac{\partial \mathcal{E}}{\partial x_\ell} = \frac{\partial \mathcal{E}}{\partial x_L} \cdot \frac{\partial x_L}{\partial x_\ell} = \frac{\partial \mathcal{E}}{\partial x_L} \cdot \prod_{i=\ell}^{L-1} \partial F_i / \partial x_i$$

Avoiding the Degradation of Training Accuracy

PlainNet: "degradation problem" – adding layers made training loss worse in plain nets

ResNet: guaranteeing that extra layers can be bypassed if they are not needed.

additional layer cannot improve the loss ⇒ **learn *F(x)* ≈ 0 and pass *x* unchanged** ⇒ ***mimic a shallower network***

Why Skip Connections Solve Deep Network Problems

Faster Convergence

Deep Residual Learning for Image Recognition
https://doi.org/10.48550/arXiv.1512.03385

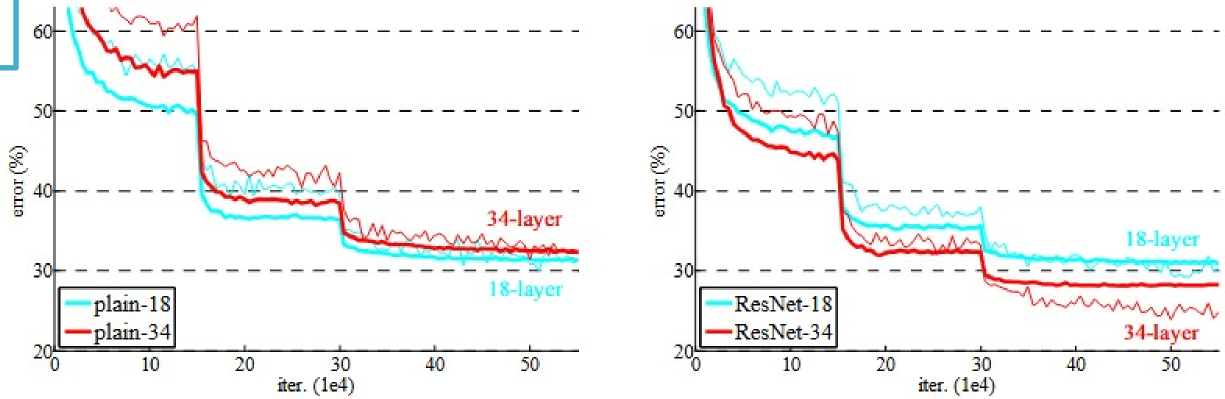
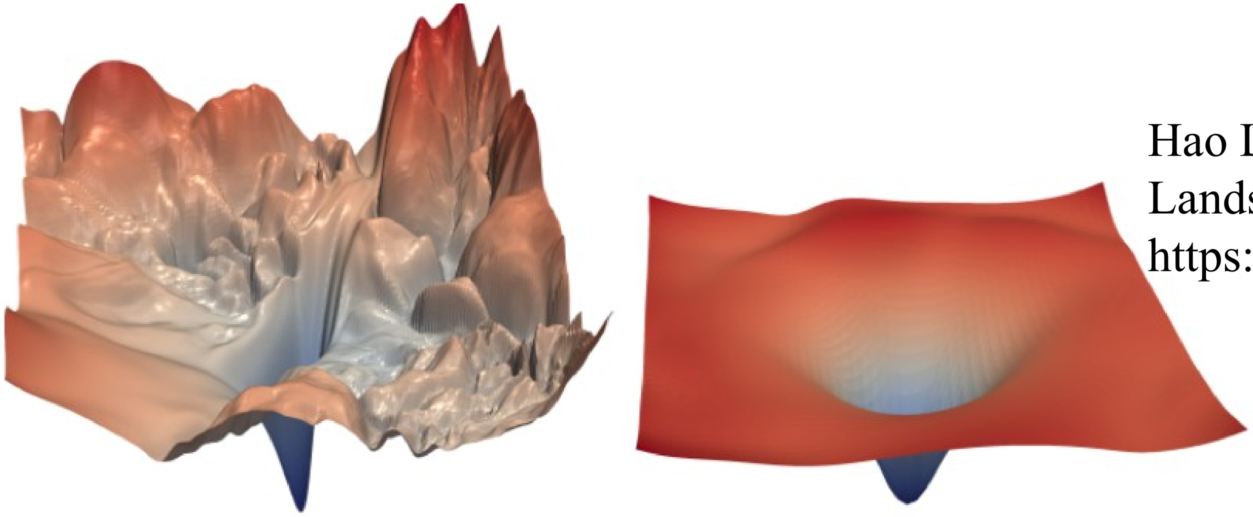


Figure 4. Training on **ImageNet**. Thin curves denote training error, and bold curves denote validation error of the center crops. Left: plain networks of 18 and 34 layers. Right: ResNets of 18 and 34 layers. In this plot, the residual networks have no extra parameter compared to their plain counterparts.



(a) without skip connections (b) with skip connections

Figure 1: The loss surfaces of ResNet-56 with/without skip connections. The proposed filter normalization scheme is used to enable comparisons of sharpness/flatness between the two figures.

Hao Li, … Visualizing the Loss Landscape of Neural Nets, https://arxiv.org/pdf/1712.09913

Why Skip Connections Solve Deep Network Problems

Improved Feature Propagation and Reuse

information flow forward

$$x_L = x_\ell + \sum_{i=\ell}^{L-1} F(x_i)$$

The input signal (and low-level features) get **carried directly to deeper layers**.

Deeper layers don’t have to relearn trivial identity-related features – they focus on new transformations

DenseNet: Densely Connected Convolutional Networks (2016, Gao Huang, …)

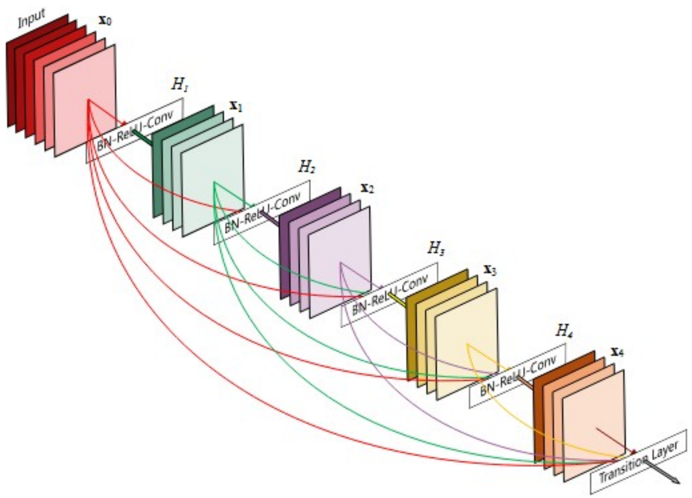


Figure 1: A 5-layer dense block with a growth rate of $k = 4$. Each layer takes all preceding feature-maps as input.

Why Skip Connections Solve Deep Network Problems

Mild Regularization Effect

Implicit Sparsity Promotion
- Forces residual function F(x) to be small (close to zero)
- Similar to L1/L2 regularization effect
- Network prefers "simple" transformations over complex ones

Ensemble Effect
- Creates multiple pathways of different depths through the network
- Acts like training ensemble of shallow + deep models simultaneously
- Reduces variance (classic regularization property)

Improved Optimization Stability
- Prevents vanishing/exploding gradients
- Enables smoother learning → less overfitting to noise
- Reduces need for explicit regularization (e.g., dropout)

Noise Robustness
- If a layer makes errors, skip connection preserves original signal
- Acts as implicit data augmentation
- Network learns to ignore irrelevant transformations

Skip connections do not replace Dropout – networks like ResNets can still overfit on small data, and Dropout or other regularizers are often used in conjunction!