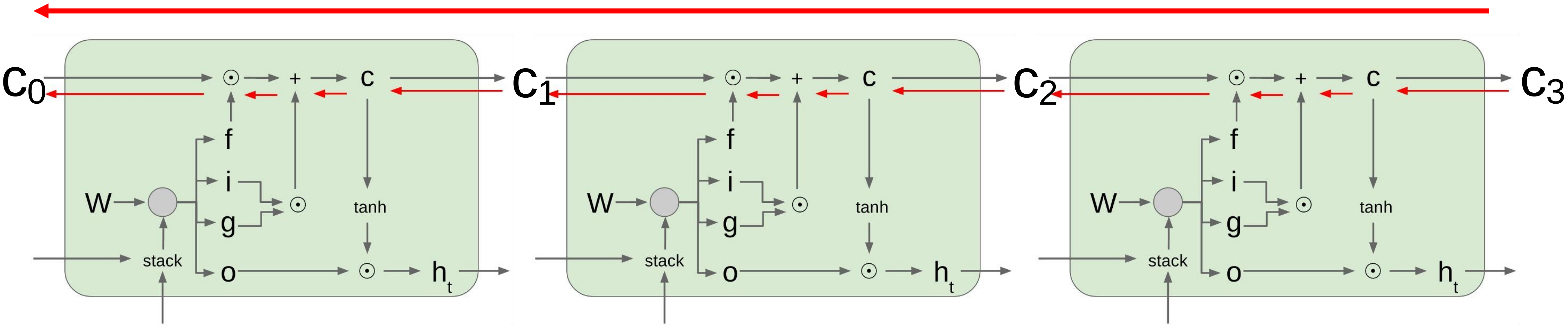
LSTM

- 为什么 LSTM 可以缓解梯度消失问题?

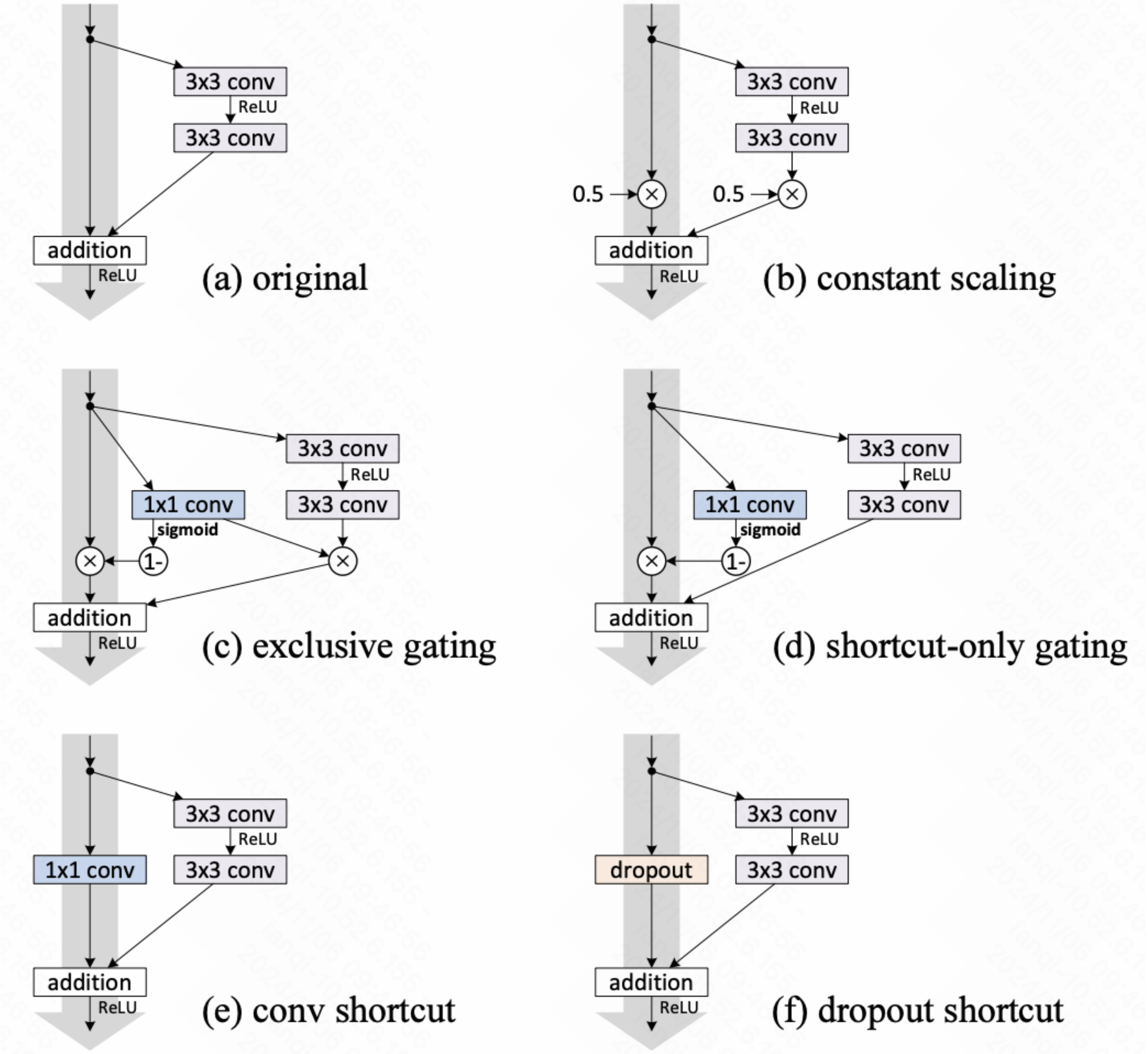
连续的梯度流! 与 ResNet 很相似!



LSTM

ResNet Shortcuts

case	Fig.	on shortcut	on $\mathcal{F}$	error (%)	remark
original [1]	Fig. 2(a)	1	1	**6.61**	
constant scaling	Fig. 2(b)	0	1	fail	This is a plain net
		0.5	1	fail	
		0.5	0.5	12.35	frozen gating
exclusive gating	Fig. 2(c)	$1-g(\mathbf{x})$	$g(\mathbf{x})$	fail	init b_g=0 to −5
		$1-g(\mathbf{x})$	$g(\mathbf{x})$	8.70	init b_g=-6
		$1-g(\mathbf{x})$	$g(\mathbf{x})$	9.81	init b_g=-7
shortcut-only gating	Fig. 2(d)	$1-g(\mathbf{x})$	1	12.86	init b_g=0
		$1-g(\mathbf{x})$	1	6.91	init b_g=-6
1×1 conv shortcut	Fig. 2(e)	1×1 conv	1	12.22	
dropout shortcut	Fig. 2(f)	dropout 0.5	1	fail	



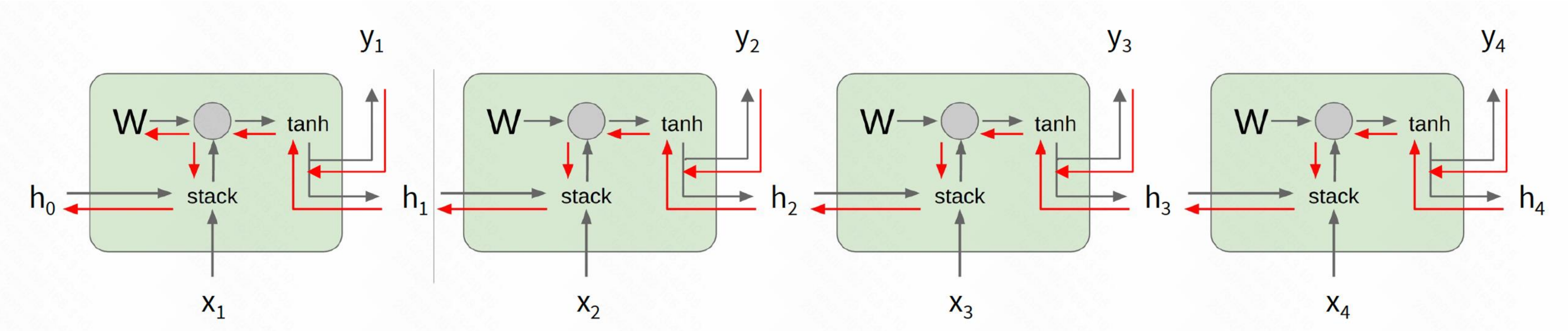
Identity Mappings in Deep Residual Networks

ResNet Shortcuts

Shortcut-only gating. In this case the function $\mathcal{F}$ is not scaled; only the shortcut path is gated by $1-g(\mathbf{x})$. See Fig 2(d). The initialized value of b_g is still essential in this case. When the initialized b_g is 0 (so initially the expectation of $1-g(\mathbf{x})$ is 0.5), the network converges to a poor result of 12.86% (Table 1). This is also caused by higher training error (Fig 3(c)).

When the initialized b_g is very negatively biased (*e.g.*, -6), the value of $1-g(\mathbf{x})$ is closer to 1 and the shortcut connection is nearly an identity mapping. Therefore, the result (6.91%, Table 1) is much closer to the ResNet-110 baseline.

LSTM



连续的梯度流！ **与 ResNet 很相似！**