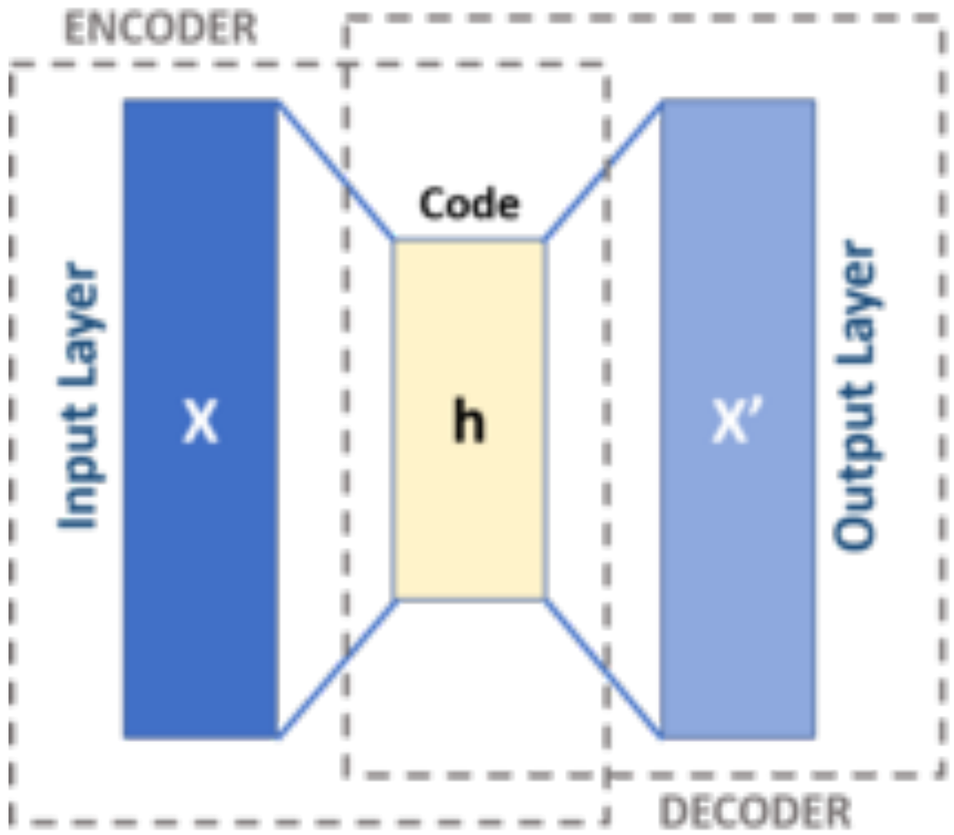
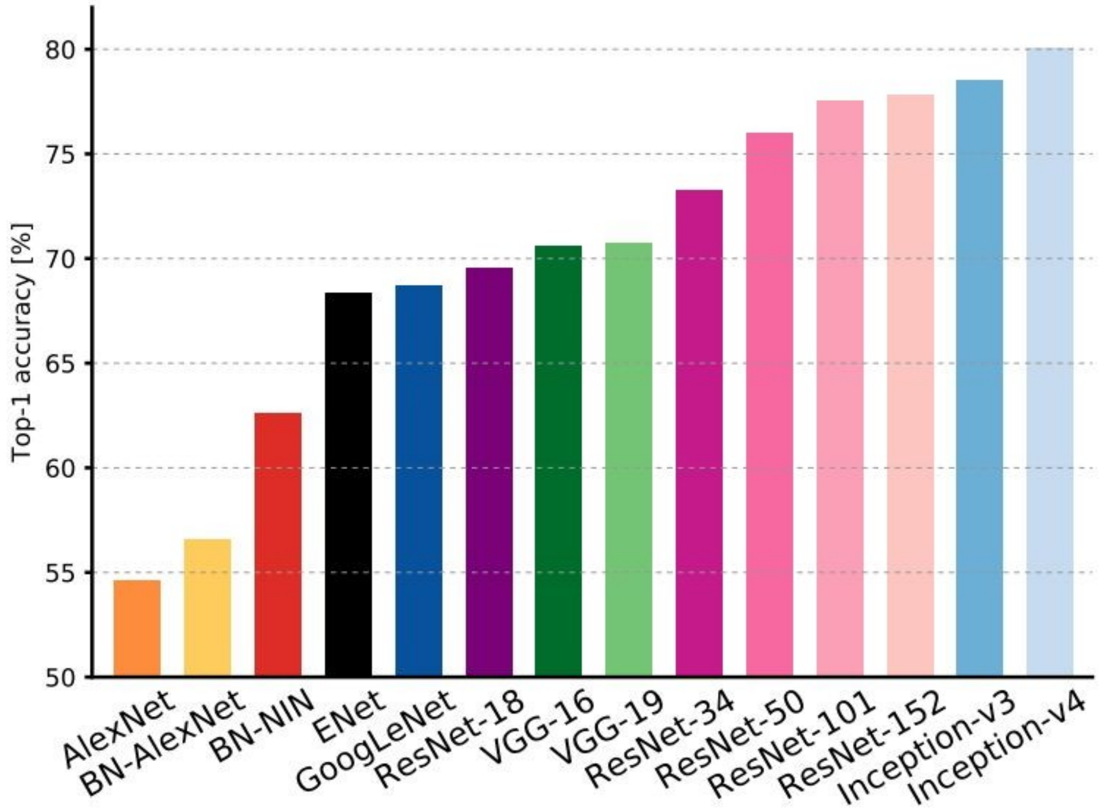
GAN Applications

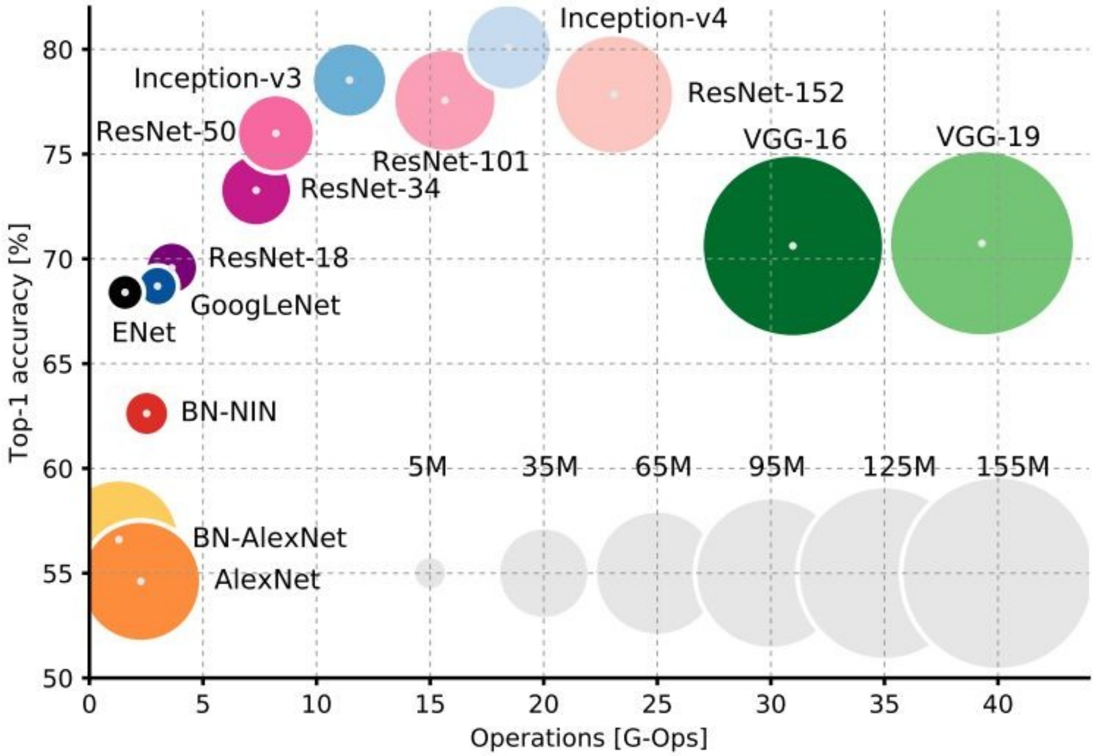
Review - Autoencoders

- Neural network trained copy its input to its output
- Divided into two pieces
  - Encoder - $h = f(x)$
  - Decoder - $r = g(h)$
- Learns $g\big(f(x)\big) = x$
- Trained like a normal neural network
- Considered a form of unsupervised learning



Review – Model Architectures

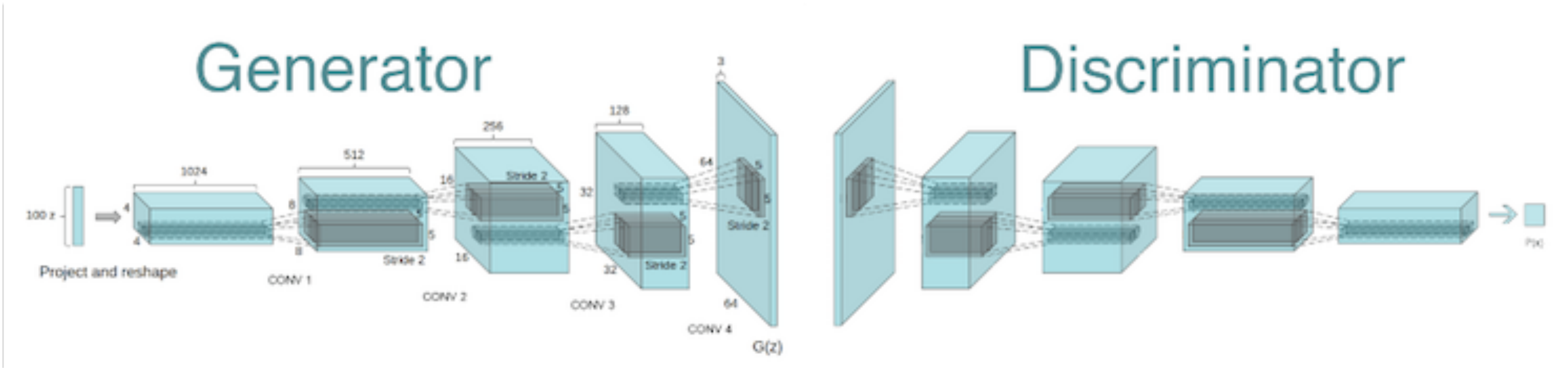




An Analysis of Deep Neural Network Models for Practical Applications, 2017.

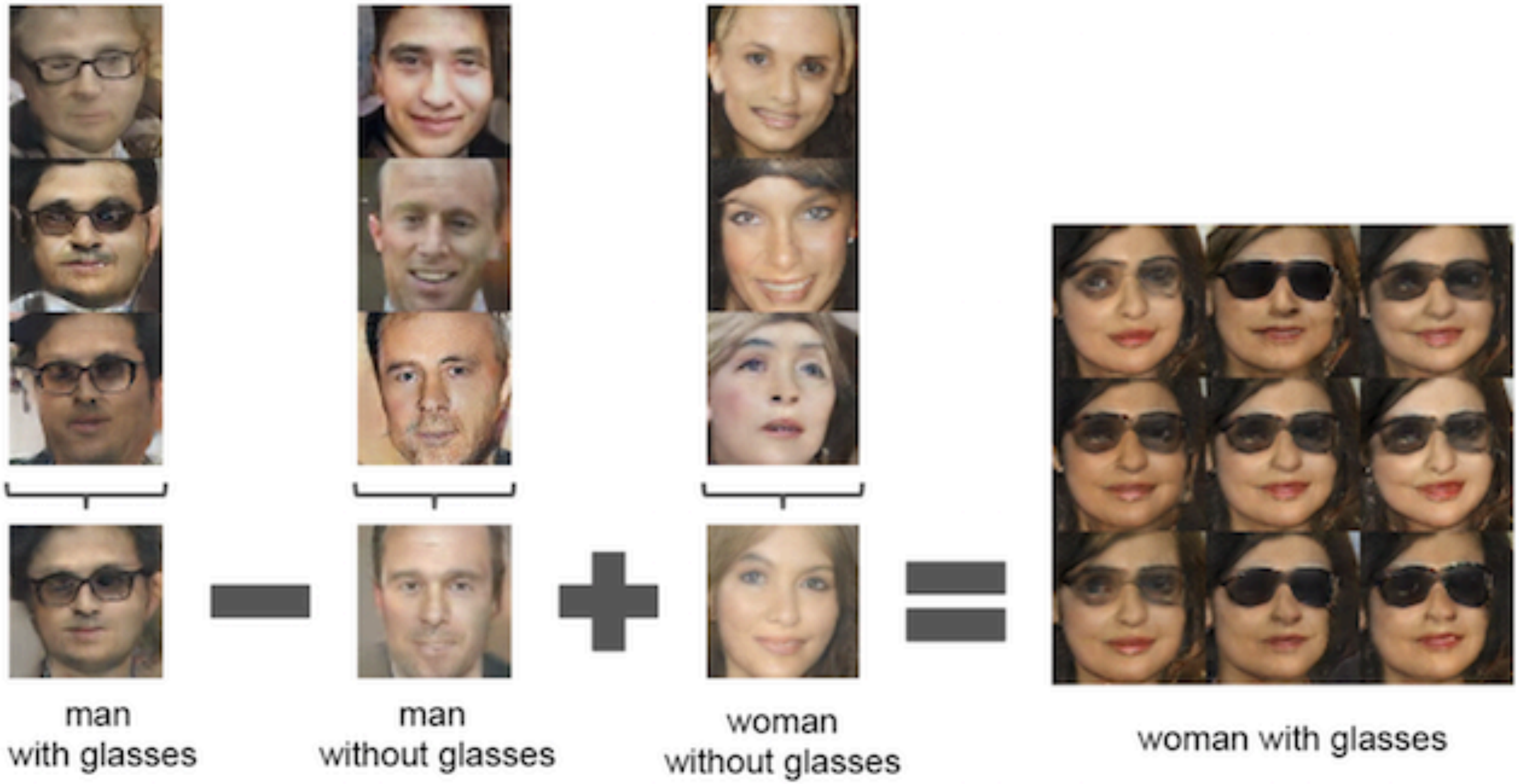
Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (2015)

- Similar Generator/Discriminator architecture to GAN

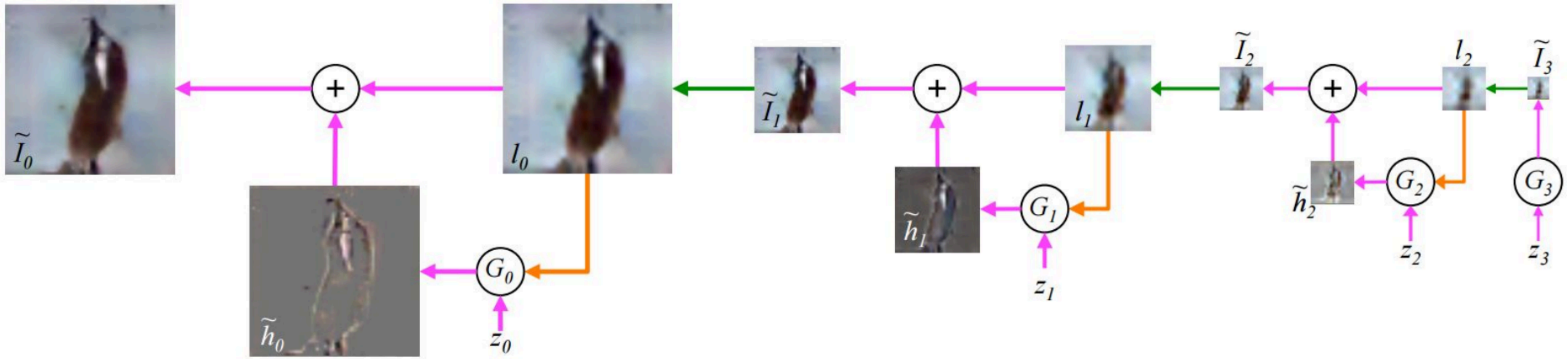


Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (2015)

Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks (2015)

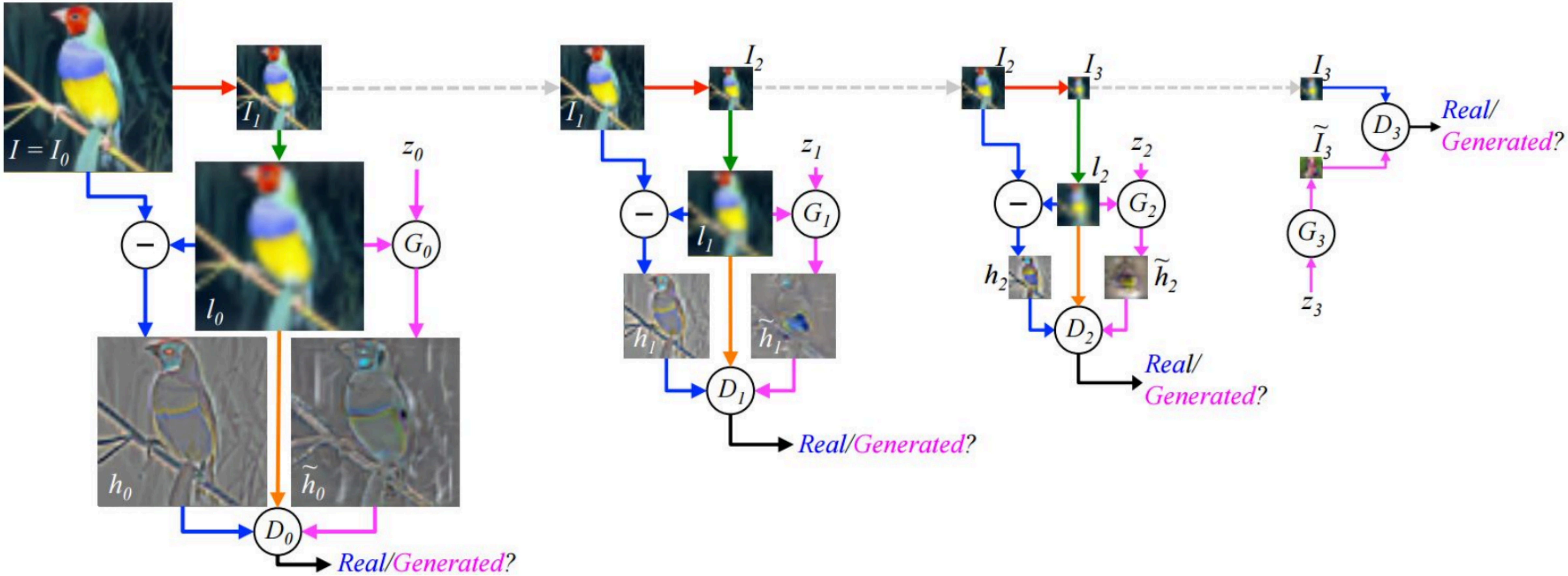


Laplacian Pyramid Generative Adversarial Network

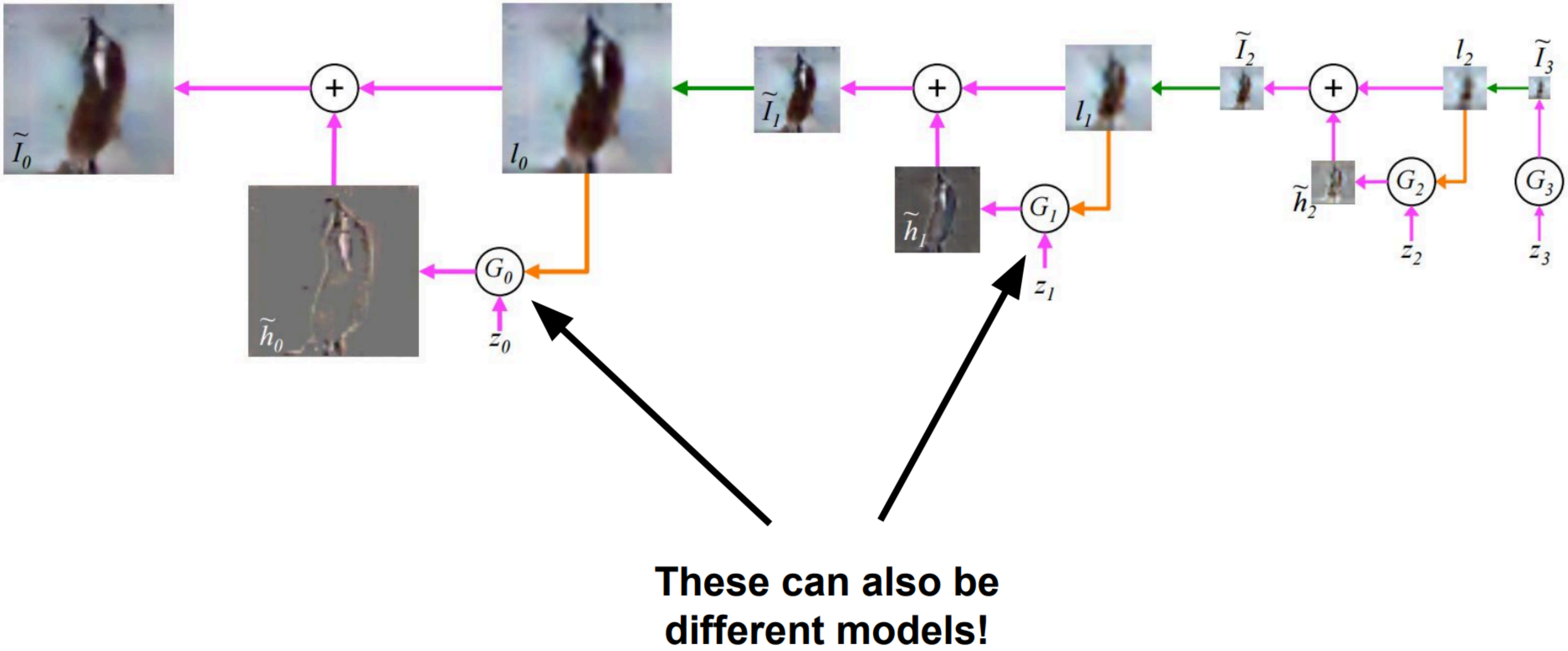


$$\tilde{I}_k = u(\tilde{I}_{k+1}) + \tilde{h}_k = u(\tilde{I}_{k+1}) + G_k(z_k, u(\tilde{I}_{k+1}))$$

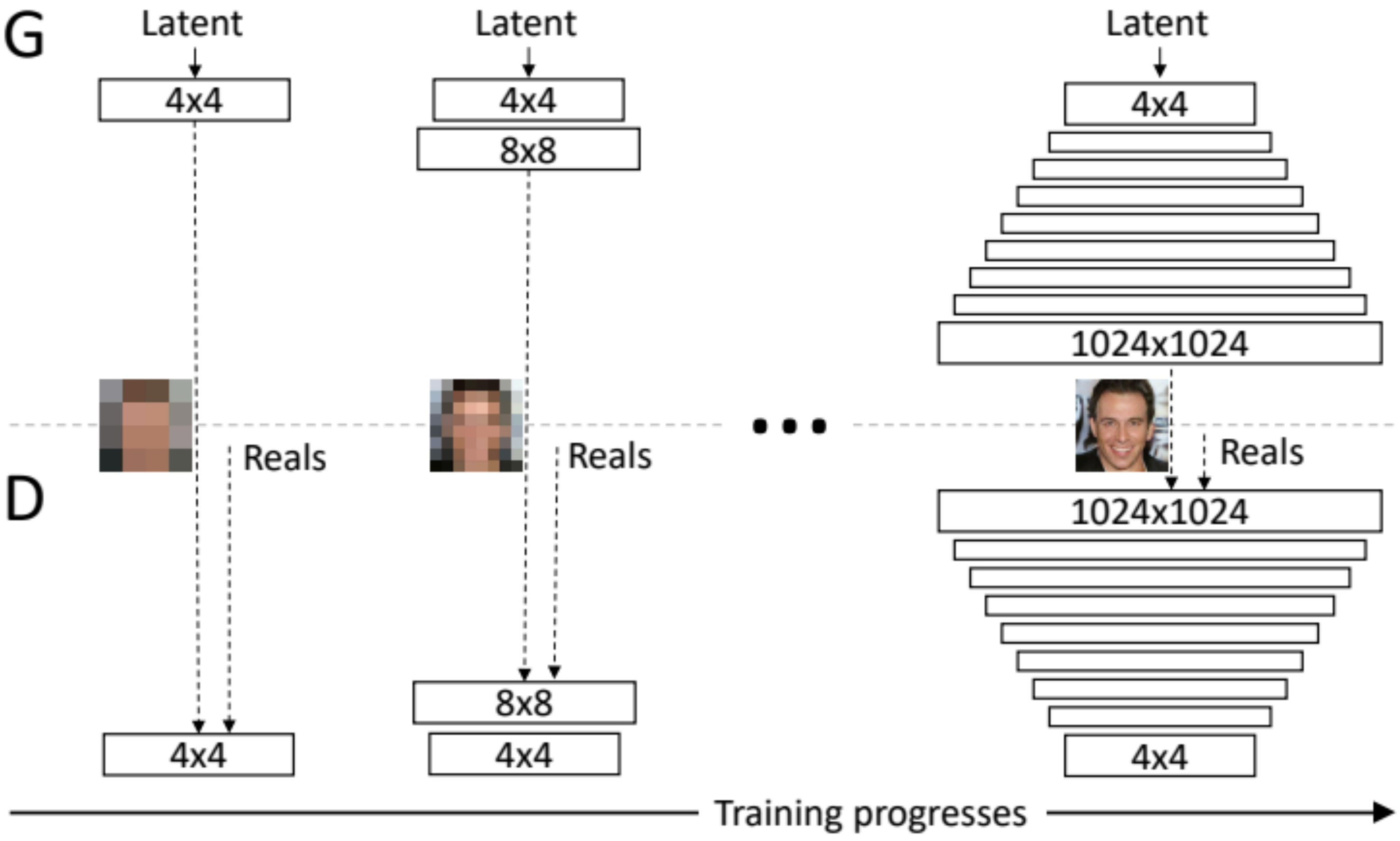
Laplacian Pyramid Generative Adversarial Network



Laplacian Pyramid Generative Adversarial Network



Progressive Growing of GANs for Improved Quality, Stability, and Variation



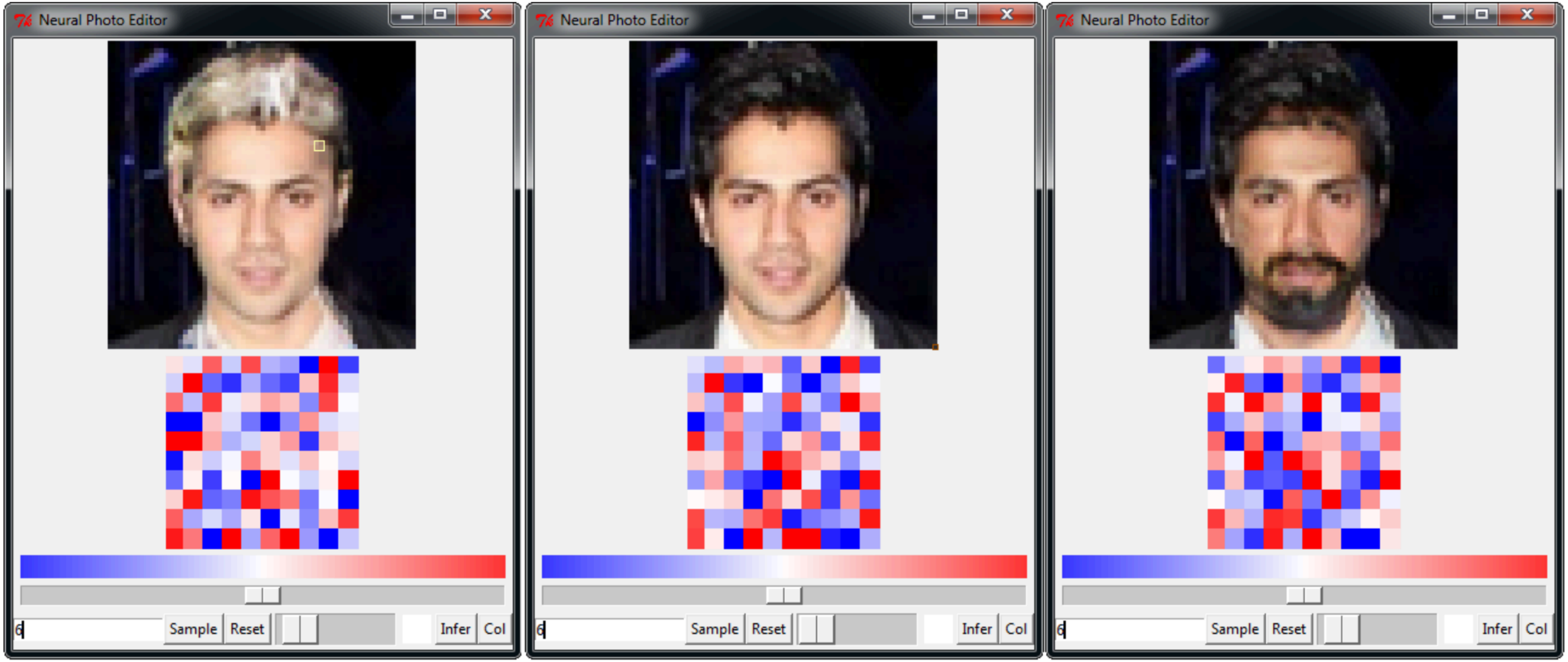
Progressive Growing of GANs for Improved Quality, Stability, and Variation

Training configuration	CelebA						LSUN bedroom					
	Sliced Wasserstein distance $\times 10^3$					MS-SSIM	Sliced Wasserstein distance $\times 10^3$					MS-SSIM
	128	64	32	16	Avg		128	64	32	16	Avg	
(a) Gulrajani et al. (2017)	12.99	7.79	7.62	8.73	9.28	0.2854	11.97	10.51	8.03	14.48	11.25	**0.0587**
(b) + Progressive growing	4.62	**2.64**	3.78	6.06	4.28	**0.2838**	7.09	6.27	7.40	9.64	7.60	0.0615
(c) + Small minibatch	75.42	41.33	41.62	26.57	46.23	0.4065	72.73	40.16	42.75	42.46	49.52	0.1061
(d) + Revised training parameters	9.20	6.53	4.71	11.84	8.07	0.3027	7.39	5.51	3.65	9.63	6.54	0.0662
(e*) + Minibatch discrimination	10.76	6.28	6.04	16.29	9.84	0.3057	10.29	6.22	5.32	11.88	8.43	0.0648
(e) Minibatch stddev	13.94	5.67	2.82	5.71	7.04	0.2950	7.77	5.23	3.27	9.64	6.48	0.0671
(f) + Equalized learning rate	4.42	3.28	2.32	7.52	4.39	0.2902	**3.61**	3.32	**2.71**	6.44	4.02	0.0668
(g) + Pixelwise normalization	**4.06**	3.04	**2.02**	**5.13**	**3.56**	0.2845	3.89	**3.05**	3.24	**5.87**	**4.01**	0.0640
(h) Converged	2.42	2.17	2.24	4.99	2.96	0.2828	3.47	2.60	2.30	4.87	3.31	0.0636

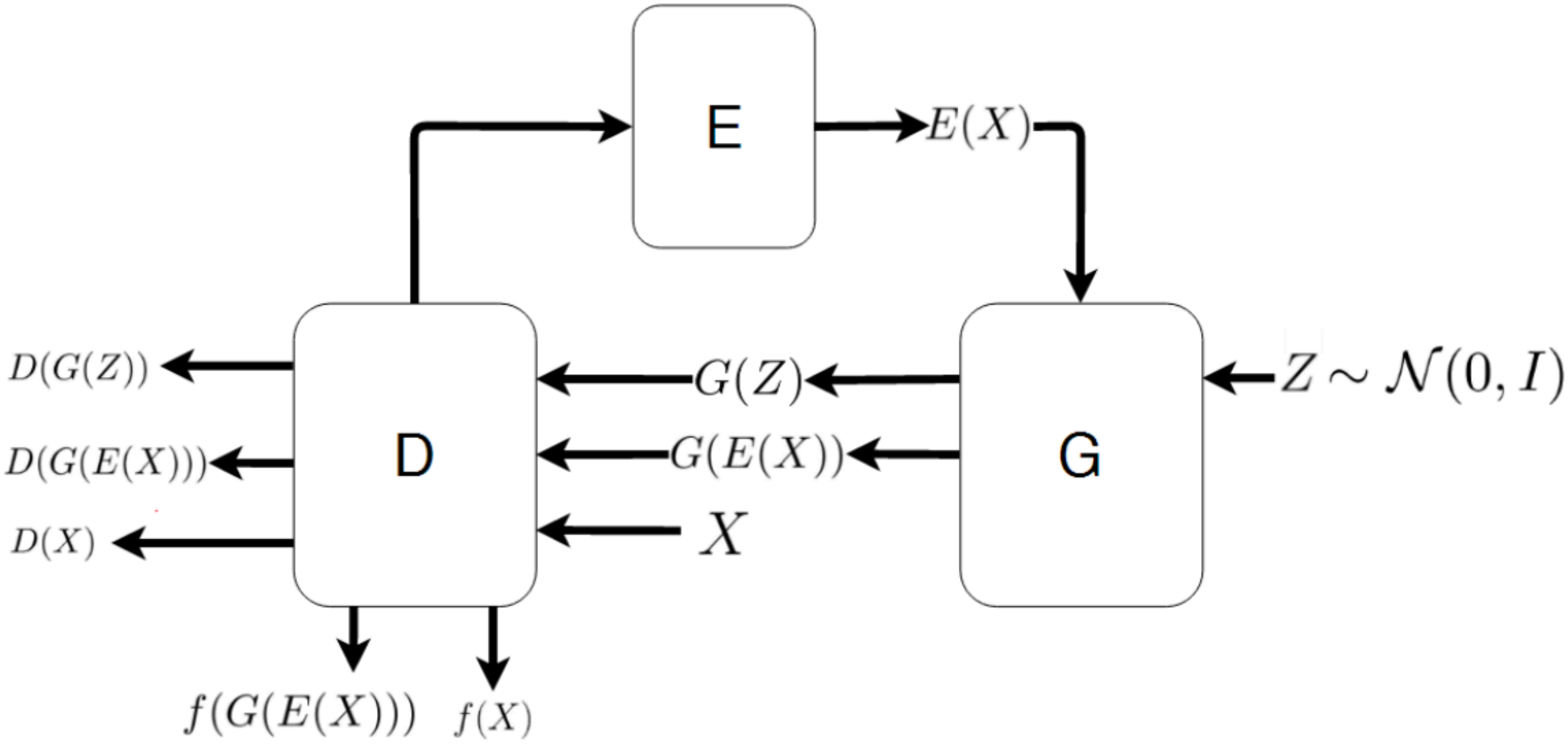
https://www.youtube.com/watch?v=G06dEcZ-QTg&feature=youtu.be

Progression of GAN Quality

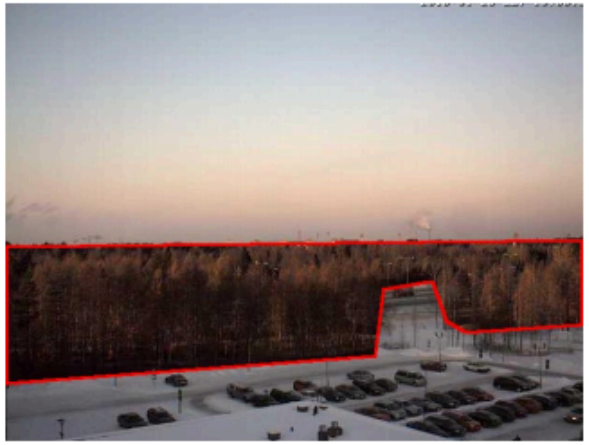
Neural Photo Editing with Introspective Adversarial Networks



Neural Photo Editing with Introspective Adversarial Networks



GP-GAN Towards High-Resolution Image Editing



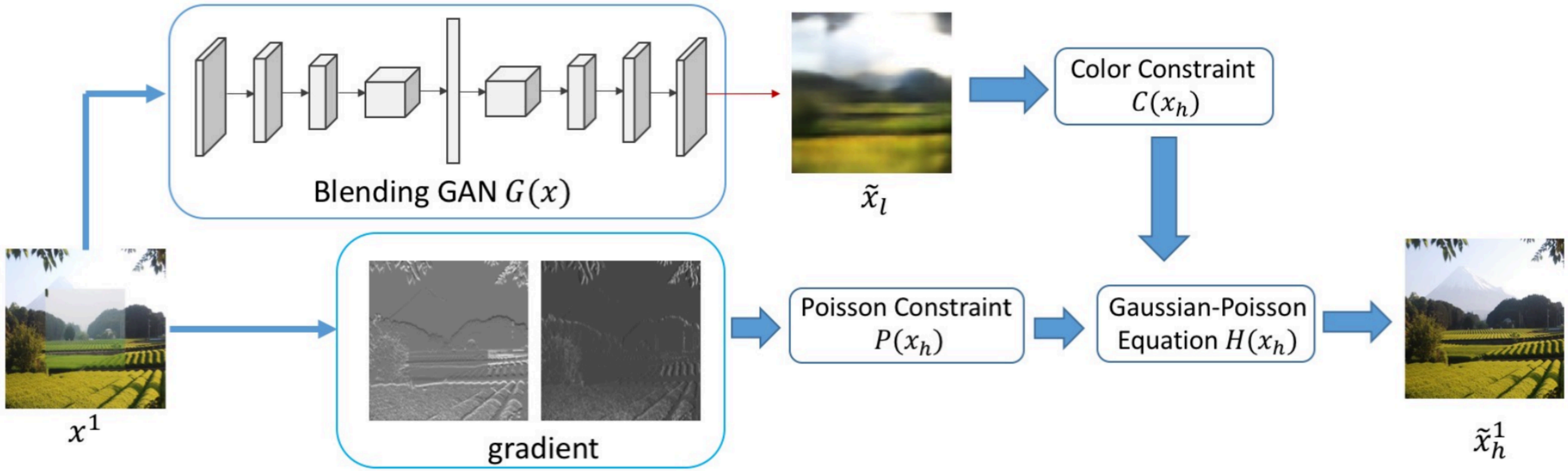
(a) Copy-and-Paste Input

(b) MPB [30]

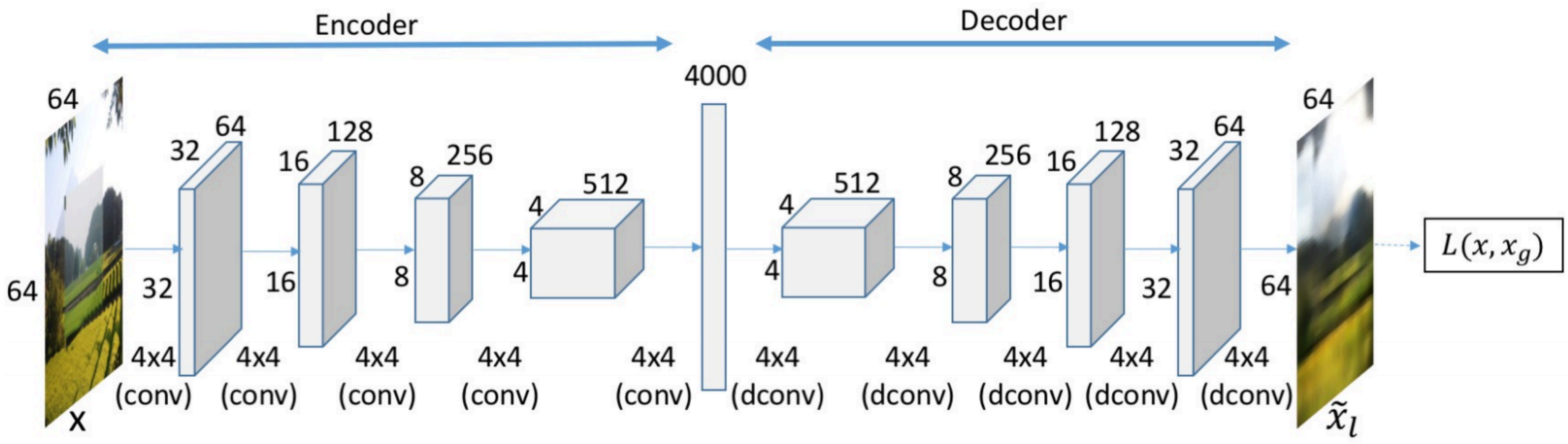
(c) MSB [28]

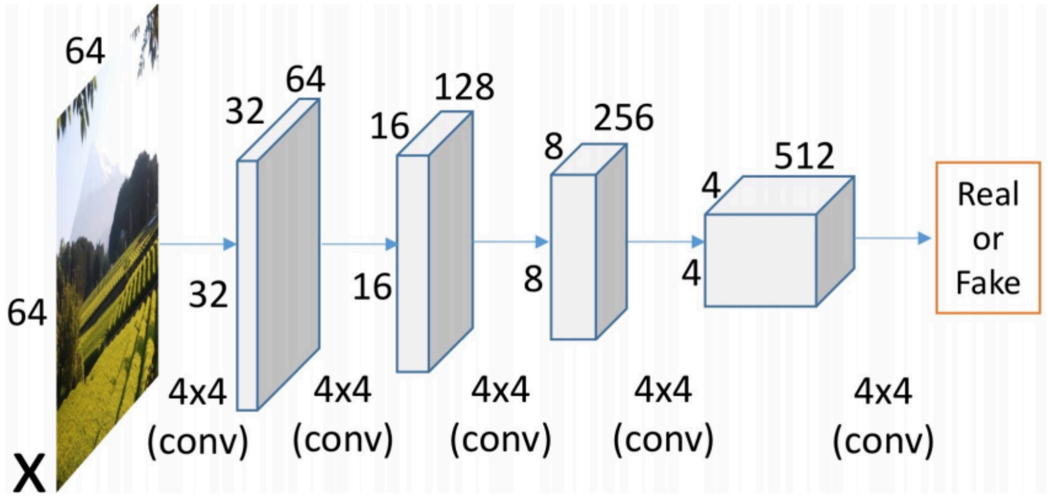
(d) Ours

GP-GAN Towards High-Resolution Image Editing

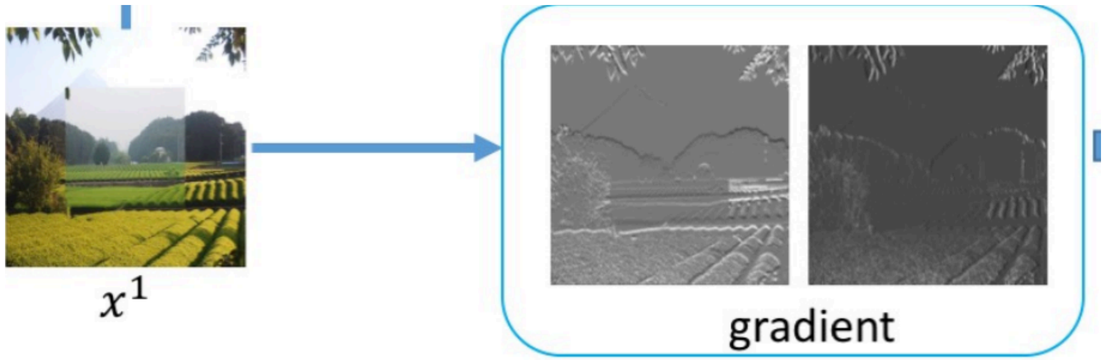


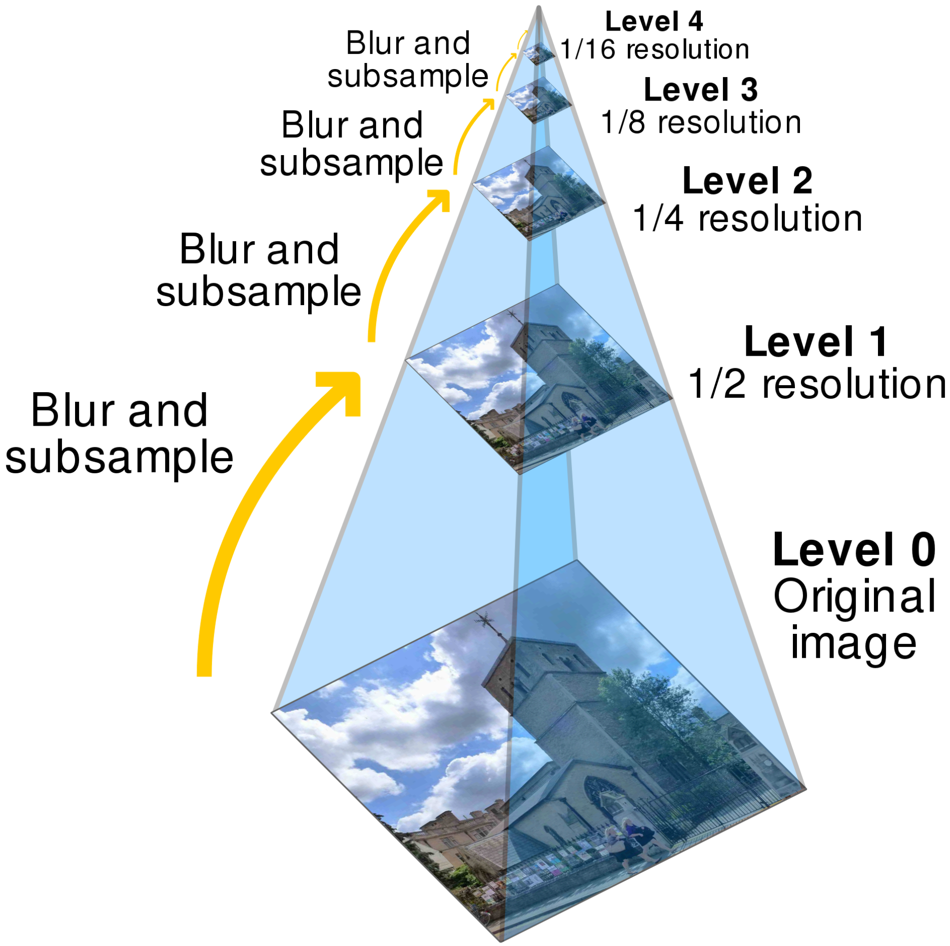
GP-GAN Towards High-Resolution Image Editing

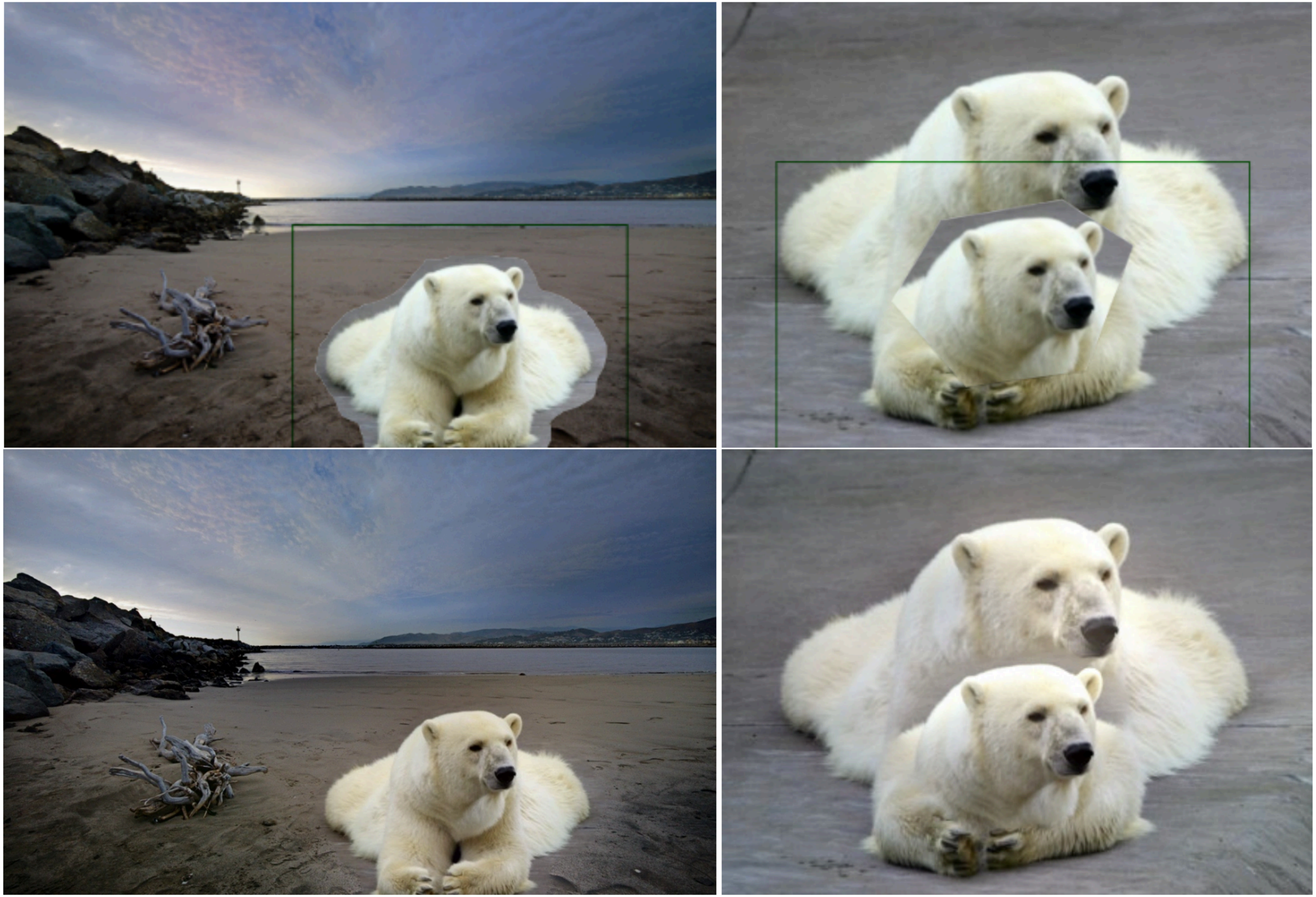




GP-GAN Towards High-Resolution Image Editing







GP-GAN Towards High-Resolution Image Editing

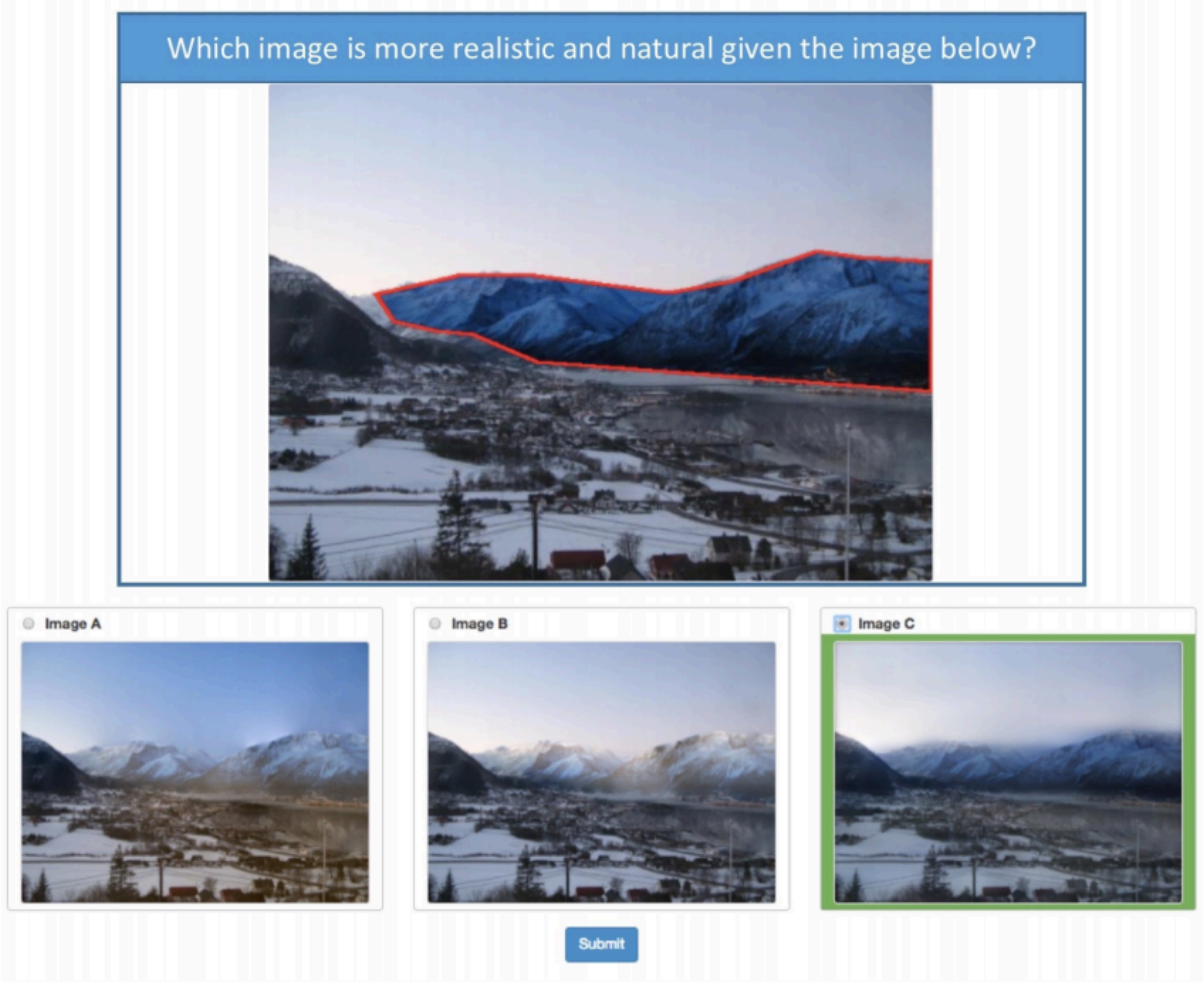


Photo-Realistic Image Super-Resolution Using a Generative Adversarial Network

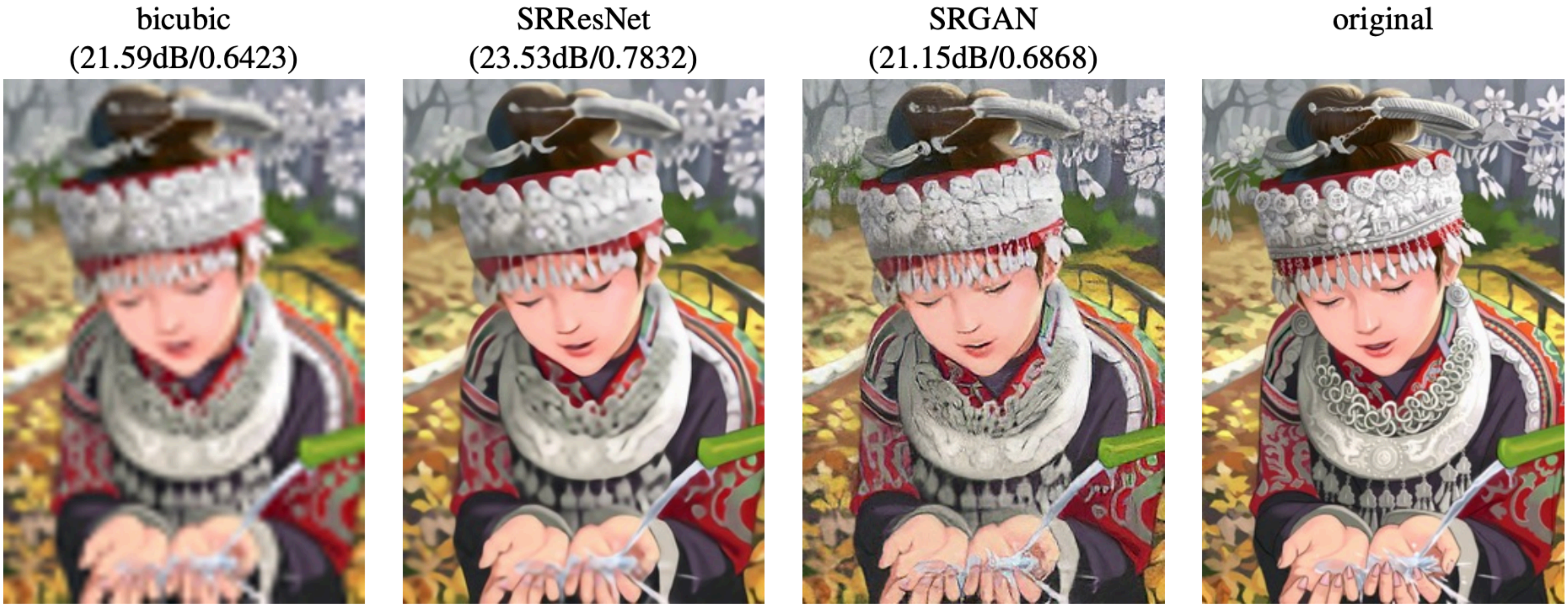


Photo-Realistic Image Super-Resolution Using a Generative Adversarial Network

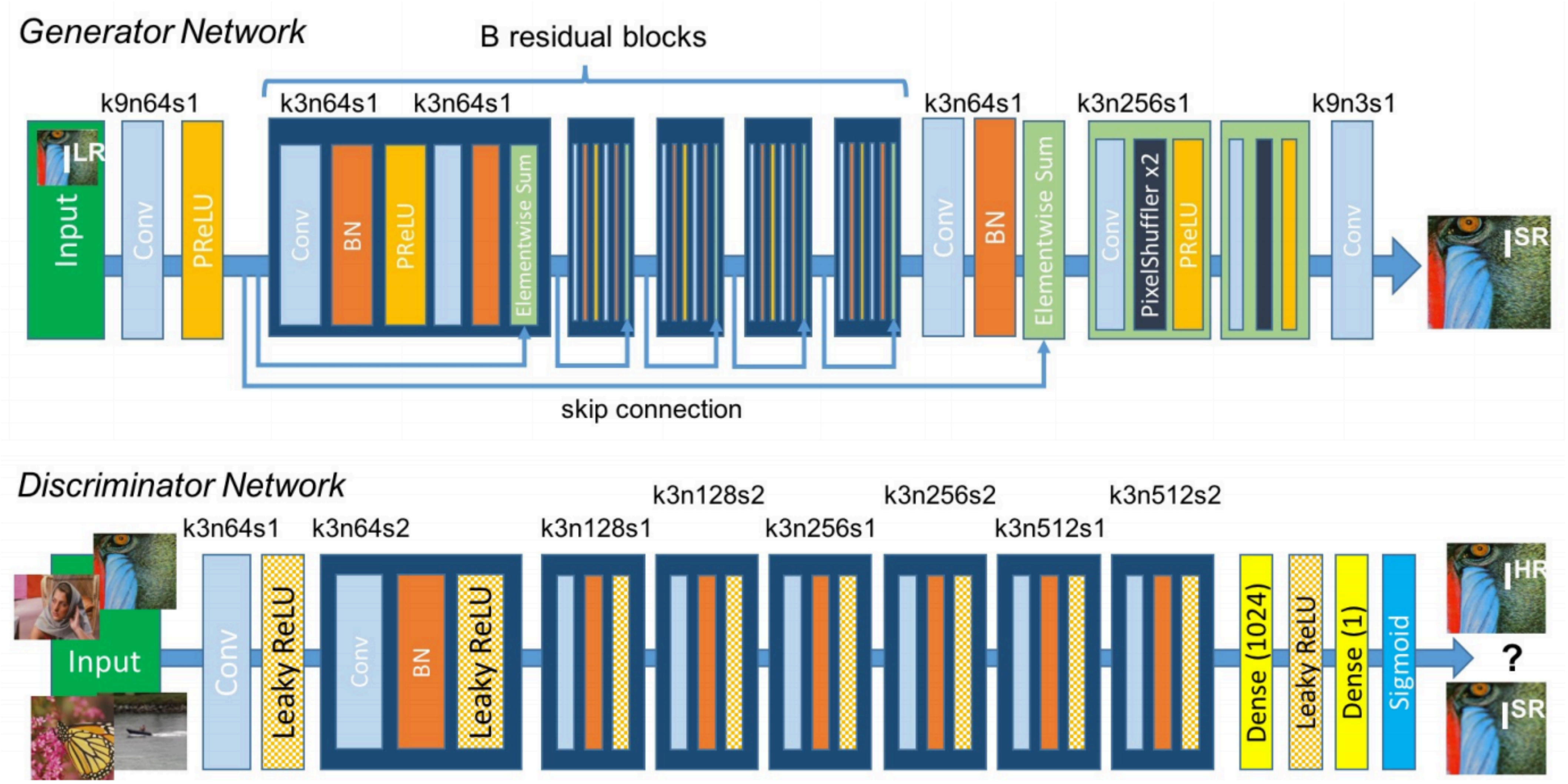


Photo-Realistic Image Super-Resolution Using a Generative Adversarial Network

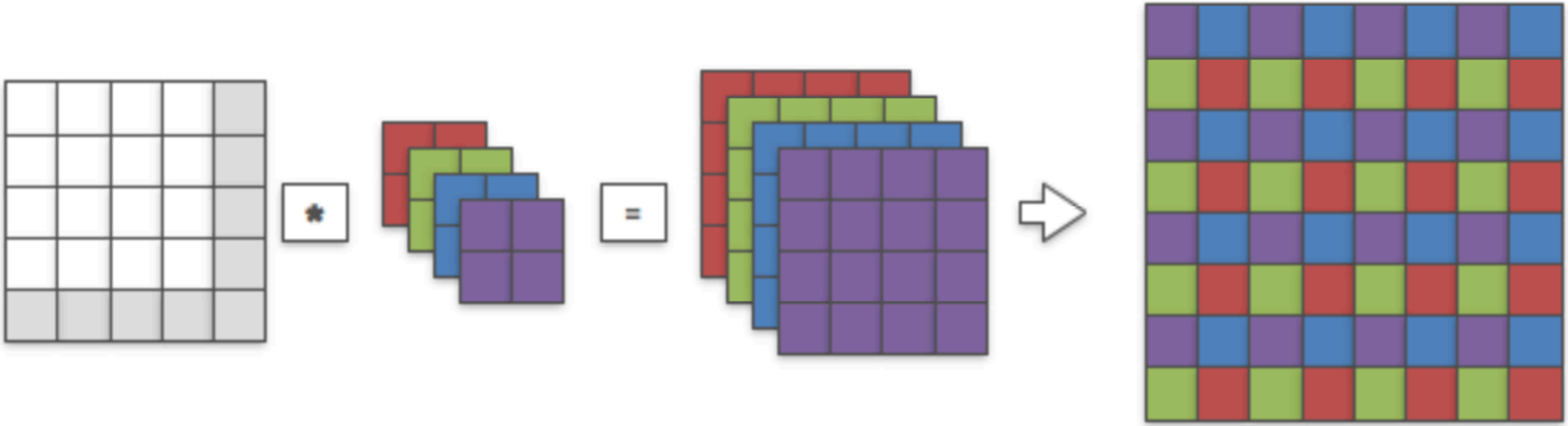


Photo-Realistic Image Super-Resolution Using a Generative Adversarial Network

- Multi-Part Loss Function –
- Pixel-Wise Loss –

$$I_{MSE}^{SR} = \sum_{i} \left(I_i^{HR} - G(I^{SR})_i \right)^2$$

- Adversarial Loss
- VGG Loss – Loss of the activation function between the reconstructed and high resolution image

Photo-Realistic Image Super-Resolution Using a Generative Adversarial Network