Unified Signal Compression Using Generative Adversarial Network

Bowen Liu   Ang Cao   Hun-Seok Kim

Department of EECS, University of Michigan - Ann Arbor, USA

45th ICASSP Virtual Meeting
May 5th, 2020

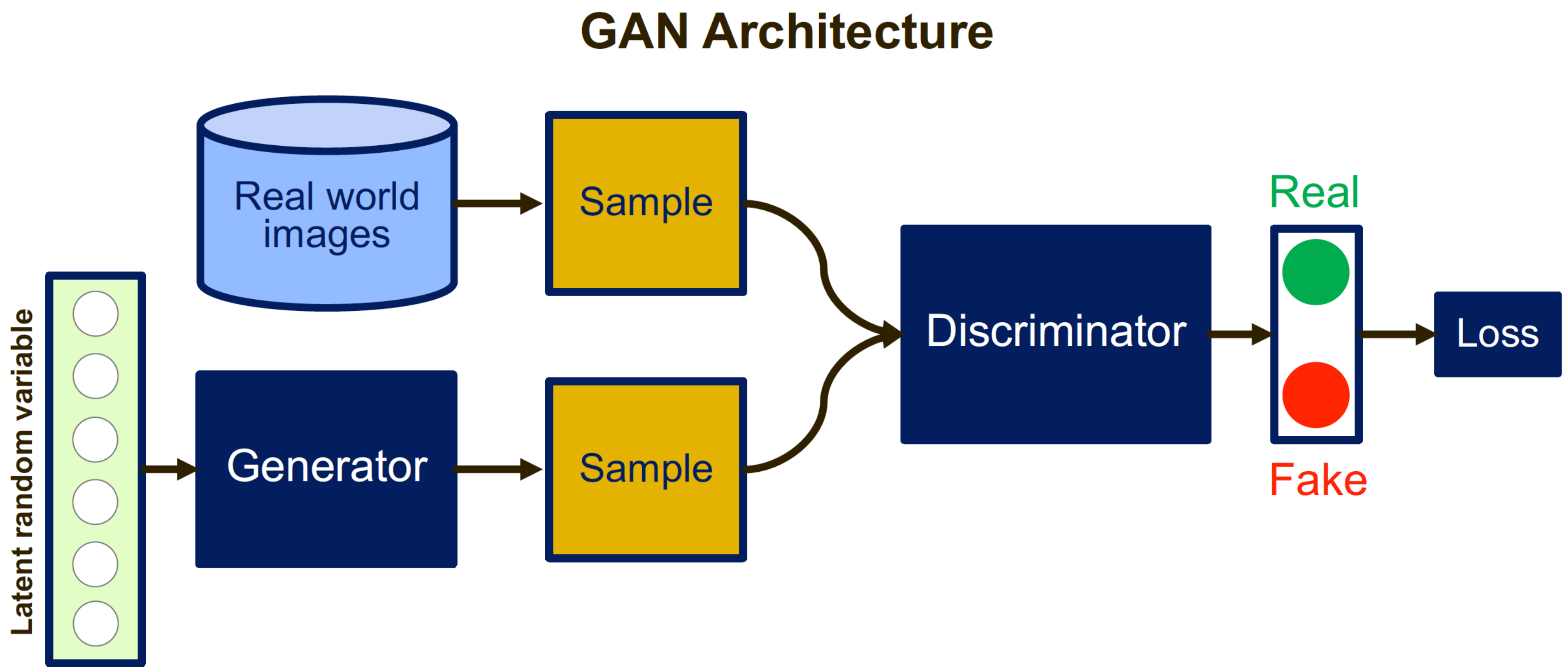
Outline

- Motivation
  - DNN-based unified signal compression algorithm for image and speech
- New Framework
  - Back Propagation Generative Adversarial Network (BPGAN)
- Methodology
  - Compression / decompression
  - BPGAN training
- Result and Evaluation

Introduction: Signal compression

- Motivation for signal compression
  - To reduce latency & bandwidth for data communication
  - To reduce space for data storage
  - 3000 color images ($1800\times2400\times24\text{bits} \approx 32\text{GB}$)
  - 60 minutes stereo audio ($320\text{kbps} \approx 1\text{GB}$)
- Image compression
  - Conventional algorithms: JPEG, BPG
- Speech compression
  - Conventional algorithms: CELP,AMR
- Research question
  - Can DNN based algorithm outperform conventional compression codecs?
  - Can we unify compression framework for different signal types (image and speech)?

Introduction: GAN



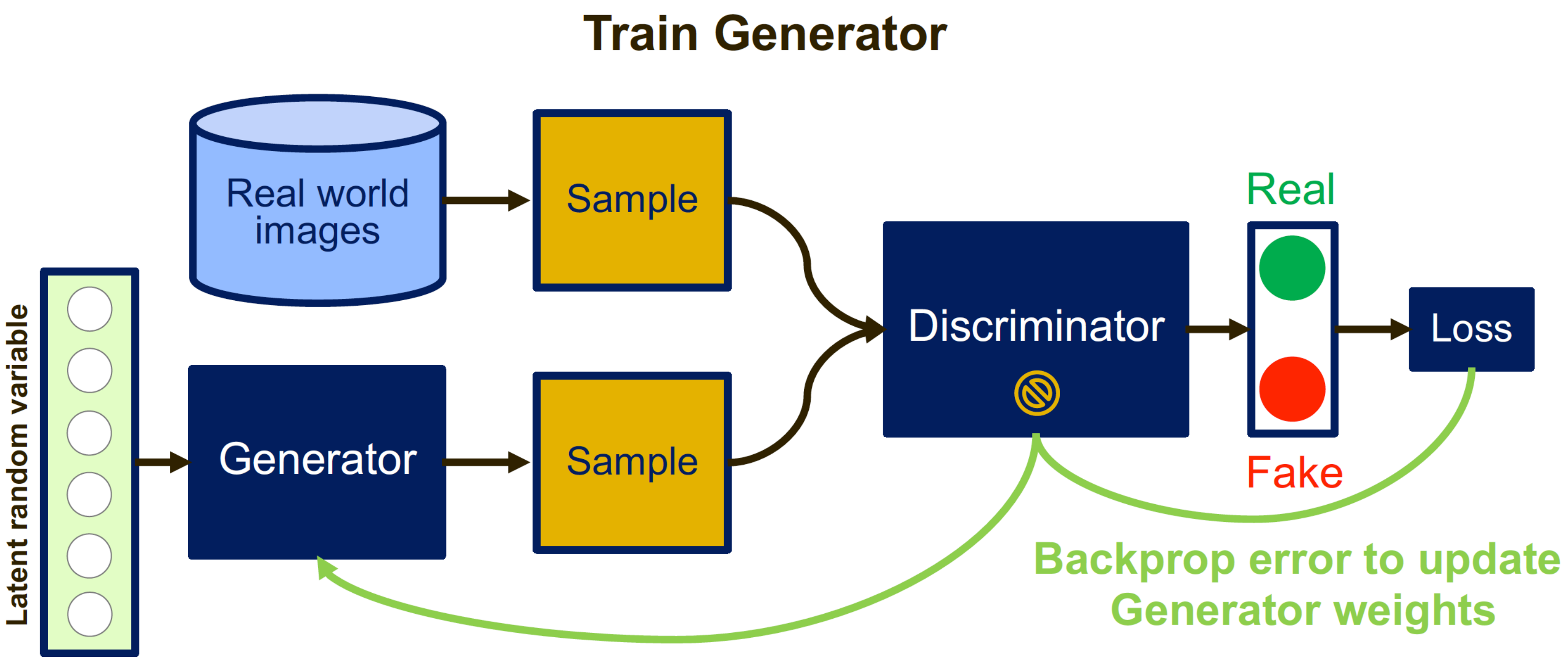
- Generative Adversarial Network (GAN)

  Generative: learn a generative model

  Adversarial: train in an adversarial setting

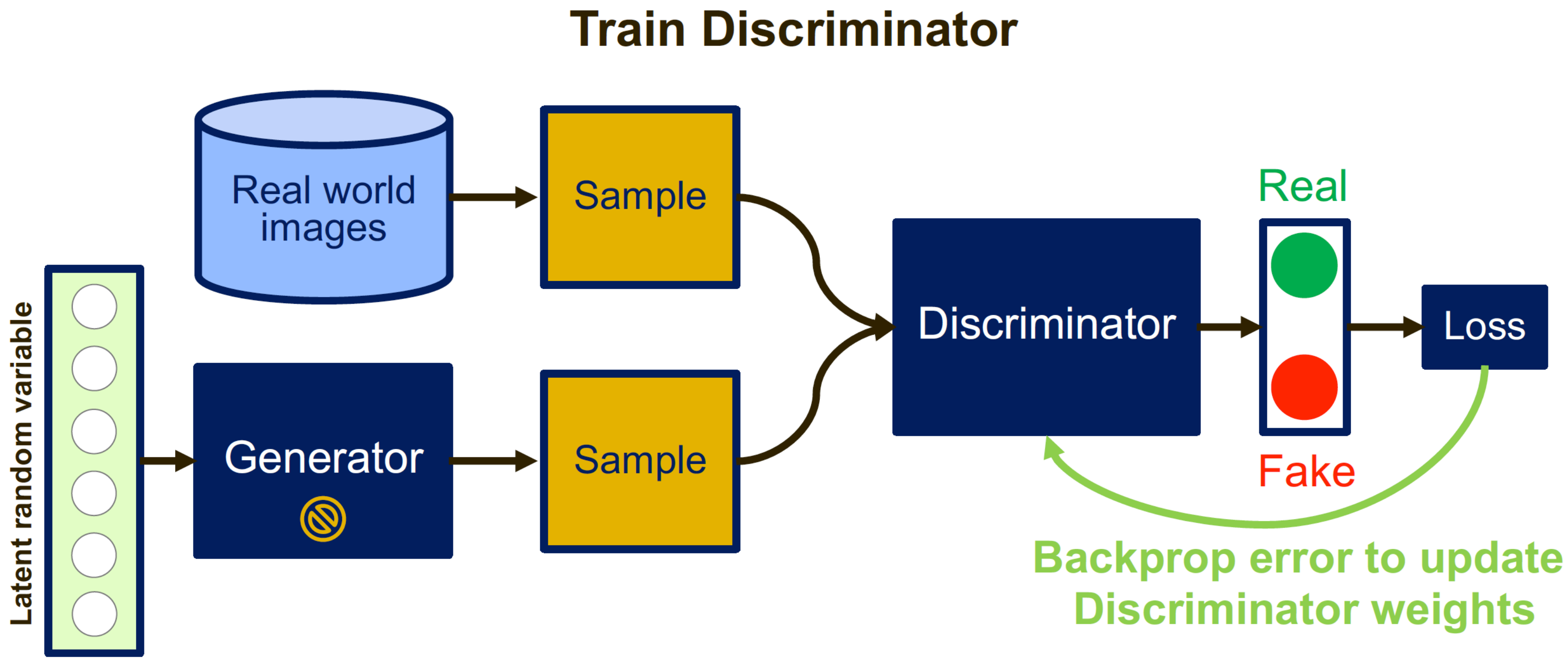
  Networks: use Deep Neural Networks

Introduction: GAN



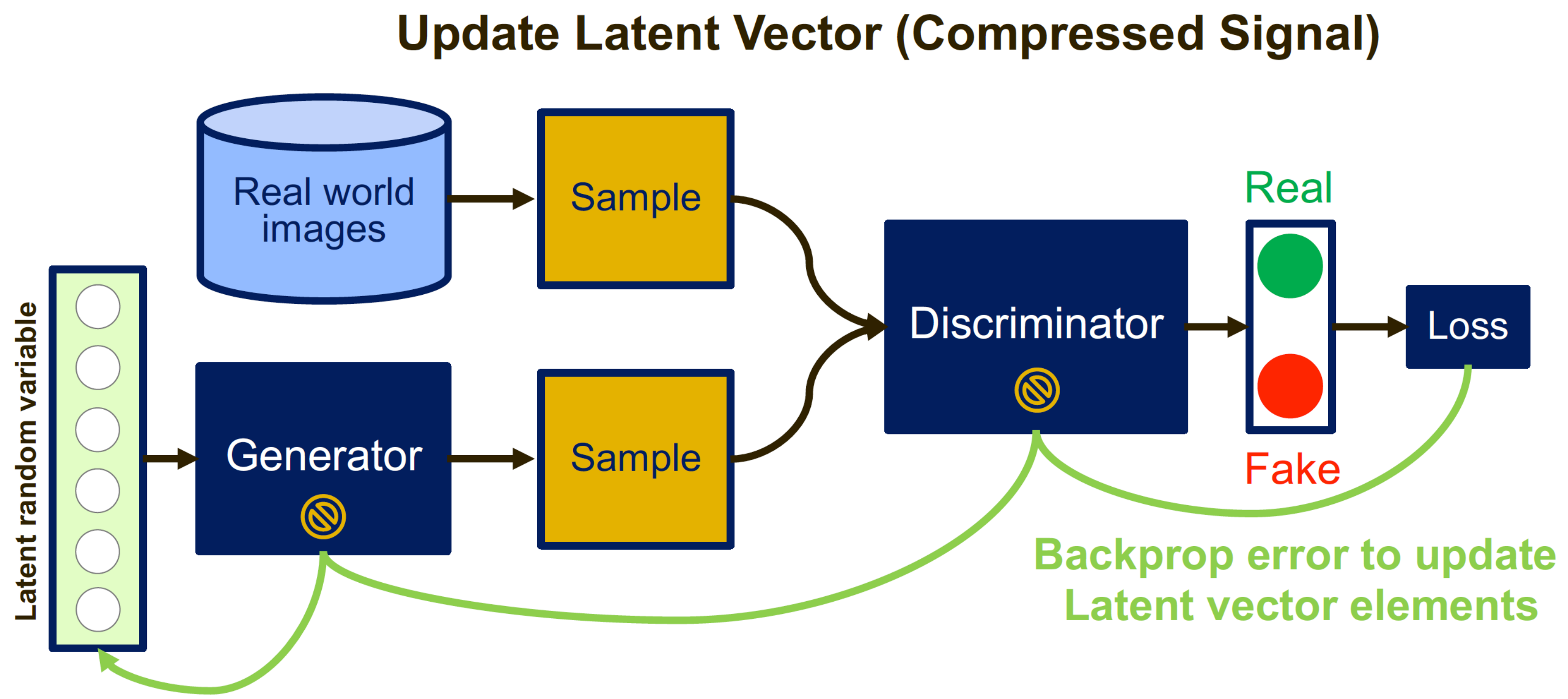
- Generative Adversarial Network (GAN)
  - Generative: learn a generative model
  - Adversarial: train in an adversarial setting
  - Networks: use Deep Neural Networks
- Core idea: Adversarial training
  - Generator: generates indiscriminative samples

Introduction: GAN



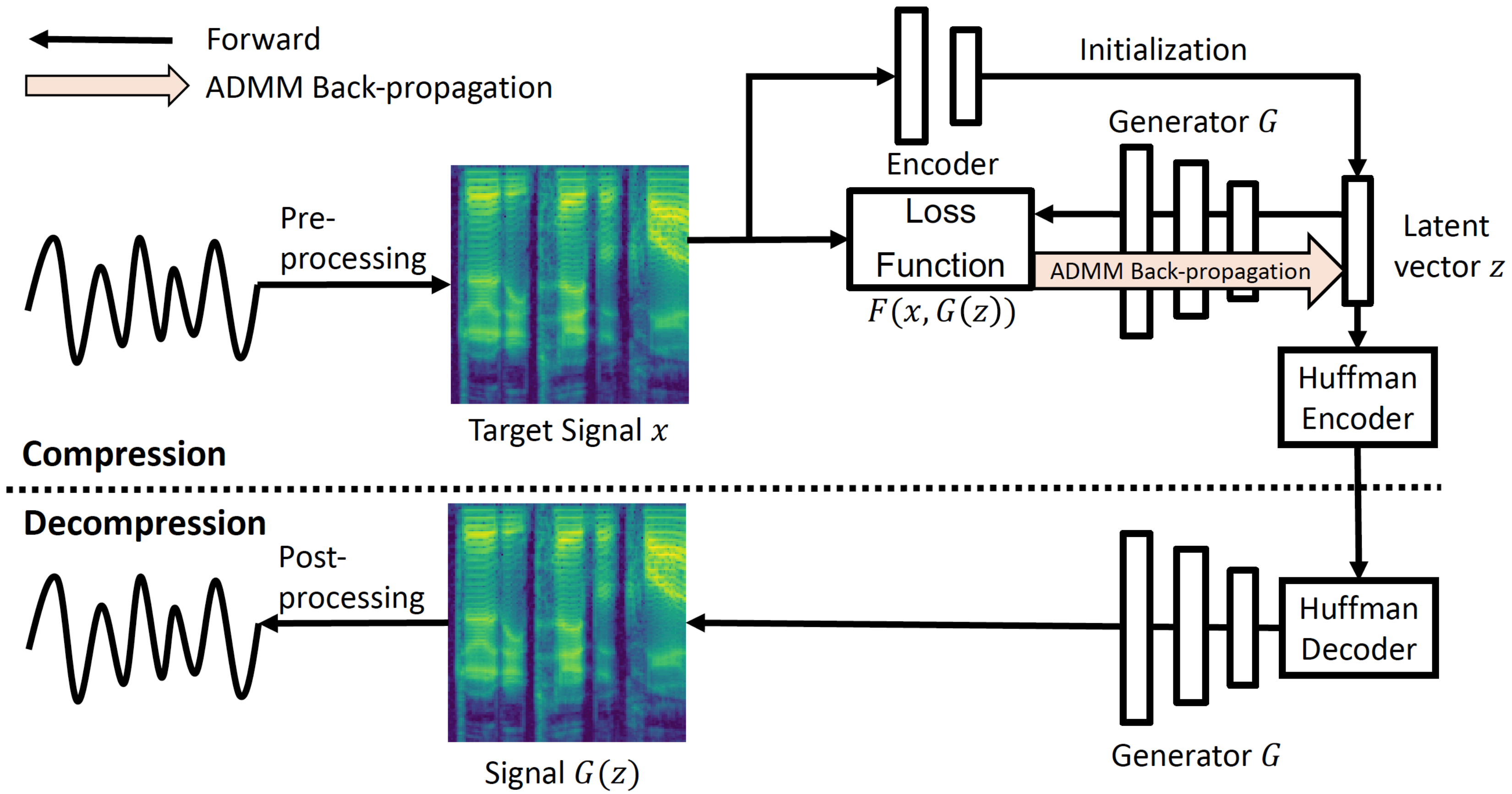
- Generative Adversarial Network (GAN)
  - Generative: learn a generative model
  - Adversarial: train in an adversarial setting
  - Networks: use Deep Neural Networks
- Core idea: Adversarial training
  - Generator: generates indiscriminative samples
  - Discriminator: distinguishes between real and fake samples

Inspiration: BPGAN



- Inspired by GAN, our algorithm updates the latent vector via back-propagation through Discriminator and Generator
  - Fix discriminator and generator weights during updating latent vector

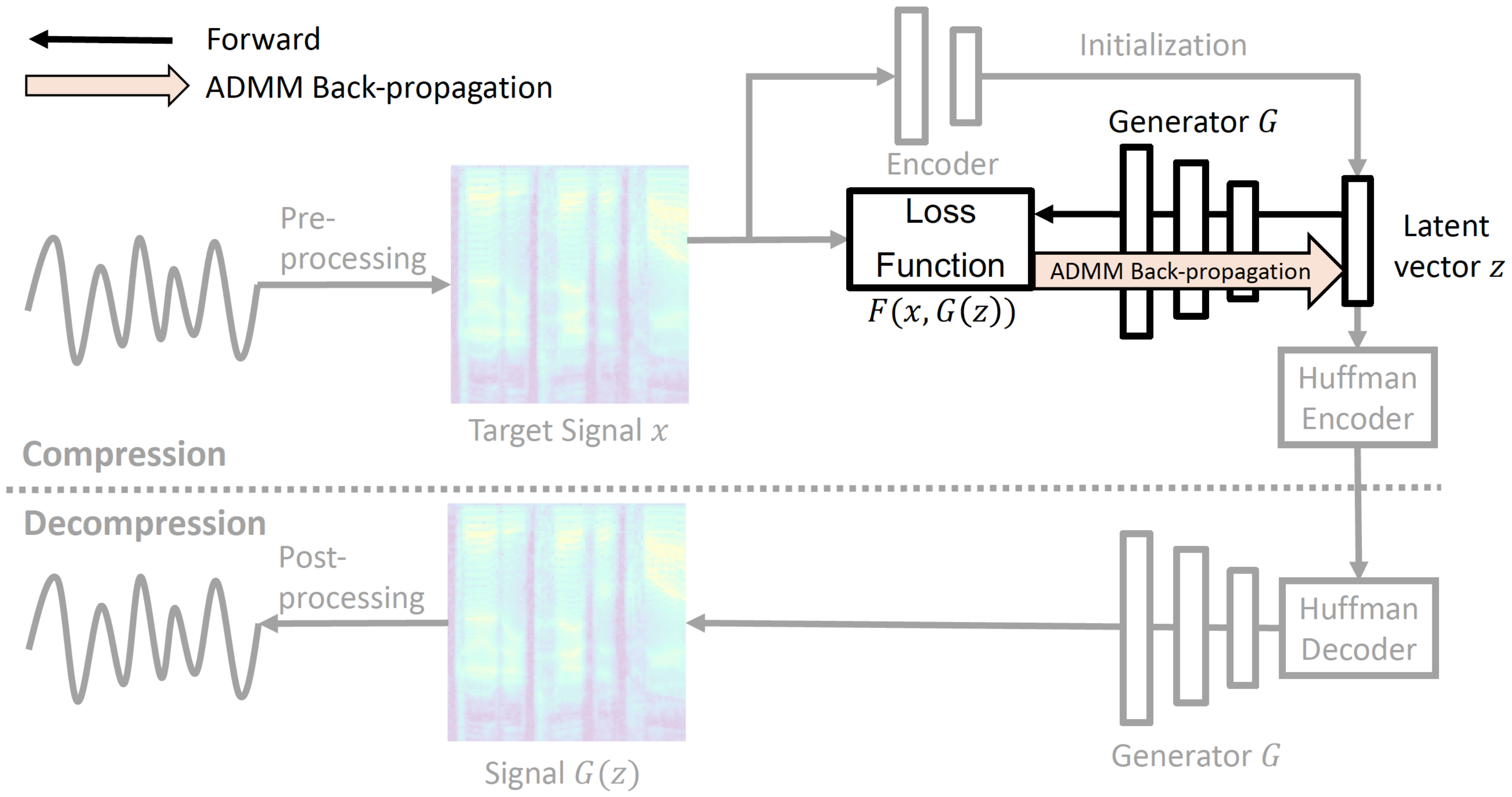
Framework: BPGAN Compression



- Applicable to image and speech compression tasks
- GAN with task specific loss functions

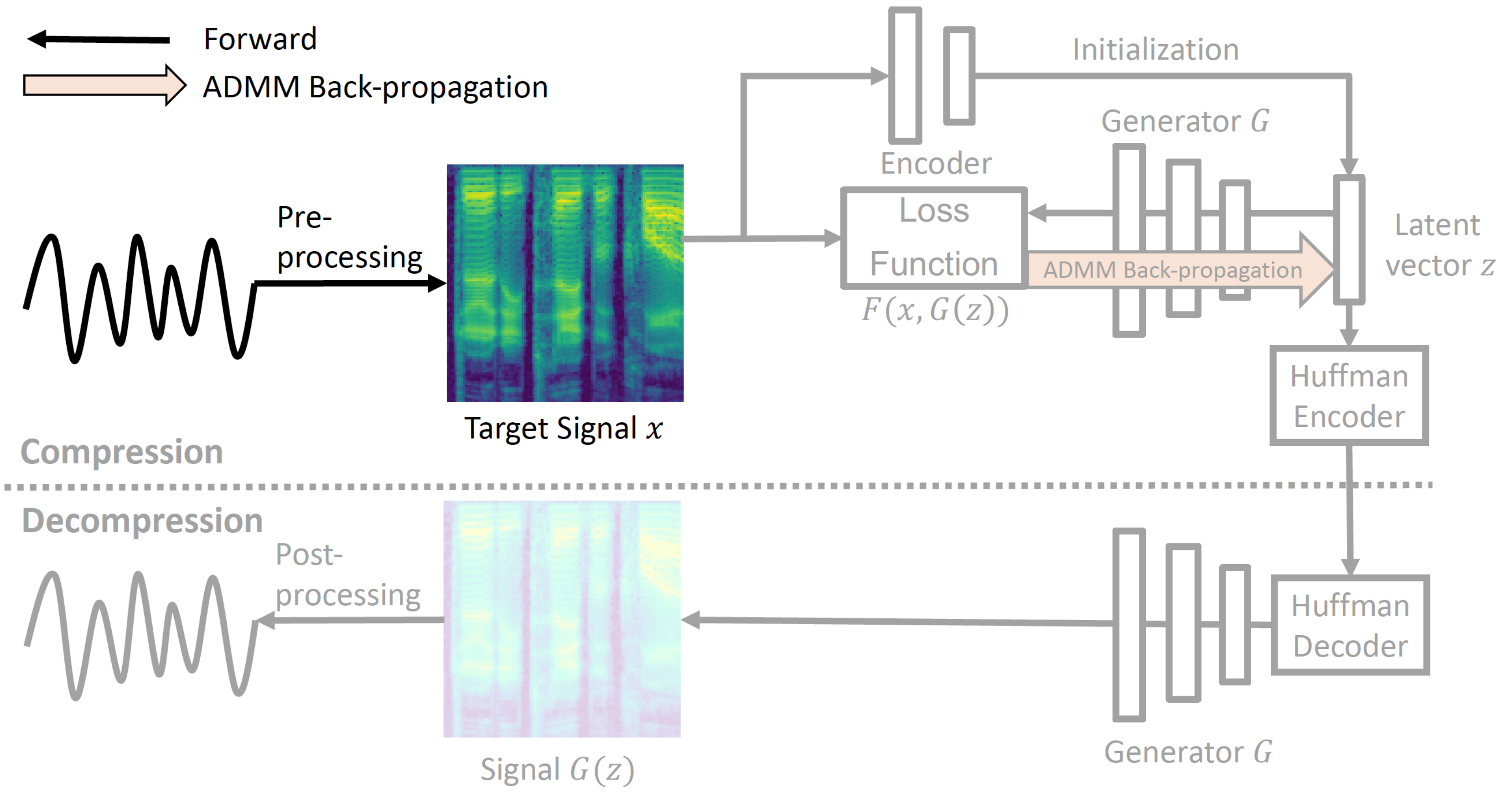
  Improve the quality of generator output

Framework: BPGAN Compression



- Search the compressed signal in latent space
  - $\boldsymbol{z}$ is the input to the generator $\boldsymbol{G}$
  - Optimize $\boldsymbol{z}$ that minimizes loss between target signal $\boldsymbol{x}$ and $\boldsymbol{G(z)}$

Step 1: Signal pre-processing



- **Image:**

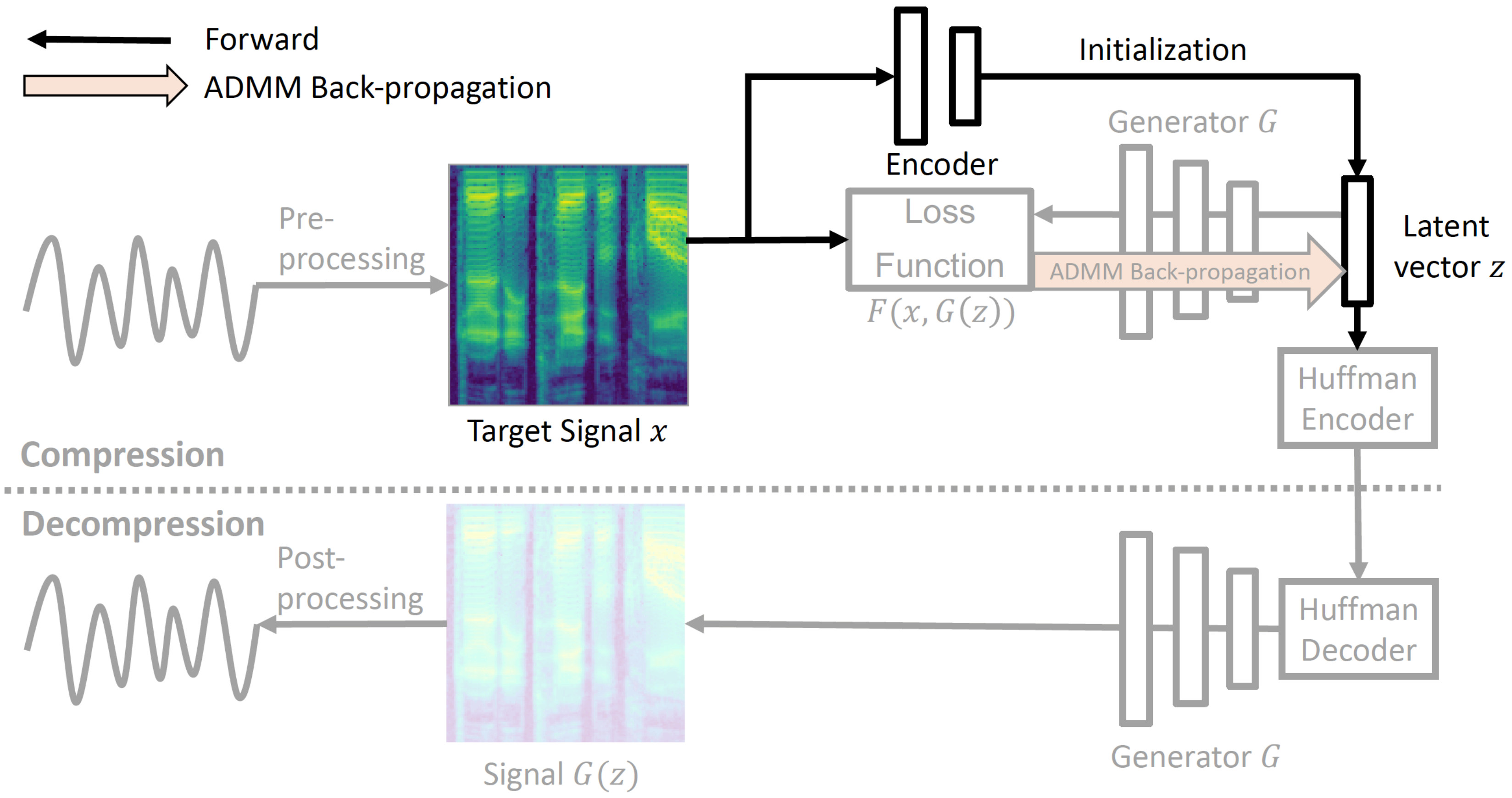
  Resize the image to $n \times m$ (pre-defined) pixels

- **Audio:**

  Use Short Time Fourier Transformation (STFT) to get the spectrogram

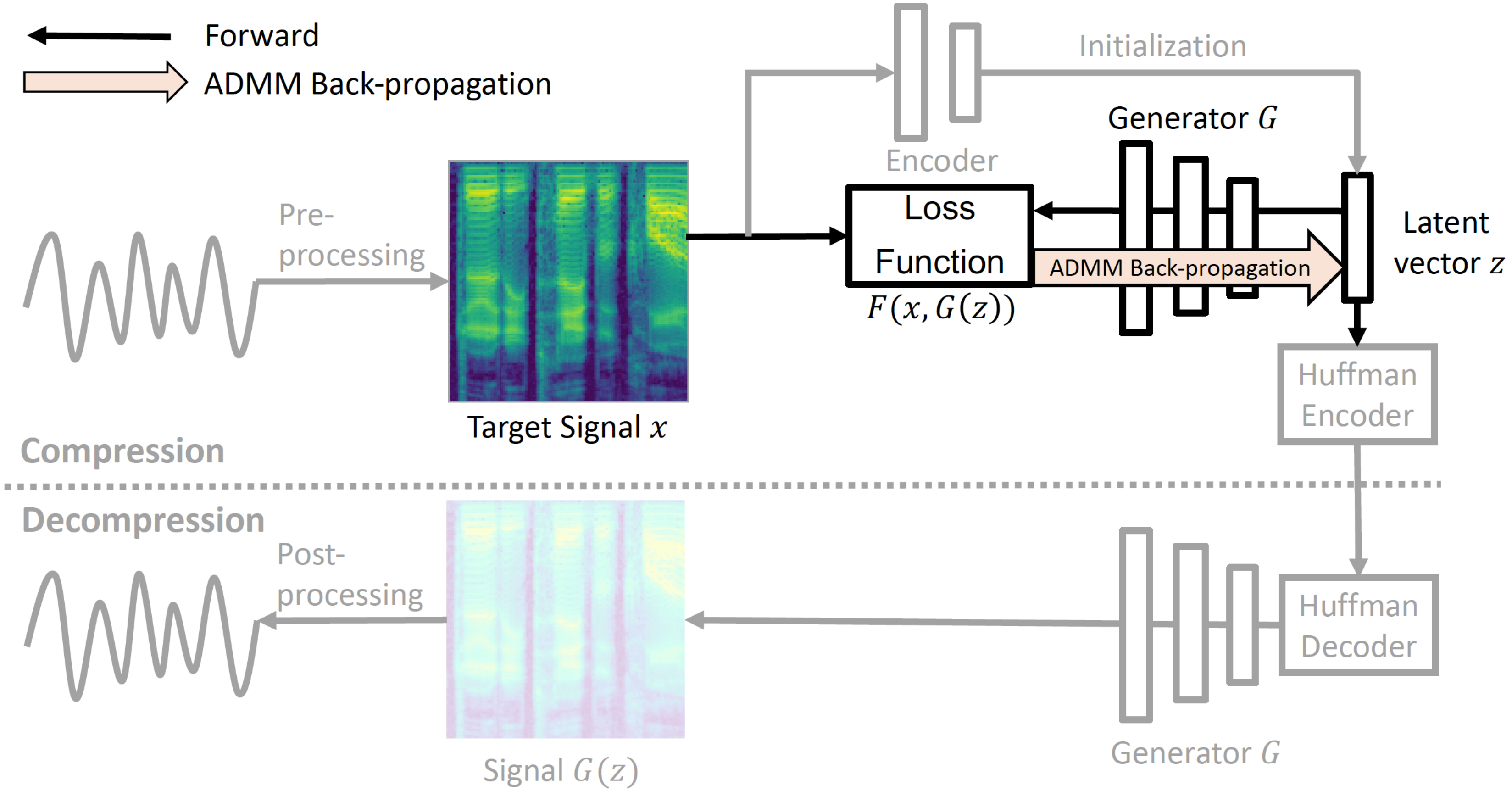
  Transform to mel-spectrogram and apply normalization

Step 2: Encode the signal



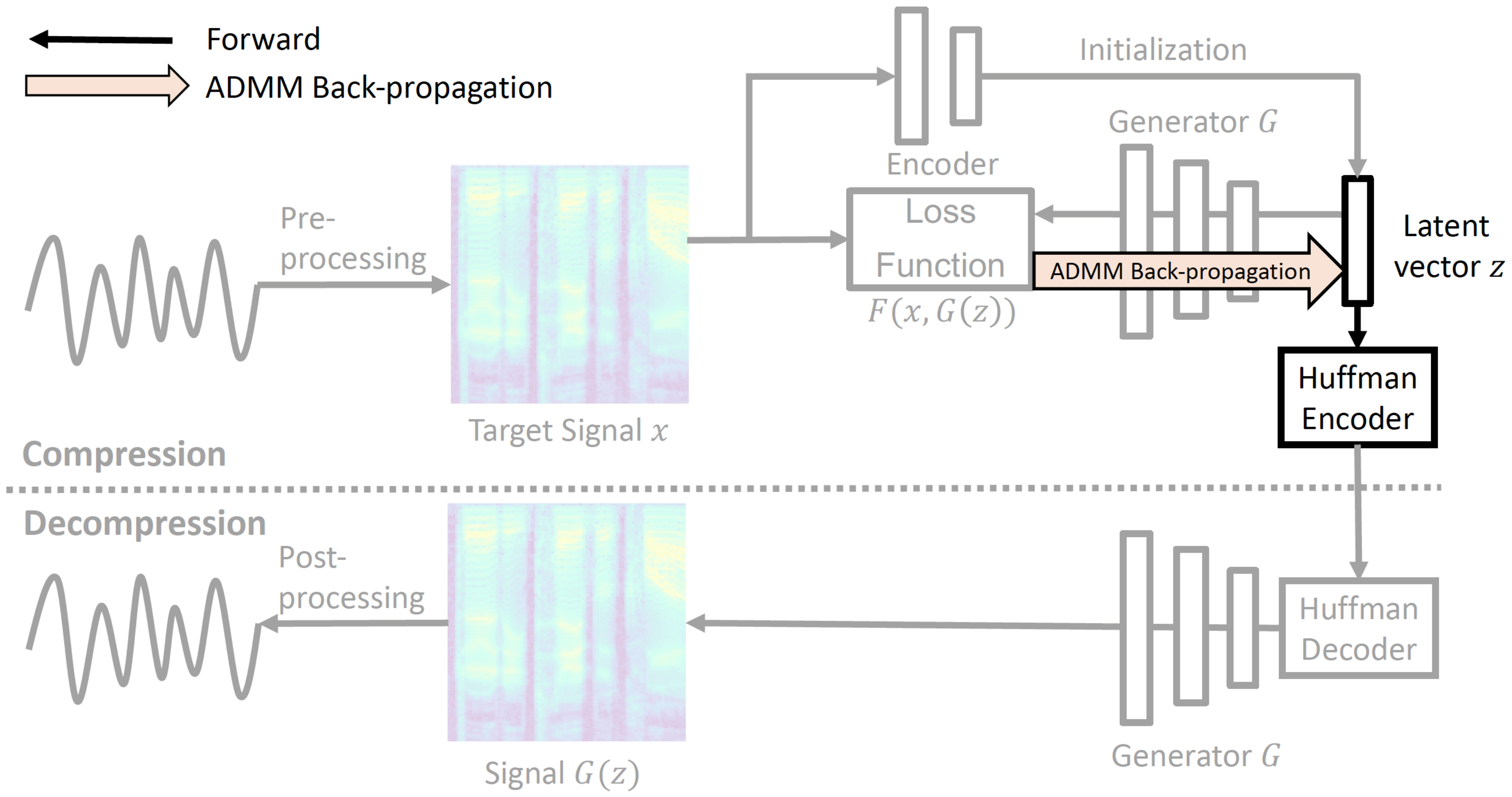
- Encode the target signal x to the latent vector z with an encoder Neural Network

Step 3: Optimize the latent vector



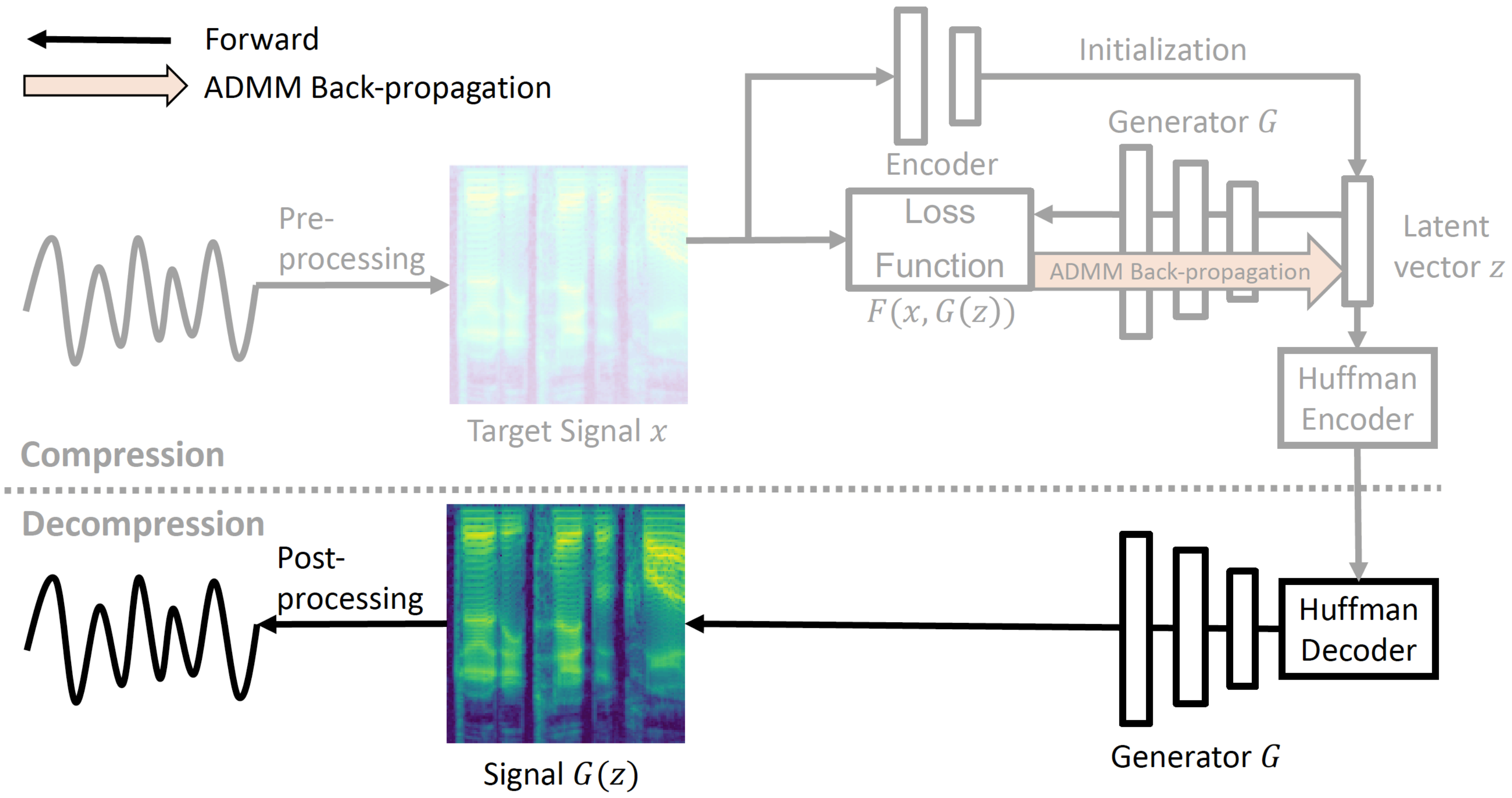
- Update the latent vector $\mathbf{z}$ via the back-propagation through the generator G
  - Compute the gradient $\boldsymbol{\partial F(x, G(z))/\partial z}$ for each iteration
  - Obtain the optimal latent vector $\tilde{\mathbf{z}}$ that minimizes the loss function
- The weights of GAN unchanged during signal compression & decompression

Step 4: Quantization and entropy coding



- Apply ADMM to quantize the latent vector $\tilde{z}$ during back propagation
- Encode the quantized result with entropy coding

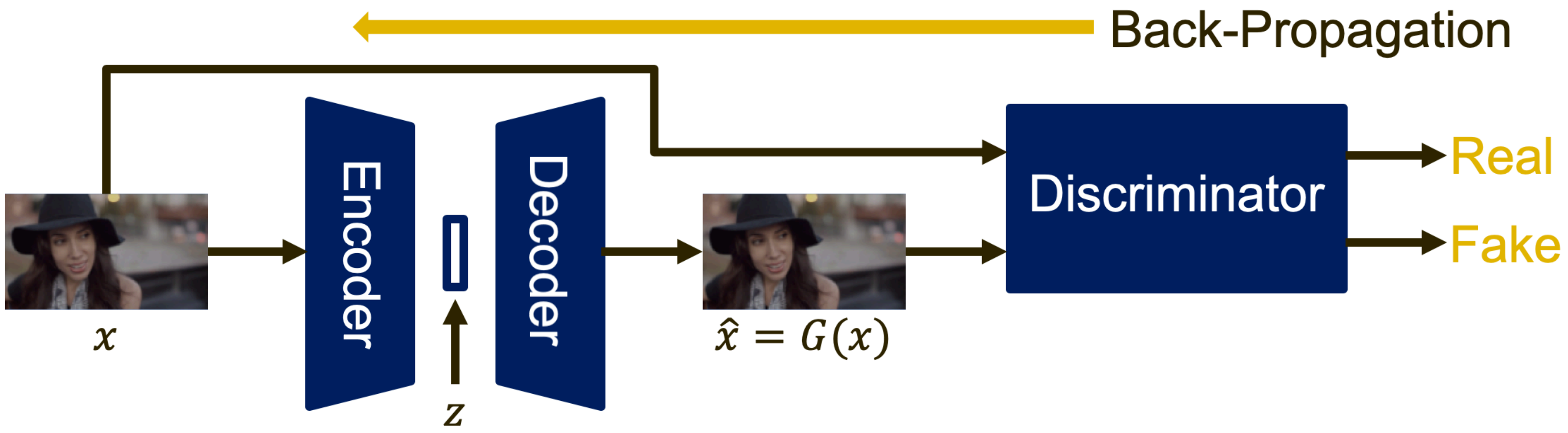
Step 5: Signal decompression and reconstruction



- Obtain the decompressed signal $G(\tilde{z})$ by feeding $\tilde{z}$ to generator G
- Reconstruct the signal by post-processing the signal $G(\tilde{z})$

Methodology: Training GAN

- Step 1. Train the GAN $(\boldsymbol{E}, \boldsymbol{G}, \boldsymbol{D})$ with unquantized (floating point) values
  - Adversarially train Generator (G) and discriminator (D)
  - Cascade an encoder by the generator to form an auto-encoder structure
  - Train the encoder to learn a mapping from the signal to a latent space vector
- Step 2. Train a GAN with quantized input
  - Regularize the latent vector to quantized input
  - Retrain generator and discriminator with regularized latent vectors



Loss function:

$$\min_{E,G} \max_{D} \mathbb{E}[\log(D(x))] + \mathbb{E}\left[\log(1 - \left(D\left(G(z)\right)\right)\right] + \lambda \cdot \mathbb{E}[d(x, G(z))]$$

Methodology: ADMM quantization

- Alternating direction method of multipliers (ADMM) quantization
  - ADMM is a divide-and-conquer optimization algorithm
  - Describe the problem of quantization as:

$$\min_{\{Z\}} \; f(\{Z\})$$
$$subject\ to \;\; Z \in S$$
$$where\ f(\{Z\})\ is\ the\ loss\ function\ , the\ set\ S\ is\ the\ quantized\ space$$

  - To apply ADMM for the above optimization problem, define indicator function:

$$g(Z) = \begin{cases} 0 & if \;\; Z \in S \\ +\infty & otherwise \end{cases}$$

  - Rewrite the problem with incorporate auxiliary variables R

$$\min_{\{Z\}} \; f(\{Z\}) + g(R)$$
$$subject\ to \;\; Z = R$$

Methodology: ADMM quantization

- Alternating direction method of multipliers (ADMM) quantization
  - Through application of the augmented Lagrangian, ADMM decomposes the problem to two subproblems
  - The first is minimizing the loss function of the original DNN with an additional L2 regularization term

$$U^k := U^{k-1} + Z^k - R^k$$

$$\min_{\{Z\}} \; f(\{Z\}) + \frac{\rho}{2} \cdot \left\| Z - R^k + U^k \right\|_2^2$$

$$\textit{where } U^k \textit{ is the dual variable updated in each iteration}$$

  - The second one can be optimally and analytically solved

$$\min_{\{R\}} \; g(R) + \frac{\rho}{2} \cdot \left\| Z^{k+1} - R + U^k \right\|_2^2$$

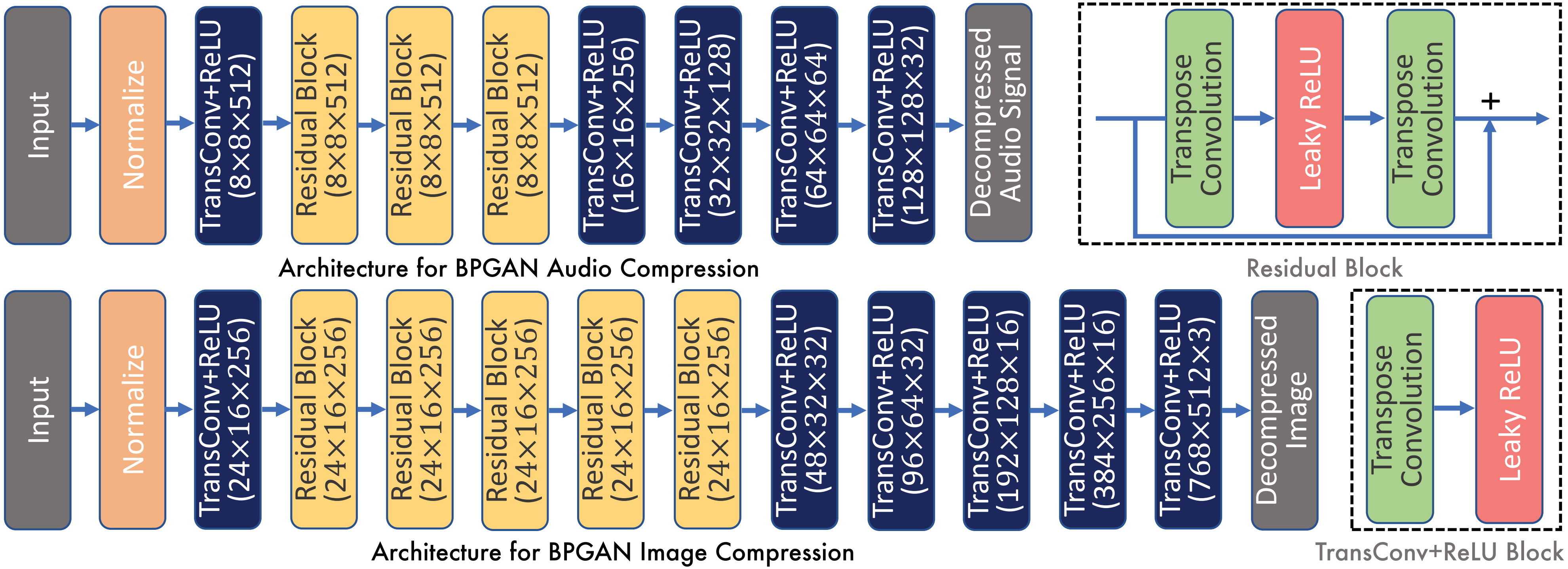
$$\text{Solution:} \;\; R^{k+1} := \Pi_S(Z^{k+1} + U^k)$$

$$\textit{where } \Pi_S(\cdot) \textit{ is Euclidean projection of } Z^{k+1} + U^k \textit{ onto the set } S$$

  - Those subproblems could be solved by updating $\mathbf{Z}$ and $\mathbf{R}$ iteratively
  - The optimal latent vector could be obtained by retraining and quantizing the latent vector iteratively

Network architecture

- Generator Network Topology



Architecture for BPGAN Audio Compression

Residual Block

Architecture for BPGAN Image Compression

TransConv+ReLU Block

- Discriminator Network

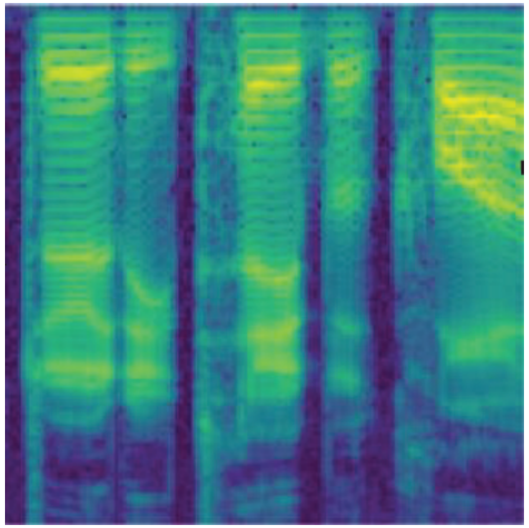
  Contains 5/8 (Speech/Image) convolutional layer

- Encoder Network

  Contains 5/9 (Speech/Image) convolutional layer

Dataset

- Open Images Dataset V5 (Image compression)
  Containing 9M images with 600 classes
- Kodak Dataset (Image compression)
  Well-known image compression dataset
- TIMIT dataset (Speech compression)
  Containing 6300 sentences spoken by 630 speakers from 8 major dialect regions



Audio signal

Image signal

Result and evaluation: Comparison

Image Methods	Bitrate (bpp)	PSNR	MS-SSIM	ImageNet Top-1 error%	ImageNet Top-5 error%	
Original	24	-	-	23.7	6.8	
BPGAN	**0.286**	**32.9**	**0.968**	**23.7**	**6.8**	
GAN based [1]	0.305	28.2	0.922	26.0	7.9	
JPEG	0.306	26.9	0.864	42.5	16.6	
BPG	0.298	32.3	0.961	25.8	7.4	

- Compression tested with different datasets unused for training
- Achieves state-of-the-art performance for both image/speech compression
  - Obtain high quality decompressed signal with extreme low bitrate

Result and evaluation: Comparison

Image Methods	Bitrate (bpp)	PSNR	MS-SSIM	ImageNet Top-1 error%	ImageNet Top-5 error%	
Original	24	-	-	23.7	6.8	
BPGAN	**0.286**	**32.9**	**0.968**	**23.7**	**6.8**	
GAN based [1]	0.305	28.2	0.922	26.0	7.9	
JPEG	0.306	26.9	0.864	42.5	16.6	
BPG	0.298	32.3	0.961	25.8	7.4	
Speech Methods	**Bitrate (bps)**	**PESQ**	**MUSHRA**	**Kaldi PER%**	**MLP PER%**	**LSTM PER%**
Original	256k	4.50	95.0	18.7	18.6	15.4
BPGAN	**2k**	**3.25**	**64.1**	**20.9**	**20.8**	**18.6**
CELP	4k	2.54	32.0	28.2	27.6	27.3
CELP	8k	3.39	59.4	23.0	23.6	21.2
Opus	9k	3.47	79.3	22.7	23.7	21.2
AMR	6.6k	3.36	58.9	22.6	23.6	22.3

- Compression tested with different datasets unused for training
- Achieves state-of-the-art performance for both image/speech compression
  - Obtain high quality decompressed signal with extreme low bitrate

Result and evaluation: Visualization

- BPGAN achieves state-of-the-art performance for image compression task

  Using ADMM technique to quantize the input latent vectors can achieve nearly no performance degradation with 6-bit quantization for each element

[1] Eirikur Agustsson et al., "Generative adversarial networks for extreme learned image compression," arXiv:1804.02958, 2018.

Result and evaluation: Speech compression

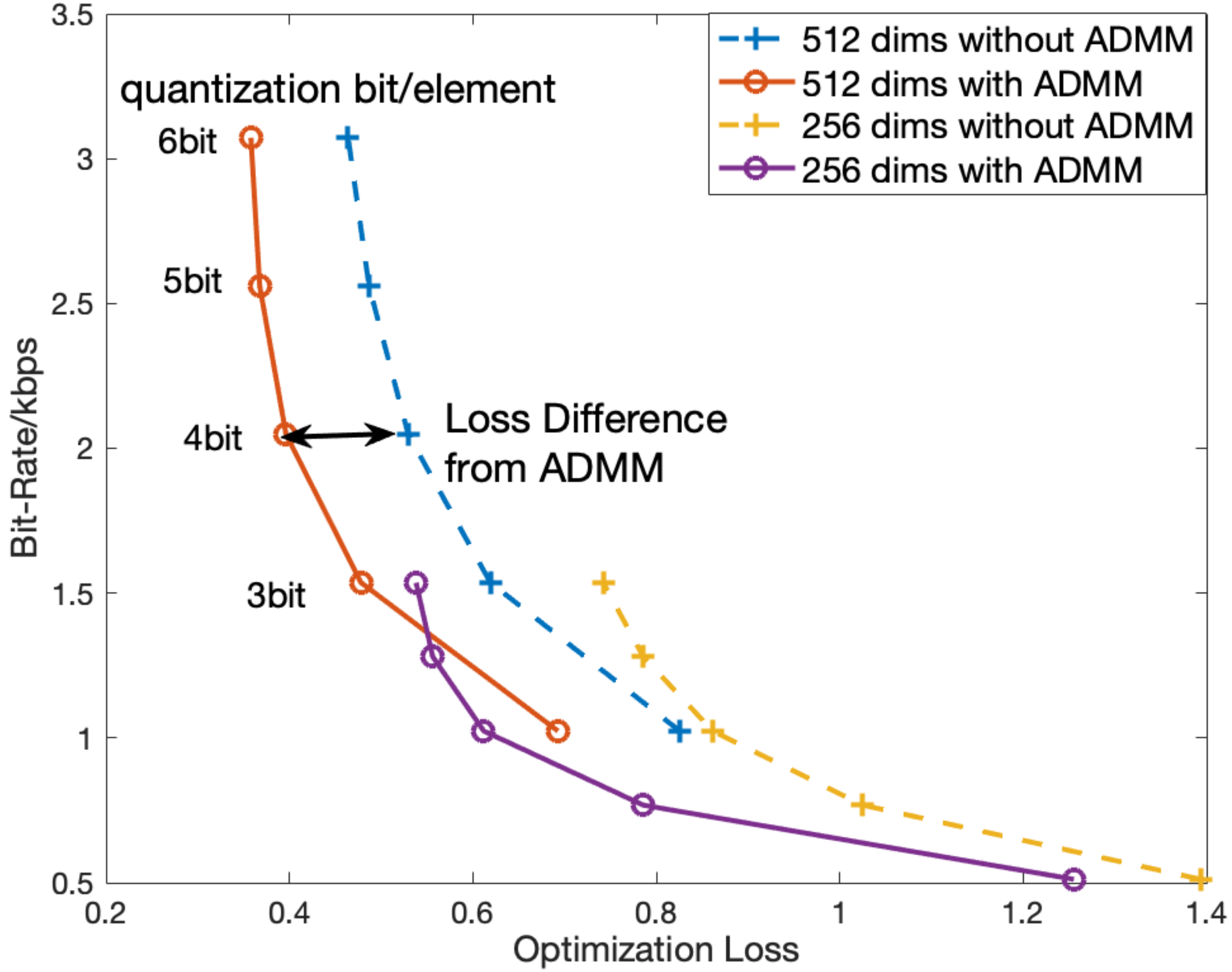
- BPGAN achieves state-of-the-art performance for speech compression

Original Audio (256kbps)   Compressed Audio (2kbps)

- Don't ask me to carry an oily rag like that.
- Don’t ask me to carry an oily rag like that “In another tune”.
- Materials: ceramic modeling clay: red, white or buff.
- Here, he is, quite persuasively, the very embodiment of meanness and slyness.
- Sometimes, he coincided with my father's being at home.

Result and evaluation: Quantization

- ADMM quantization outperforms regular uniform quantization



Summary

- BPGAN: New GAN-based unified signal compression framework
  - Applicable to both image and speech signal
  - Achieves variable bitrate vs. quality tradeoff for compressed signal
  - Outperform state-of-the-art compression algorithms

Thank you!