Universal Rate-Distortion-Classification Representations for Lossy Compression

Nam Nguyen[1]

Collaborators: Thuan Nguyen[2], Thinh Nguyen[1], Bella Bose[1]

[1]Oregon State University
[2]East Tennessee State University

IEEE Information Theory Workshop 2025
Sydney, Australia
December 4, 2025

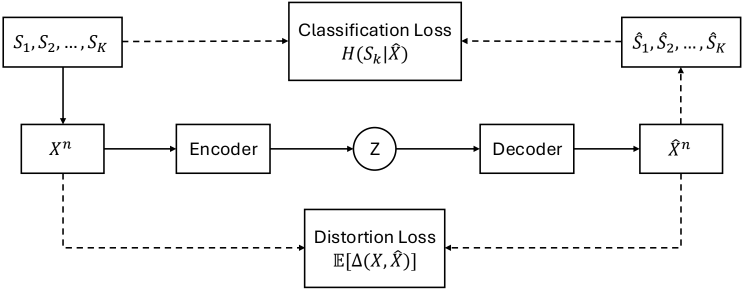
Background: Deep Learning + Lossy Compression

Why deep learning for lossy compression?

- Requires retraining per dataset, but provides major benefits:
- Higher compression efficiency
- Better perceptual quality and realism
- Supports multi-task learning for downstream applications

Figure: Degradation of JPEG. As the rate decreases, the result is pixelated.

Background: Task-oriented Lossy Compression



Source and Target labels:

- Source: $X \sim p_X(x)$.
- Target labels: $S_1, \cdots, S_K \sim p_S(s_1, \cdots, s_K)$, where $p_{X,S}(x, s_1, \cdots, s_K)$.

Lossy Compression: $X_1, X_2, \cdots, X_n \overset{\text{i.i.d}}{\sim} p_X(x)$.

- Encoder: $f : \mathcal{X}^n \mapsto \{1, 2, \cdots, 2^{nR}\}$ maps the source X^n to a message Z.
- Decoder: $g : \{1, 2, \cdots, 2^{nR}\} \mapsto \hat{\mathcal{X}}^n$ reproduces data $\hat{X}^n$ to satisfy *task-oriented demands* of downstream applications.

Background: Rate-Distortion-Classification (RDC) Function

The rate-distortion-classification function:

- Distortion between symbols: $\mathbb{E}[\Delta(X, \hat{X}_{D,C})] \geq 0$, with equality iff $X = \hat{X}$
- Classification constraint [Wang et al. 2024]: the uncertainty of classification variables S_k given $\hat{X}$

$$H(S_k|\hat{X}) \leq C_k, \qquad \forall k \in [K].$$

$$R(D,C) = \min_{p_{\hat{X}|X}} \quad I(X;\hat{X}) \tag{1a}$$

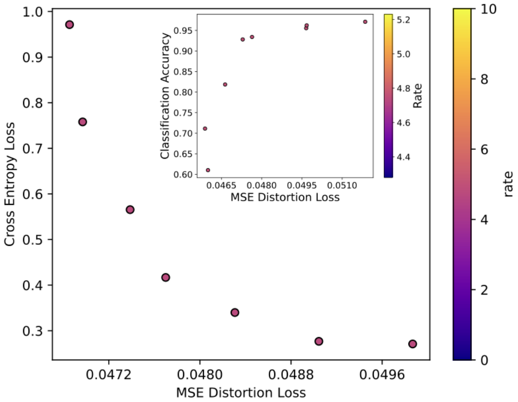
$$\text{s.t.} \quad \mathbb{E}[\Delta(X, \hat{X}_{D,C})] \leq D, \tag{1b}$$

$$H(S|\hat{X}) \leq C. \tag{1c}$$

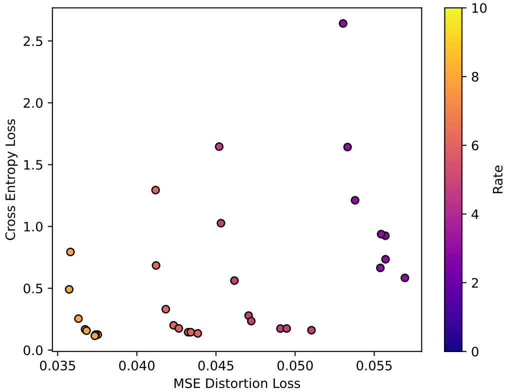
where S is a classification variable.

Background: Rate-Distortion-Classification Tradeoff

- Tradeoff between distortion and classification with given rate

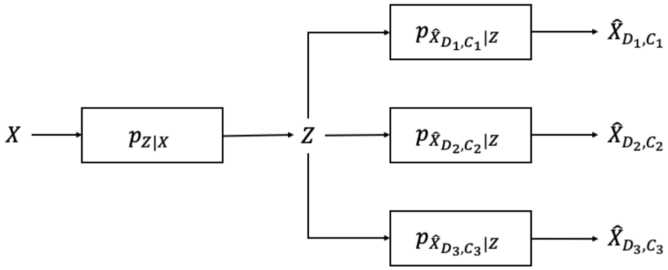


(a) The RDC curve on MNIST.



(b) The RDC curves at multiple rates on MNIST.

Universal Representations: Motivation



Motivation for Universal Representations

- $R(D, C)$ corresponds to designing an encoder-decoder pair for each (D, C) tradeoff point (i.e., variable-encoder variable-decoder)
- **Main question:** *Is it possible to design/reuse an encoder for multiple tradeoff points?*

Universal Representation: Definition

The Universal Rate-Distortion-Classification Function

- Let $X \sim p_X$ and Θ be an arbitrary set of (D, C) pairs
- **Idea:** find a **reprentation** Z which can be transformed into **reconstruction** $\hat{X}_{D,C}$ to meet constraints $(D, C) \in \Theta$

$$R(\Theta) = \inf_{p_{Z|X}} I(X; Z), \tag{2}$$

where

$$\mathbb{E}[\Delta(X, \hat{X}_{D,C})] \leq D \quad \text{and} \quad H(S|\hat{X}_{D,C}) \leq C.$$

Universal Representation: Rate Penalty

The *rate penalty* incurred by meeting *all* constraints in Θ with **fixed encoder** is defined as:

$$A(\Theta) = R(\Theta) - \sup_{(D,C)\in\Theta} R(D,C), \tag{3}$$

- $\sup_{(D,C)\in\Theta} R(D,C)$ is used for satisfying the stringiest individual constraints
- Ideally, $A(\Theta) = 0$ for each R, meaning a **single encoder** suffices for the entire tradeoff

Let $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ be a Gaussian source and $S \sim \mathcal{N}(\mu_S, \sigma_S^2)$ be a classification variable with $\mathrm{Cov}(X, S) = \theta_1$. Let Θ be any non-empty set of constraint pairs (D, C). Then,

$$A(\Theta) = 0. \tag{4}$$

Application with Deep Learning: Introduction

Task-oriented Lossy Compression

- Theoretical results assume the source distribution is known
- In practice, these distributions must be inferred from data
- **Question:** *Can we use existing architectures to achieve approximate universality in practice?*

DL-based Lossy Compression: Schematic

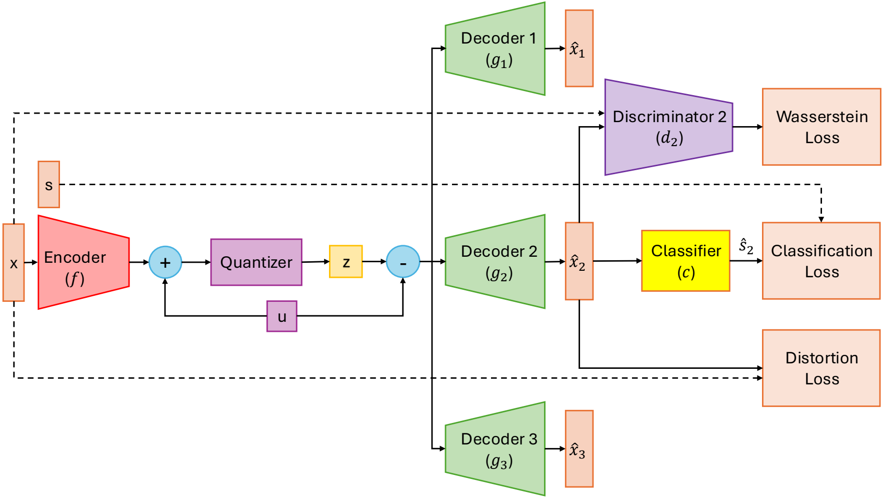


Figure: An illustration of the universal RDC scheme.

DL-based Lossy Compression: Algorithm

Phase 1: Training initial conventional model [Blau et al. 2019]

- Start with untrained encoder f, decoder g, discriminator d, classifier c.
- (f, g) form an autoencoder and (g, h) form a GAN: so decoder is also a generator
- Alternate between training (f, g) and training (h, c)
- Objective with hyperparameter $(\lambda_d, \lambda_c, \lambda_p)$, $\hat{X} = g(f(X))$:

$$\mathcal{L} = \lambda_d \underbrace{\mathbb{E}[\|X - \hat{X}\|^2]}_{\text{Distortion loss}} + \lambda_c \underbrace{\mathrm{CE}(S, \hat{S})}_{\text{Cross-entropy loss}} + \lambda_p \underbrace{W_1(p_X, p_{\hat{X}})}_{\text{Wasserstein loss}}. \tag{5}$$

Phase 2: Training (approximately) universal model

- Use f from Phase 1 with frozen weights, initialize new decoder g_1, discriminator h_1, classfier c_1
- Repeat procedure
- Loss function: same as (5), with different $(\lambda_d, \lambda_c, \lambda_p)$ tradeoff

DL-based Lossy Compression: Compression and Stochasticity

Compression

- Use $\tanh$ activation so output of encoder lies in $(-1, +1)^d$
- Choose L uniformly spaced quantization centers. Rate upper bounded by $d \log L$
- Use soft gradients [Agustsson et al. 2019] to backpropagate

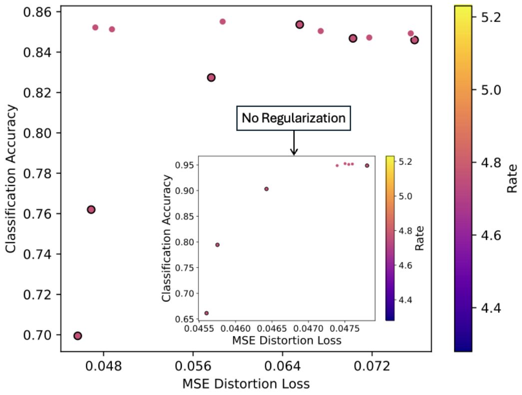
Stochasticity

- GANs require stochasticity to train
- Use *universal/dithered quantization* [Gray et al. 1993; Ziv 1985]: assume sender and receiver both have access to $u \sim \mathrm{Unif}\left[-\frac{1}{L-1}, +\frac{1}{L-1}\right]^d$. The sender computes

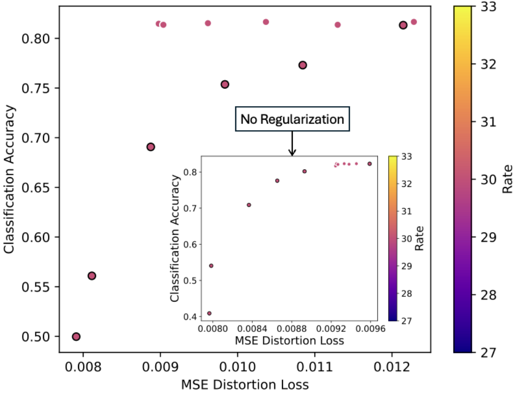
$$z = \arg\min_{c \in \mathcal{C}} \|f(x) + u - c\|$$

and gives z to receiver. Receiver reconstructs image by passing $z - u$ through decoder.

DL-based Lossy Compression: MNIST/SVHN



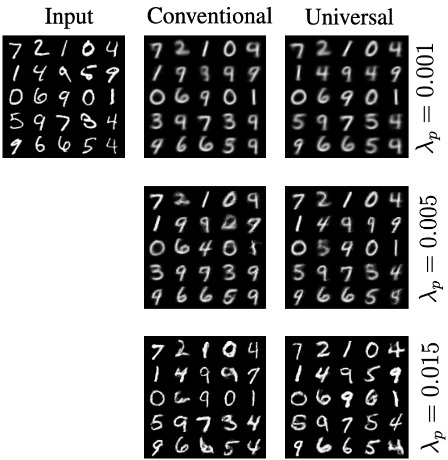
(a) RDC curve on MNIST.



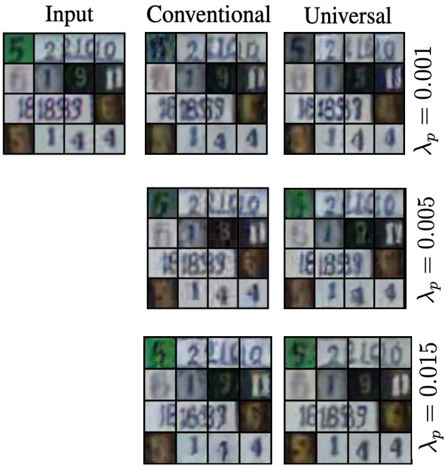
(b) RDC curve on SVHN.

- Bolded points denote the conventional models
- Unbolded points denote universal models

DL-based Lossy Compression: MNIST/SVHN



(a) Decompressed outputs on MNIST.



(b) Decompressed outputs on SVHN.