DEEP LEARNING APPROACHES TO LOW-LEVEL VISION PROBLEMS

DEEP LEARNING APPROACHES TO LOW-LEVEL VISION PROBLEMS

BY

HUAN LIU, B.Eng.,

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL & COMPUTER ENGINEERING
AND THE SCHOOL OF GRADUATE STUDIES
OF MCMASTER UNIVERSITY
IN PARTIAL FULFILMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Doctor of Philosophy (2022)	McMaster University
(Electrical & Computer Engineering)	Hamilton, Ontario, Canada

TITLE:	Deep Learning Approaches to Low-Level Vision Problems
AUTHOR:	Huan Liu B.Eng., (Communication Engineering) University of Electronic Science and Technology of China, Chengdu, China
SUPERVISOR:	Dr. Jun Chen
NUMBER OF PAGES:	xx, 182

Abstract

Recent years have witnessed tremendous success in using deep learning approaches to handle low-level vision problems. Most of the deep learning based methods address the low-level vision problem by training a neural network to approximate the mapping from the inputs to the desired ground truths. However, directly learning this mapping is usually difficult and cannot achieve ideal performance. Besides, under the setting of unsupervised learning, the general deep learning approach cannot be used. In this thesis, we investigate and address several problems in low-level vision using the proposed approaches.

To learn a better mapping using the existing data, an indirect domain shift mechanism is proposed to add explicit constraints inside the neural network for single image dehazing. This allows the neural network to be optimized across several identified neighbours, resulting in a better performance.

Despite the success of the proposed approaches in learning an improved mapping from the inputs to the targets, three problems of unsupervised learning is also investigated. For unsupervised monocular depth estimation, a teacher-student network is introduced to strategically integrate both supervised and unsupervised learning benefits. The teacher network is formed by learning under the binocular depth estimation setting, and the student network is constructed as the primary network for monocular depth estimation. In observing that the performance of the teacher network is far better than that of the student network, a knowledge distillation approach is proposed to help improve the mapping learned by the student. For single image dehazing, the current network cannot handle different types of haze patterns as it is trained on a particular dataset. The problem is formulated as a multi-domain dehazing problem. To address this issue, a test-time training approach is proposed to leverage a helper network in assisting the dehazing network adapting to a particular domain using self-supervision.

In lossy compression systems, the target distribution can be different from that of the source and ground truths are not available for reference. Thus, the objective is to transform the source to target under a rate constraint, which generalizes the optimal transport. To address this problem, theoretical analyses on the trade-off between compression rate and minimal achievable distortion are studied under the cases with and without common randomness. A deep learning approach is also developed using our theoretical results for addressing super-resolution and denoising tasks.

Extensive experiments and analyses have been conducted to prove the effectiveness of the proposed deep learning based methods in handling the problems in low-level vision.

To my beloved family.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my supervisor Dr. Jun Chen for his continuous support throughout my Ph.D study, for his patience, enthusiasm, and immense knowledge. Without his invaluable guidance and dedication, none of the thesis work would have been materialized. I could not have imagined having a better supervisor for my Ph.D study.

I would like to thank my committee members Dr. Shahram Shirani and Dr. Sorina Dumitrescu for giving me guidance and support over the years. I would also like to thank Dr. Zhou Wang for being my external examiner.

My gratitude also goes to my friends and fellow collaborators, especially to Yangyi, Siyao, Jingjing, Botao, Minghan, Yankun, Liangyan, Zijun, Kangdi, Peiyao, Sihong and George, for generously sharing their advice and knowledge. I appreciate the friendship built among us and I wholeheartedly believe that our friendships will last even beyond this journey.

Last but not least, I would like to thank my families for their unwavering supports throughout my PhD journey. To them I dedicate this thesis.

Contents

List of Figures

List of Tables

Abbreviations

AI	Artificial Intelligence
ASM	Atmospheric Scattering Model
BPG	Better Portable Graphics
CNN	Convolutional Neural Network
DL	Deep Learning
DNN	Deep Neural Network
GT	Ground Truth
GAN	Generative Adversarial Network
GPU	Graphics Processing Units
HR	High-resolution
IDS	Indirect Domain Shift
JPEG	Joint Photographic Experts Group
LR	Low-resolution

ML	Machine Learning
MSE	Mean Squared Error
MDL	Multi-domain Learning
MAML	Model Agnostic Meta-learning
OOD	Out of Domain
PSNR	Peak Signal-to-noise Ratio
RNN	Recurrent Neural network
SGD	Stochastic Gradient Decent
SSIM	Structural Similarity Index
SOTA	State of the Art
WNNM	Weighted Nuclear Norm Minimization

Chapter 1

Introduction

1.1 Deep Learning and Low-level Vision

Deep learning as a machine learning method has attracted tremendous attention in recent years. Despite the fact that deep learning is a sub-field of machine learning, it distinguishes itself in the way how learning is carried out. Machine learning usually depends on human intervention to process data, while deep learning can automate the feature process and approximates the desired mapping for a large-scaled dataset. To be specific, deep learning is typically realized using convolutional neural networks (CNNs) that have three or more convolutional layers. Each layer acts as a non-linear mapping to transform the input data into a compact and composite representation. The mapping is not formed by human experts but automatically discovered using optimization techniques based on backpropagation. Deep learning algorithms can be learned in three different ways, i.e., supervised learning, semi-supervised learning and unsupervised learning. Supervised learning can be performed if the dataset is labeled and can provide input-target pairs. Unsupervised learning can use unstructured data and automatically determine the desired features. Semi-supervised learning lies between supervised and unsupervised learning. It requires both labeled and unlabeled data for training. In this thesis, we provide several approaches, mainly in the context of supervised and unsupervised learning.

Low-level vision as a sub-filed of computer vision has witnessed significant progress because of the success use of deep learning. Generally, computer vision can be categorized as low-level, mid-level, and high-level. The low-level vision includes image restoration, depth estimation and edge detection. It usually concerns the extraction of image properties; mid-level vision mainly focuses on how to integrate the image properties into perceptual organizations, such as semantic segmentation and structure of motion; high-level vision requires performing analysis on image properties and perceptual organizations, such as image recognition, image captioning and visual question answering. The problems to be addressed in this thesis all belong to the low-level vision category.

1.2 Supervised Learning for Low-level Vision

Supervised learning aims at approximating a function from the inputs to the desired outputs based on example pairs. The example pairs are built according to the tasks. For example, in single image dehazing, one can form such pairs by collecting and synthesizing hazy and haze-free images that respectively act as input and target. With adequate examples, it is possible to train a neural network end-to-end to learn a mapping from inputs to targets. The mapping is then fixed and can be used to process newly collected data.

Such a learning strategy is especially desired in addressing low-level vision tasks.

As is known, low-level vision tasks are usually highly ill-posed. Classic approaches are usually designed using some analytical priors, such as edge prior [124] for image super-resolution and dark channel prior [48] for image dehazing. Considering the fact that these priors are designed under the observation of particular image properties, they usually generalize poorly and cannot handle complex scenarios. In favor of the remarkable ability of deep neural networks, most low-level vision problems have been addressed using supervised learning in recent years. It eliminates the burden of finding analytical priors for a specific task. In contrast, a better-performed approach can be designed by the following three steps. First, one should construct datasets by synthesizing realistic data pairs or collecting data pairs in the real world; second, one should construct deep neural networks that can process on such data pairs; finally, one can use the datasets to train the deep neural networks for achieving satisfactory performance. Nowadays, many deep learning based approaches are designed in this way. Their success indicates the effectiveness of learning task-specific mapping on examples.

However, deep neural networks are trained using an optimization process that requires loss functions to calculate the inference error. The selection of different loss functions can significantly influence the model's performance. The typical loss functions used in low-level vision tasks are L1 loss, MSE and SSIM [136]. In observing that using proper loss functions can boost the performance of the neural networks, recently, many novel loss functions have been proposed. Knowledge distillation loss [52] introduces a teacher-student scheme for boosting the performance of small-scaled networks; contrastive loss [50, 51, 96] is proposed to ensure the output images are closer to the targets and far away from the inputs; GAN loss [3, 99] is also widely used in low-level vision for achieving perceptually pleasing results. Using the above loss functions in the training process of neural networks can further improve inference accuracy.

In observing the fact that the current dehazing network cannot deliver satisfactory results on single image dehazing, chapter 2 provides an indirect domain shift approach to accurately estimate clear images under supervised learning. However, it can be very costly and in some cases impossible to obtain pixel-wise ground truth annotations for supervised training. This naturally leads to the topic of next subsection – unsupervised learning.

1.3 Unsupervised Learning for Low-level Vision

Unsupervised learning aims to approximate a function that can map the input to the desired form of output without explicitly set targets. In other words, unlike supervised learning that usually requires ground truth in the training process, unsupervised learning does not have access to the paired ground truth data. Usually, the construction of a paired dataset is extremely costly and has become a major hurdle for developing advanced approaches for low-level vision tasks. As a consequence, unsupervised learning is attracting extensive attention.

Unsupervised approaches have shown their superb performance in high-level vision tasks, such as representation learning [50]. However, achieving unsupervised learning in low-level vision is extremely hard. Unlike high-level vision tasks that require a neural network to extract representations as very low dimensional vectors, low-level vision tasks require a global understanding of image content and local restoration of fine texture details. Without the supervision from ground truths, it is hard to set a training objective that can guide the network to acquire the ability to produce desired outputs. Surprisingly, despite the difficulty, many attempts have been made to use unsupervised learning in handling low-level vision tasks.

In summary, unsupervised learning is usually adopted in low-level vision in three ways. 1) Using strong physical prior to form supervision with additional information. This kind of work usually requires a strong physical model that can learn to predict desired output with auxiliary information, such as unsupervised monocular depth estimation [19, 41, 42]. Although the unsupervised monocular depth estimation does not need depth maps as supervision, it requires predicting the matching relations (termed disparity map) of inputs with the other view of the scene. And the disparity map can be transformed to scene depth given a few already known parameters, such as the distance between two cameras and focal length. 2) using unsupervised learning as a helper to boost the performance of a model that is learned in a supervised manner. This line of research aims to exploit the possibility of improving the performance of current supervised learning approaches, such as test-time training [23]. Usually, a model trained under supervised learning should be fixed for inference during the test time. It is not able to further adapt to a particular scene that has not been exposed in training. Ideally, a network should utilize the internal information within a specific image. Therefore, test-time training is proposed to enable the network quickly adapt to a particular image with self-supervision. For example, reference [23] constructs an auxiliary task, i.e., image reconstruction, during the supervised training phase. Then the image reconstruction task can provide supervision in an unsupervised manner during testing. The update of the network in test time can further boost its performance to produce better deblurring results. 3) Pure unsupervised learning.

This kind of work explores a pure unsupervised method for low-level vision tasks, such as image denoising [65, 129]. Some of the methods assume to only have access to degraded input images. [65] applies basic statistical reasoning to signal reconstruction. [129] proposes to recover a clean image using the early stopping strategy as the authors find that a convolutional image generator can capture most image statistics instead of the learning process. The other works require an statistical likelihood model of the corruption or target distribution, such as PULSE [82], and Cycle-GAN [160].

In this thesis, three different methods are proposed following the above three unsupervised learning strategies and providing innovations in each category. Chapter 3 provides new insight into unsupervised monocular depth estimation. The proposed method turns the unsupervised problem into a supervised counterpart with a teacher-student structure. Chapter 4 presents a novel approach to handling multi-domain learning problem using unsupervised test-time training for single image dehazing. Finally, chapter 5 provides a solution to unsupervised image denoising and super-resolution in a compression system using optimal transport.

1.4 Contributions and Thesis Organization

The thesis is in a *sandwich thesis format* following the terms and regulations of McMaster University. It consists of four published/unpublished articles that address low-level vision problems using deep learning approaches. The contributions to each article are listed in the preface of Chapter 2, Chapter 3, Chapter 4 and Chapter 5. Here is the reference information for the four articles:

- Huan Liu and Jun Chen. ”Indirect Domain Shift for Single Image Dehazing”.

IEEE Access. 2021 Sep 3;9:122959-70.

- Huan Liu, Junsong Yun, Chen Wang, and Jun Chen. ”Pseudo Supervised Monocular Depth Estimation with Teacher-Student Network”. arXiv preprint arXiv:2110.11545. 2021 Oct 22.
- Huan Liu, Zijun Wu, Liangyan Li, Sadaf Salehkalaibar, Jun Chen, and Keyan Wang. ”Towards Multi-domain Single Image Dehazing via Test-time Training”. Accepted by IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022.
- Huan Liu, George Zhang, Jun Chen, and Ashish Khisti. ”Lossy Compression with Distribution Shift as Entropy Constrained Optimal Transport. Accepted by International Conference on Learning Representations 2022.

The rest of the thesis is organized as follows:

- **Chapter 2** provides the detailed indirect domain shift approach for handling single image dehazing using supervised learning.
- **Chapter 3** provides the detailed teacher-student scheme for boosting the performance of the current unsupervised monocular depth estimation approach.
- **Chapter 4** provides a multi-domain formulation of the current single image dehazing task and gives a test-time training solution.
- **Chapter 5** formulates an optimal transport problem under lossy compression and addresses it using a pure unsupervised approach.

- **Chapter 6** provides the conclusion of this thesis and the discussion of future works.

The following chapter is a reproduction of an Institute of Electrical and Electronics Engineers (IEEE) copyrighted, published paper:

Huan Liu and Jun Chen. ”Indirect Domain Shift for Single Image Dehazing”. IEEE Access. 2021 Sep 3;9:122959-70.

Contribution Declaration: Huan Liu (the author of this thesis) is the first author and main contributor of this article. He proposed the method, conducted experiments and composed the article. Prof. Jun Chen is the supervisor of Huan Liu.

Chapter 2

Indirect Domain Shift for Single Image Dehazing

2.1 Abstract

Despite their remarkable expressibility, convolution neural networks (CNNs) still fall short of delivering satisfactory results on single image dehazing, especially in terms of faithful recovery of fine texture details. In this chapter, we argue that the inadequacy of conventional CNN-based dehazing methods can be attributed to the fact that the domain of hazy images is too far away from that of clear images, rendering it difficult to train a CNN for learning direct domain shift through an end-to-end manner and recovering texture details simultaneously. To address this issue, we propose to add explicit constraints inside a deep CNN model to guide the restoration process. In contrast to direct learning, the proposed mechanism shifts and narrows the candidate region for the estimation output via multiple confident neighborhoods. Therefore, it is capable of consolidating the expressibility of different architectures, resulting in

a more accurate indirect domain shift (IDS) from the hazy images to that of clear images. We also propose two different training schemes, including hard IDS and soft IDS, which further reveal the effectiveness of the proposed method. Our extensive experimental results indicate that the dehazing method based on this mechanism dramatically outperforms the state-of-the-arts.

2.2 Introduction

Deep convolutional neural networks (CNNs) have been tremendously successful in many high-level computer vision tasks, e.g. image recognition [49, 60] and object detection [40, 106]. Although recent works have shown that it is also possible to learn an end-to-end CNN model for low-level vision tasks, e.g. image dehazing [17, 56], the resulting performance is still not completely satisfactory. For high-level vision tasks, it suffices to extract specific features and simply express them as very low dimensional vectors [60], which results in a relatively simple mapping. In contrast, low-level vision tasks require both global understanding of image content and local inference of texture details; as such, the associated mappings are more complicated.

One possible explanation for performance discrepancies on high-level and low-level vision tasks is as follows. For high-level vision tasks such as image recognition, a slight perturbation of the output tends to be inconsequential since the perturbed output is likely to get converted to the same one-hot vector and consequently the classification label remains unaffected. However, for low-level vision tasks such as image dehazing, any perturbation can potentially manifest in the final result, jeopardizing the image quality. From this point of view, despite the fact that a deep CNN can in principle approximate any function, it is still difficult to train an accurate mapping that lifts the input to the target domain in one shot, since the loss function is typically very close to zero in the neighborhood of the target image [73]. We argue that a different mechanism for domain shift is needed for image dehazing, which requires both memory and understanding of image contents.

To this end, we provide explicit guidance during model optimization to lead the domain shift path across several identified confident neighborhoods , resulting in the proposed framework shown in Figure 2.1. More specifically, instead of only imposing the loss function on the model output, we introduce multi-scale estimation, multi-branch diversity, and adversarial loss inside the model, thereby pulling the interim outputs to specific regions then merging them in the target domain; this yields an indirect but more accurate mapping. The contributions of this chapter include:

- By introducing loss functions inside a CNN model, we propose the framework of indirect domain shift (IDS) for image dehazing, which aggregates powerful expressibility of different architectures, i.e., multi-scale, multi-branch, and generator for lifting degraded images to the target domain indirectly.
- We provide theoretic justifications for IDS and show that it provides valuable guidance for network construction. (1) A multi-scale module takes the advantage of coarse-fine network to maintain global-local consistency. (2) A multi-branch architecture is adopted to enable precise inference of local details by providing diverse confident neighborhoods. (3) A FusionNet further improves the perceptual quality by informed 'imagination', rather than blindly pursuing a higher PSNR, as the multi-scale multi-branch structure has shifted degraded images close enough to the corresponding ground truth in terms of objective image quality metrics.

- It is demonstrated that IDS leads to remarkable performance improvements compared with the state-of-the-art algorithms.

2.3 Related Works

Image dehazing, which aims to recover a haze-free image from its hazy version, is a highly ill-posed restoration problem. The haze effect is often approximated using the atmospheric scattering model [89] given as follows:

$$\mathbf{I}(x) = \mathbf{J}(x)t(x) + \mathbf{A}(1 - t(x)), \tag{2.3.1}$$

where $\mathbf{I}(x)$, $\mathbf{J}(x)$, and $\mathbf{A}$ are the observed hazy image, clear scene radiance, and global atmospheric light, respectively. The scene transmission $t(x)$ describes the portion of light that is not scattered and reaches the camera. It can be expressed as $t(x) = e^{-\beta d(x)}$, where β is the medium extinction coefficient and $d(x)$ is the depth map of pixel x.

Based on this atmospheric scattering model [89], many strategies have been proposed by taking advantage of various prior knowledge. For example, the dark channel prior [48] assumes that in non-sky patches, at least one color channel has very low intensity. The color attenuation prior [161] assumes that the image saturation decreases sharply at hazy patches, so that the difference between brightness and saturation can be utilized to estimate the haze concentration. To address the weakness of DCP for the sky region, [111] proposes to separately deal with the non-sky region and the sky region using dark channel prior and luminance stretching. In [112], the authors come up with a new color channel method to remove atmospheric scattering for single image dehazing. The overall algorithm consists of atmospheric light calculation, transmission map estimation, radiance estimation and post enhancement. Furthermore, based on the assumption that a linear relationship exits in the minimum channel between hazy and haze-free images, a fast linear-transformation-based dehazing algorithm is introduced in [133].

Recently, data-driven approaches to image dehazing have received increasing attention. [107] and [17] propose to use CNN for medium transmission estimation, which is further leveraged to recover the haze-free image. In [107], a multi-scale deep neural network is proposed to learn a mapping between hazy images and their corresponding transmission maps. A densely connected pyramid network is proposed in [149] to jointly estimate the transmission map, atmospheric light, and dehazed images, while an effective iteration algorithm is developed in [77] to learn the haze-relevant priors. [29] further embeds the atmospheric model into the designing of CNN and proposes a feature dehazing unit to ensure end-to-end trainable. However, it is known that the atmospheric scattering model (ASM) is not valid in certain scenarios [72], which limits the applicability of the aforementioned dehazing methods. Unlike those ASM-dependent methods, [27] integrates multiple models to perform haze removal with attention, and [76] uses a GridNet-based network [35] to directly predict dehazed images via an ASM-agnostic approach. To further improve the performance in ASM-agnostic setting, [28] propose an multi-scale boosted dehazing network (MSBDN) with boosting strategy and back-projection technique. [53] firstly introduces knowledge distillation in solving dehazing problem. It allows dehazing model learn to dehaze from both ground truths and teacher outputs.

Many methods that have been developed for other image restoration tasks, e.g. deblurring, denoising, are also highly relevant. To remove blurring caused by the dynamic scenes, a multi-scale convolutional neural network is proposed in [87] to restore sharp images in an end-to-end manner. In [44], the weighted nuclear norm minimization (WNNM) problem is studied and applied to image denoising by exploiting non-local self-similarity. This work is later extended to handle arbitrary degradation, including blur and missing pixels [143]. To tackle the long-term dependency problem, the MemNet [123] is proposed by introducing a memory block, consisting of a recursive unit and a gate unit, to explicitly mine persistent memory through an adaptive learning process. To make the deep networks implementable on limited resources, a new activation unit is proposed [59], which enables the net to capture much more complex features, thus requiring a significantly smaller number of layers in order to reach the same performance. A super-resolution generative adversarial network (SRGAN) is developed in [64] to recover high-frequency details and produce more natural-looking images.

2.4 Formulation for Indirect Domain Shift

In this section, we provide a theoretical formulation of the image dehazing problem and propose an indirect domain shift method as an effective approach to obtaining an approximation solution.

Denote the prior distribution of clear images of size $m \times n$ by p_X, which is defined on a low dimensional manifold $\mathcal{M}$ in $\mathbb{R}^{3\times m\times n}$. The image degradation mechanism can be modeled as a conditional distribution $p_{X|Y}$, i.e., given the clear image x, a distorted image y is generated according to $p_{Y|X}$. Note that p_X and $p_{Y|X}$ induce the joint distribution $p_{X,Y}$ as well as the conditional distribution $p_{X|Y}$; in general, both p_X and $p_{Y|X}$ need to be learned from the training data. Image dehazing can be formulated

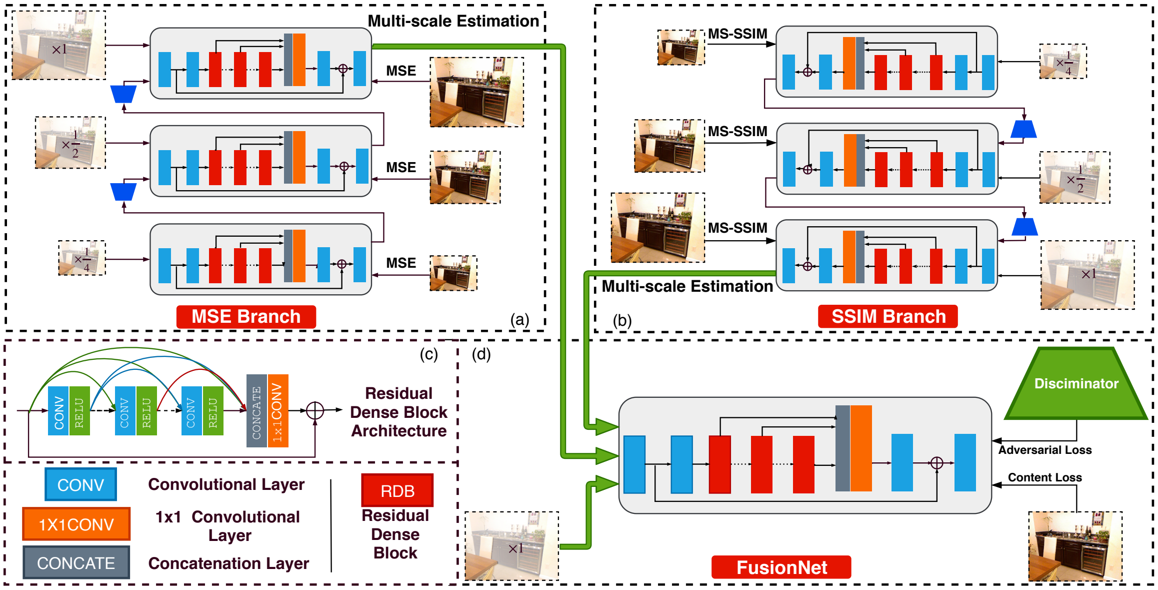


Figure 2.1: One example of the proposed IDS network. (a) and (b) are the multi-scale estimation with MSE and SSIM loss, respectively. (d) is the FusionNet with adversarial and content loss. (c) shows the legend.

as a maximum a posterior estimation problem:

$$\hat{x}_{\text{map}} = \arg\max_{\hat{x}\in\mathcal{M}} p_{X|Y}(\hat{x}|y). \tag{2.4.1}$$

In practice, one often considers the following alternative formulation:

$$\begin{aligned}\hat{x}_\ell &= \min_{\hat{x}\in\mathbb{R}^{3\times m\times n}} \mathbb{E}\left[\ell(X,\hat{x})|Y=y\right] \\ &= \min_{\hat{x}\in\mathbb{R}^{3\times m\times n}} \int_{\mathcal{M}} p_{X|Y}(x|y)\ell(x,\hat{x})\mathrm{d}x,\end{aligned} \tag{2.4.2}$$

where ℓ is a loss function. In general it is expected that both $\hat{x}_{\text{map}}$ and $\hat{x}_\ell$ are close to the ground truth. However, there is no guarantee that $\hat{x}_\ell$ belongs to $\mathcal{M}$.

We shall describe an IDS method, which leverages multi-scale estimation and multi-branch diversity to obtain an approximate solution of (2.4.2), then lifts it into $\mathcal{M}$ using the adversarial loss to produce a candidate solution of (2.4.1). A network that realizes the IDS method is shown in Figure 2.1.

2.4.1 Multi-scale Estimation

Note that (2.4.2) requires the knowledge of $p_{X|Y}$, which needs to be estimated from the training data, hence we solve the following approximated version of (2.4.2), i.e.,

$$\hat{x}'_{\ell} = \min_{\hat{x} \in \mathbb{R}^{3 \times m \times n}} \int_{\mathcal{M}} p'_{X|Y}(x|y)\ell(x, \hat{x})\mathrm{d}x, \tag{2.4.3}$$

where $p'_{X|Y}$ is an approximation of $p_{X|Y}$ learned from the training data. To ensure that $\hat{x}'_{\ell} \approx \hat{x}_{\ell}$ (and consequently close to the ground truth), we need $p'_{X|Y}(x|y) \approx p_{X|Y}(x|y)$ for $x \in \mathcal{M}$ (at least for x in a neighborhood of y that contains the ground truth). However, since the difference between the ground truth and the distorted version y is not negligible, this neighborhood could be quite large, rendering a good approximation of $p_{X|Y}(\cdot|y)$ in this neighborhood difficult to obtain. Indeed, the number of parameters need to specify $p_{X|Y}(\cdot|y)$ in this neighborhood might be comparable or even larger than the available training data, hence a direct approximation can be highly unreliable, especially considering the fact that the approximation is in general done in a suboptimal way. For this reason, it is sensible to first approximate $p_{\tilde{X}|Y}$ (with $\tilde{x}$ being a low-resolution version of the ground truth), which itself is an approximation of $p_{X|Y}$ and can be specified by a significantly smaller number of parameters (as compared to $p_{X|Y}$). In this way, we can get a good approximation of $p_{\tilde{X}|Y}$, denoted by $p'_{\tilde{X}|Y}$, and solve the following optimization problem instead:

$$\tilde{x}_{\tilde{\ell}} = \min_{\hat{x} \in \mathbb{R}^{3\times m\times n}} \int_{\mathcal{M}} p'_{\tilde{X}|Y}(x|y)\tilde{\ell}(x,\hat{x})\mathrm{d}x. \tag{2.4.4}$$

Since $p'_{\tilde{X}|Y}(x|y)$ is a good approximation of $p_{\tilde{X}|Y}(x|y)$, it is expected that $\tilde{x}_\ell$ is close to $\tilde{x}$ and consequently not very far away from the ground truth. Now with $\tilde{x}_\ell$ at hand, we can further convert (2.4.2) to the following problem:

$$\hat{x}_\ell = \min_{\hat{x} \in \mathbb{R}^{3\times m\times n}} \int_{\mathcal{N}(\tilde{x}_\ell)} p_{X|\tilde{X}_\ell,Y}(x|\tilde{x}_\ell, y)\ell(x,\hat{x})\mathrm{d}x, \tag{2.4.5}$$

where $\mathcal{N}(\tilde{x}_\ell)$ is a neighborhood of $\tilde{x}_\ell$ that is large enough to cover the ground truth. It suffices to have a good approximation $p_{X|\tilde{X}_\ell,Y}(\cdot|\tilde{x}_\ell, y)$ over $\mathcal{N}(\tilde{x}_\ell)$. The above procedure is repeated until the required neighborhood is small enough.

We assume that the smaller the neighborhood becomes, the fewer number of parameters are needed to specify the distribution defined over this neighborhood and consequently the approximation becomes easier. Multi-scale estimation is introduced to mimic conventional coarse-to-fine optimization methods and has been widely applied in many computer vision tasks [30, 31, 87, 107].

2.4.2 Multi-branch Diversity

The idea underlying multi-branch diversity is similar. Suppose we adopt two branches with different loss functions, denoted by ℓ_1 and ℓ_2, respectively, then (2.4.5) becomes

$$\hat{x}_\ell = \min_{\hat{x} \in \mathbb{R}^{3\times m\times n}} \int_{\mathcal{N}(\tilde{x}_{\ell_1})\cap\mathcal{N}(\tilde{x}_{\ell_2})} p_{X|\tilde{X}_{\ell_1},\tilde{X}_{\ell_2},Y}(x|\tilde{x}_{\ell_1}, \tilde{x}_{\ell,2}, y)\ell(x,\hat{x})\mathrm{d}x. \tag{2.4.6}$$

It should be clear that multi-branch diversity further narrows the region over which the distribution needs to be estimated. In our experiments, we choose ℓ_1 and ℓ_2 to be mean square error (MSE) and structural similarity index (SSIM) loss, respectively. The reason we choose MSE and SSIM as loss functions is that MSE focuses on the pixel-level difference while SSIM pays more attention to the perceptual quality. See Figure 2.1 (a) and (b) for the architecture of two multi-scale estimation branches of the proposed IDS network.

2.4.3 Adversarial Loss

The role of the adversarial loss ℓ_{ad} is to lift $\hat{x}_\ell$ into $\mathcal{M}$. Specifically, consider a neural network subject to the weighted loss $\ell + \lambda\ell_{ad}$, which can be interpreted as solve the following problem:

$$\hat{x}_{\ell+\lambda\ell_{ad}} = \arg \max_{\hat{x}\in\mathcal{N}(\hat{x}_\ell,\lambda)} p_X(\hat{x}), \tag{2.4.7}$$

where $\mathcal{N}(\hat{x}_\ell, \lambda)$ is a neighborhood of $\hat{x}_\ell$. In general, this optimization problem tends to give a reconstruction that falls into $\mathcal{M}$ since p_X is only positive on $\mathcal{M}$. Note that the size of $\mathcal{N}(\hat{x}_\ell, \lambda)$ depends on λ. Specifically, $\mathcal{N}(\hat{x}_\ell, \lambda)$ is large when λ is large. In the extreme case of $\lambda \to \infty$, we have $\hat{x}_{\ell+\lambda\ell_{ad}} \to \arg\max_{\hat{x}\in\mathcal{M}} p_X(\hat{x})$; while when λ is very small, $\mathcal{N}(\hat{x}_\ell, \lambda)$ may have no intersection with $\mathcal{M}$, and in this case (2.4.7) reduces to (2.4.2). In principle it is desirable to choose the smallest λ such that $\mathcal{N}(\hat{x}_\ell, \lambda)$ intersects with $\mathcal{M}$. It is also worth noting that p_X is in general unknown. So one has to solve a modified version of (2.4.7) with p_X replaced by p'_X, which is an approximation of p_X learned from the training data.

The adversarial loss serves an important role of generating texture details in image

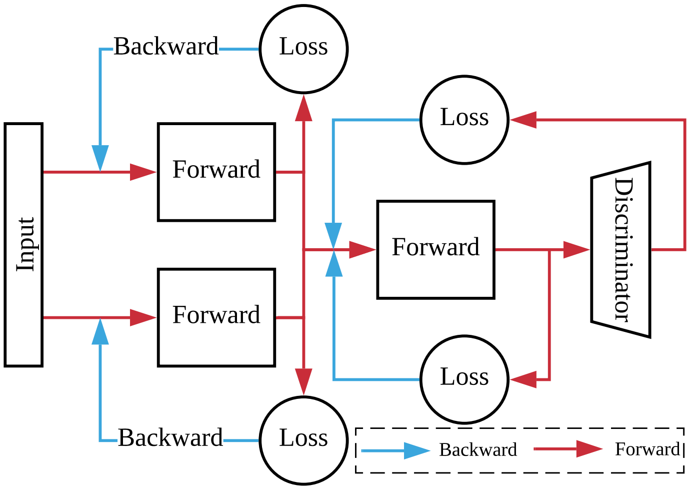


Figure 2.2: The isolated training of one iteration in hard IDS.

restoration. One of the reasons for its success in our framework is that, by leveraging multi-scale estimation and multi-branch diversity, one can already obtain an good estimate $\hat{x}_\ell$ which is in a narrow neighboring region of $\mathcal{M}$, and consequently the generator does not need much "imagination" to produce a natural-looking image. However, we observe the similar phenomenon reported in [64] that adversarial loss is helpful for faithful reproduction, even though the final PSNR metric is slightly lower. Nevertheless, we introduce the adversarial loss to obtain better perceptual quality but not expect higher PSNR value. The relevant ablation study can be found in Section 2.6.3.

2.5 Implementation

In this section, we provide a detailed implementation of the indirect domain shift (IDS). We also propose two training schemes, i.e., the hard IDS and soft IDS.

2.5.1 Network Architecture

The proposed IDS network is shown in Figure 2.1, which consists of three basic components, i.e., the MSE branch, the MS-SSIM branch, and the FusionNet. The MSE and SSIM branches are built with multi-scale structure to successively map hazy images to their clear counterpart at different resolution levels (as in (2.4.5)); moreover, they are supervised by non-identical loss functions to ensure differentiated outputs. The FusionNet completes the domain shift process by merging the outputs from the two branches together with the input hazy image into a single clear image (as in (2.4.6)). We train the FusionNet (see Figure 2.1 (d)) using a content loss defined as the weighted sum of MSE loss and perceptual loss [56]. The weight is carefully selected by searching from 1.0, 10^{-1}, 10^{-2}, and 10^{-3}. We find that our network achieves the best performance when the weight is set to 10^{-2}. An adversarial loss (see (2.4.7)) is also imposed on the FusionNet to enhance the perceptual quality of the final result.

To be specific, inside each diversity branch, there are three sub-networks, each performing domain shift at a different scale level. The input of the coarse-scale sub-network is obtained from the original hazy image via bi-linear interpolation with a down-sampling factor of 4. Its output is up-sampled with a factor of 2 via pixel shuffle [118], then fed into the medium-scale sub-network, together with the down-sampled hazy image representation by a factor of 2. The input of the fine-scale sub-network is the concatenation of the original hazy image representation and the up-sampled output of medium-scale sub-network.

It is known that residual networks (ResNets) can facilitate gradient flow while dense networks (DenseNets) help maximize the use of feature layers via concatenation and dense connection. To capitalize on their respective strengths, [153] proposes

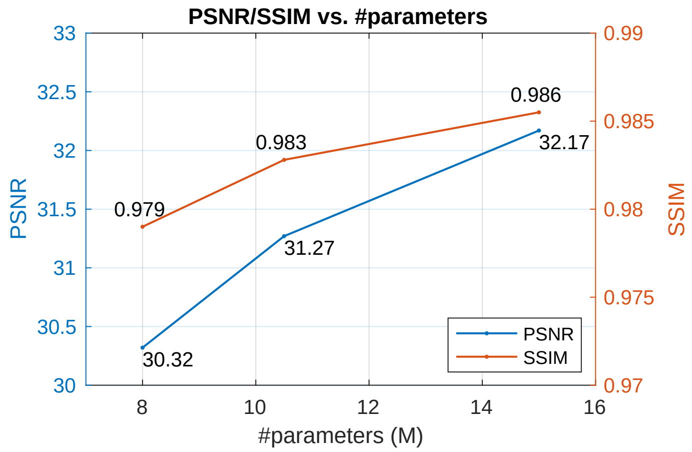


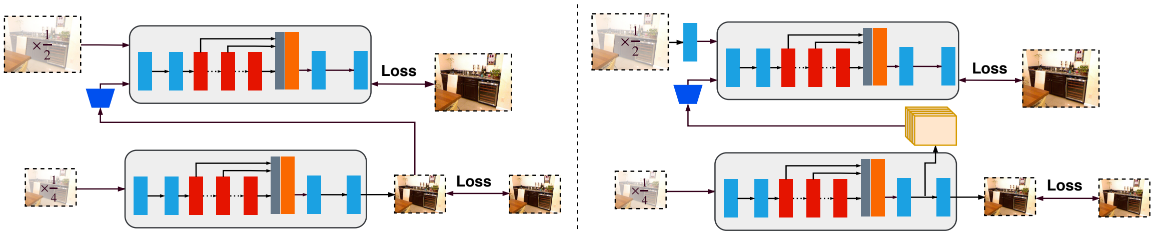
Figure 2.3: The performance of hard IDS with different parameters.

so-called residual dense networks (RDNs), which consist of contiguous memory blocks, local residual learning blocks and global feature fusion blocks.

In this work, we use RDNs as the fundamental building components of the proposed IDS network. See Table 2.1 for detailed specifications. Note that hard IDS and soft IDS adopt the same network structure, but differ in terms of the number of trainable parameters. Model depth will be detailed in Section 2.6.4.

2.5.2 Training Scheme

To handle the coexistence of multiple loss functions, we propose two back-propagation strategies characterized by different effective ranges of the loss functions. Specifically, we can separately update each module according to the associated loss function or jointly update all modules according to a global loss that aggregates the local ones. This results in the two IDS training schemes, i.e., hard IDS and soft IDS.



(a) (b)

Figure 2.4: The difference between (a) Hard IDS and (b) Soft IDS.

#RDB/#Conv		Shadow	Medium	Deep
Branch	Coarse	4/3	5/3	6/3
	Finer	6/4	7/4	8/4
	Finest	8/5	9/5	10/5
FusionNet		10/6	12/6	15/6

Table 2.1: The configuration of the shadow, medium, and deep Hard IDS corresponding to Figure 2.3.

Hard IDS

We first present the isolated training strategy for hard IDS shown in Figure 2.2. Specifically, each module is supervised independently by the associated loss functions and deliver dehazed images to the next stage after updating their weights. Note that in this case, the convergence of the entire network does not depend on the convergence of all loss functions, which means that the network performance may become stable before all loss functions are small enough. This is a consequence of direct mapping, since for each mapping step it suffices to enter one of many (almost) equally good confident neighborhoods, resulting in lower computational load. One advantage of isolated updating is that the gradient vanishing problem can be alleviated. Recall that this problem is caused by the emergence of small gradients in the earlier layers of very deep networks during back-propagation. As a comparison, isolated training shortens the back-propagation path, but maintains the depth of forward inference, at the expense of heterogeneous convergence rates of different loss functions. It is also worth noting that the isolated training strategy closely follows our analytical formulation which dictates how to shift from one domain to another. Therefore, the success of hard IDS can be viewed as a good indication of the correctness of our theoretical framework.

Soft IDS

In contrast to hard IDS, here a global loss function obtained by combining all local module losses is used to update network parameters via end-to-end back-propagation. Although the local losses are evaluated based on the images output by the respective modules, only the feature map from the penultimate convolutional layer of each module is delivered to the next module. This enables soft IDS to accomplish the desired task largely in the feature space. The fact that each module no longer has to re-map the previous module's output images back to the feature space is helpful for reducing the number of parameters and also making the indirect shifting path 'smoother'. Another advantage of soft IDS is that there is no need to be concerned with the convergence of a specific module as in hard IDS, which facilitates the training process.

In summary, the differences between Hard and Soft IDS are in two main aspects: (1) As in Figure 2.4, Hard IDS and Soft IDS deliver images and features to the next stages, respectively. (2) The Hard IDS adopts isolated training (optimization over modules independently), while Soft IDS computes the summation of all the local module losses and optimizes the entire notwork in an iteration.

Scale	Branch	Adversarial	PSNR	SSIM
✗	✓	✓	31.13	0.983
✓	✗	✓	29.80	0.982
✓	✓	✗	**31.32**	**0.986**
✗	✗	✓	30.55	0.975
✓	✓	✓	$\underline{32.17}$	**0.986**

Table 2.2: Ablation studies on the SSIM/PSNR performance of Hard IDS. The best performance is shown in **bold**, while second best results are with $\underline{\text{underline}}$.

Scale	Branch	Adversarial	PSNR	SSIM
✗	✓	✓	34.12	0.985
✓	✗	✓	32.75	0.984
✓	✓	✗	**34.74**	$\underline{0.986}$
✗	✗	✓	33.92	0.981
✓	✓	✓	**34.74**	**0.987**

Table 2.3: Ablation studies on the SSIM/PSNR performance of Soft IDS. The best performance is shown in **bold**, while second best results are with $\underline{\text{underline}}$.

2.6 Ablation Study

We conduct ablation studies to investigate the respective contributions of multi-scale estimation, multi-branch diversity, and adversarial loss using RESIDE-standard indoor dataset [68] that will be introduced in detail in Section 2.7.1. To eliminate the influence of other factors, all training configurations are kept the same as that presented in Section 2.7.2, including the total number of trainable parameters for each network. More detailed analysis is shown in supplementary.

2.6.1 Multi-scale Estimation

As mentioned in Section 2.4.1, a direct mapping can be highly unreliable, since the number of trainable parameters might be comparable or even larger than the available

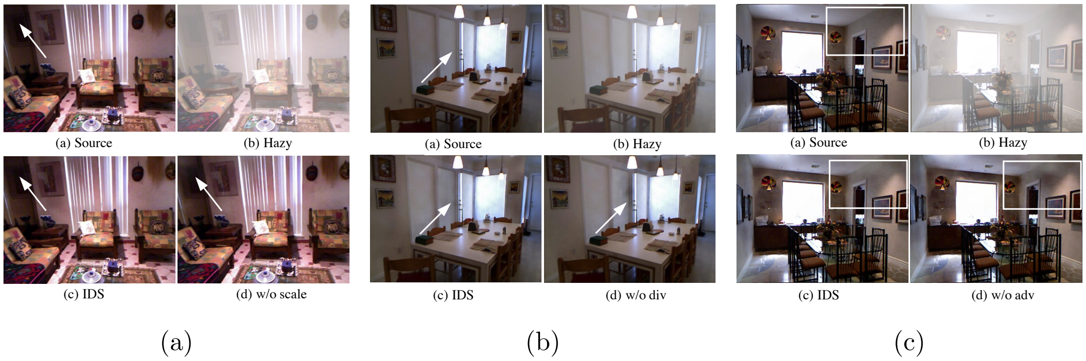


Figure 2.5: Some output examples of Hard IDS (a) without multi-scale estimation (w/o scale), (b) without multi-branch diversity (w/o div), (c) only with adversarial loss (o/w adv), and without adversarial loss (w/o adv) in the ablation study, respectively.

training data. To overcome this problem, a multi-scale network is applied in the first stage of IDS. Another important property of such coarse-to-fine estimation is the local-global consistency: the coarse-scale network first estimates the holistic structure of the image scene, and then a fine-scale network performs refinement based on both local information and the coarse global estimation. To further study the influence of such coarse-to-fine structure, we test the performance of IDS framework without multi-scale estimation (w/o scale).

Following the ablation principle, we remove the coarse-scale network and make the fine-scale network deeper to have the same number of parameters. One output example is presented in Figure 2.5(a) indicating that hard IDS w/o scale is able to recover the image reasonably well, but with some local inconsistency: the haze at the up-left corner is not removed faithfully. This verifies the above analysis that multi-scale network is able to capture both local and global features. We present the performance on PSNR and SSIM for both hard IDS and soft IDS in Table 2.2 and Table 2.3, respectively. It can be seen that IDS w/o scale performs worse than IDS (especially in soft IDS), indicating that the local inconsistency has impact on both the quantitative metrics and perceptual quality.

2.6.2 Multi-branch Diversity

Using multi-scale estimation with MSE loss, one can realize domain shift to a certain extent. However, some important information may get lost along the way. To keep the information diversity, we introduce one more multi-scale branch and employ SSIM loss in this branch. This strategy enables a more precise inference of local details by providing distinctive confident neighborhoods identified by different branches. To further illustrate its effectiveness of this strategy, we test the performance of IDS without multi-branch diversity (w/o div).

Similarly, we remove the second branch and make the first branch deeper. One of the examples is presented in Figure 2.5(b), in which the IDS w/o div sometimes delivers erroneous detail inference, since the "dark area" between the light and the wall clearly should not exist. This is further verified by the overall validation shown in the Table 4.2, in which there is a large performance gap between IDS and IDS w/o div, indicating that it is well worth having two branches.

2.6.3 Adversarial Loss

The adversarial loss (together with the content loss) is employed at the last stage (i.e., the FusionNet) of the proposed IDS framework and is served to obtain high visual quality. The FusionNet takes the estimates from the two branches, in conjunction with the original hazy image, as the input and generates the final output with perceptually satisfactory high-frequency details via proper fusion. Since the estimates produced by the two branches are already in the neighboring domains of the target, the generator does not need to rely on pure "imagination" to create texture details; instead, it could, to a great extent, maintain the perceptual reality rather than blindly pursue a higher PSNR [64].

To prove this, we show that IDS without adversarial loss is able to produce a high PSNR but NOT able to obtain better perceptual quality. Following the ablation principle, we construct IDS IDS without adversarial loss (w/o adv) by simply removing discriminator. As can be seen, IDS w/o adv produces a slightly higher PSNR in Figure 2.5(c) (26.508), but obviously lower perceptual quality than IDS (26.094), as the wall is printed "darker" partially to minimize the MSE distance. This demonstrates the generalization capability of the generator and provides further justifications for the IDS framework.

To further prove the necessity of adversarial loss, we compare with GridDehaze [76]. GridDehaze [76] is a pure CNN based dehazing method without adopting adversarial loss to generate natural distributed outputs. From Figure 2.6, it shows that the generated images from Soft IDS tend to be closer to the ground truth with less inconsistent color gradients on the road, sky, and wall. This verifies the phenomenon that the adversarial loss is introduced to obtain better perceptual quality but not blindly pursue higher PSNR value.

2.6.4 Model Depth

This section is devoted to investigating the impact of model depth on the performance of our hard IDS method. By adjusting the number of convolutional and dense residual blocks, we construct shadow, medium, and deep models with 8 M, 10.5

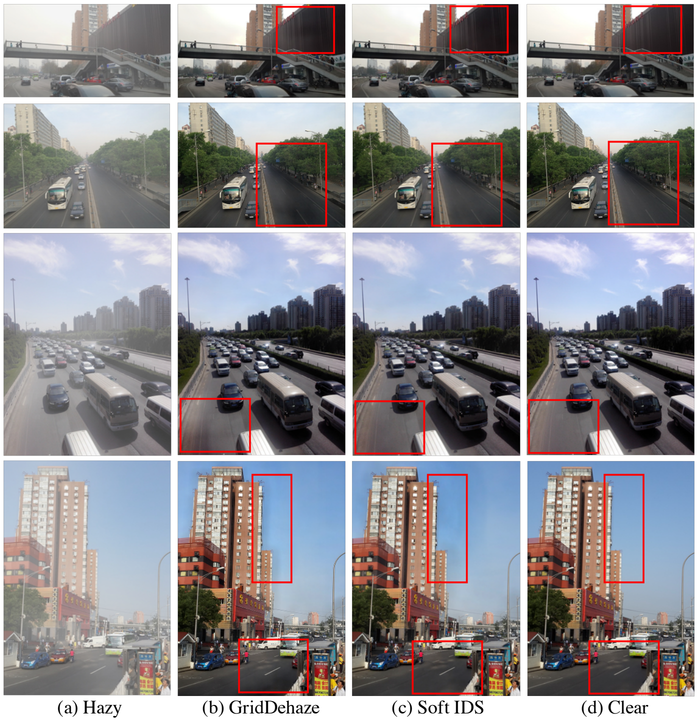
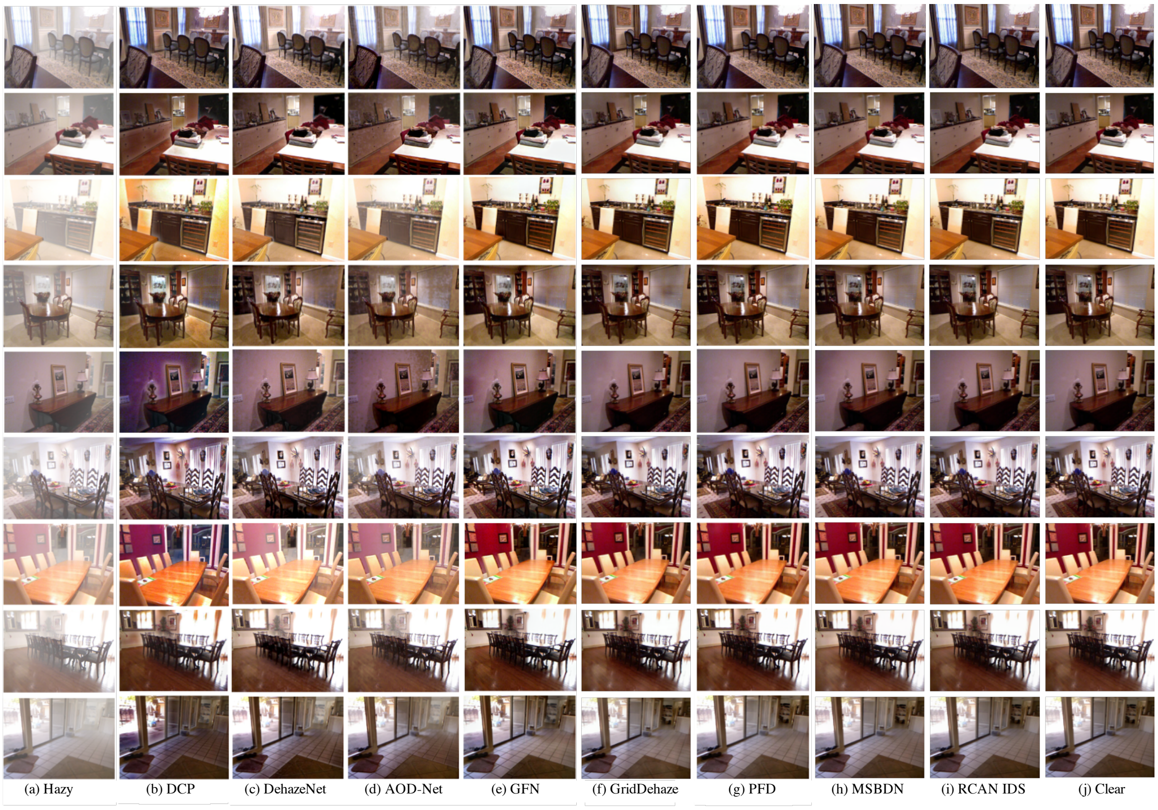


Figure 2.6: The output examples from SOTS outdoor testing set.

M, and 15 M trainable parameters, respectively. Detailed specifications is shown in Table 2.1. As expected, the deep model achieves the best overall performance in terms of both PSNR and SSIM. As illustrated in Figure 2.3, both PSNR and SSIM values improve dramatically as the number of parameters increases, which further verifies the effectiveness of the IDS framework. It is worth mentioning that albeit with fewer trainable parameters (around 4.3 M), soft IDS still manages to outperform hard IDS as shown in Table 2.4.

Dataset	Metrics	DCP	DehazeNet	AOD-Net	GFN	GridDehaze	PFD	MSBDN	Hard IDS	Soft IDS	RCAN IDS
Indoor	PSNR	16.62	21.14	19.06	22.30	32.16	32.68	33.79	32.17	**34.74**	**35.34**
	SSIM	0.8179	0.8472	0.8504	0.880	0.9836	0.9760	0.9840	0.9860	**0.9869**	**0.9901**
Outdoor	PSNR	19.13	24.75	24.14	28.29	30.86	31.17	31.33	30.78	**31.52**	**32.73**
	SSIM	0.8605	0.9269	0.9198	0.9621	0.9819	0.9825	0.9832	0.9815	**0.9832**	**0.9873**

Table 2.4: The SSIM/PSNR performance of different methods on SOTS-indoor, and SOTS-outdoor. Our proposed methods and improved network with RCAN outperform the others.



(a) Hazy (b) DCP (c) DehazeNet (d) AOD-Net (e) GFN (f) GridDehaze (g) PFD (h) MSBDN (i) RCAN IDS (j) Clear

Figure 2.7: The output examples from SOTS indoor testing set of the SOTA methods.

2.7 Experiments

In this section, we further compare the proposed IDS network with several state-of-the-art dehazing algorithms, including dark channel prior (DCP) [48], DehazeNet [17], AOD-Net [67], gated fusion network (GFN) [108], GridDehazeNet (GridDehaze) [76], PFD [29] and MSBDN [28]. For a fair comparison, all these algorithms are evaluated on both synthetic and realistic datasets in terms of visual effect and quantitative accuracy. We adopt the peak signal to noise ratio (PSNR) [155] and the structural similarity index (SSIM) [136] for evaluation.

2.7.1 Benchmark Dataset

For training and testing purposes, we use the RESIDE-standard dataset [68], which is a benchmark for single image dehazing. The indoor training set (ITS) of RESIDE-standard contains 13990 synthetic hazy indoor images (together with haze-free counterparts). These synthetic images are generated using NYU2 [91] and Middlebury stereo [115] with the medium extinction coefficient β chosen uniformly from $(0.6, 1.8)$ and the global atmospheric light $\mathbf{A}$ chosen uniformly from $(0.7, 1.0)$. The outdoor training set (OTS) of RESIDE-standard contains 296695 hazy images generated from 8477 clear counterparts with β chosen uniformly from $(0.04, 0.2)$ and $\mathbf{A}$ chosen uniformly from $(0.8, 1.0)$. The testing set (SOTS) of RESIDE-standard contains 500 synthetic hazy indoor/outdoor images (together with haze-free counterparts). We also perform comparisons using the real-world hazy image dataset in [33] to show the perceptual difference.

2.7.2 Training Details

Our algorithm is implemented using the PyTorch library [97] and all tests are conducted on the same GPU of Nvidia Titan Xp. We train the network with the following configuration: the Adam optimizer [58] is applied with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, where a mini-batch size of 10, a patch size of 180×180, an initial learning rate of 10^{-4} are

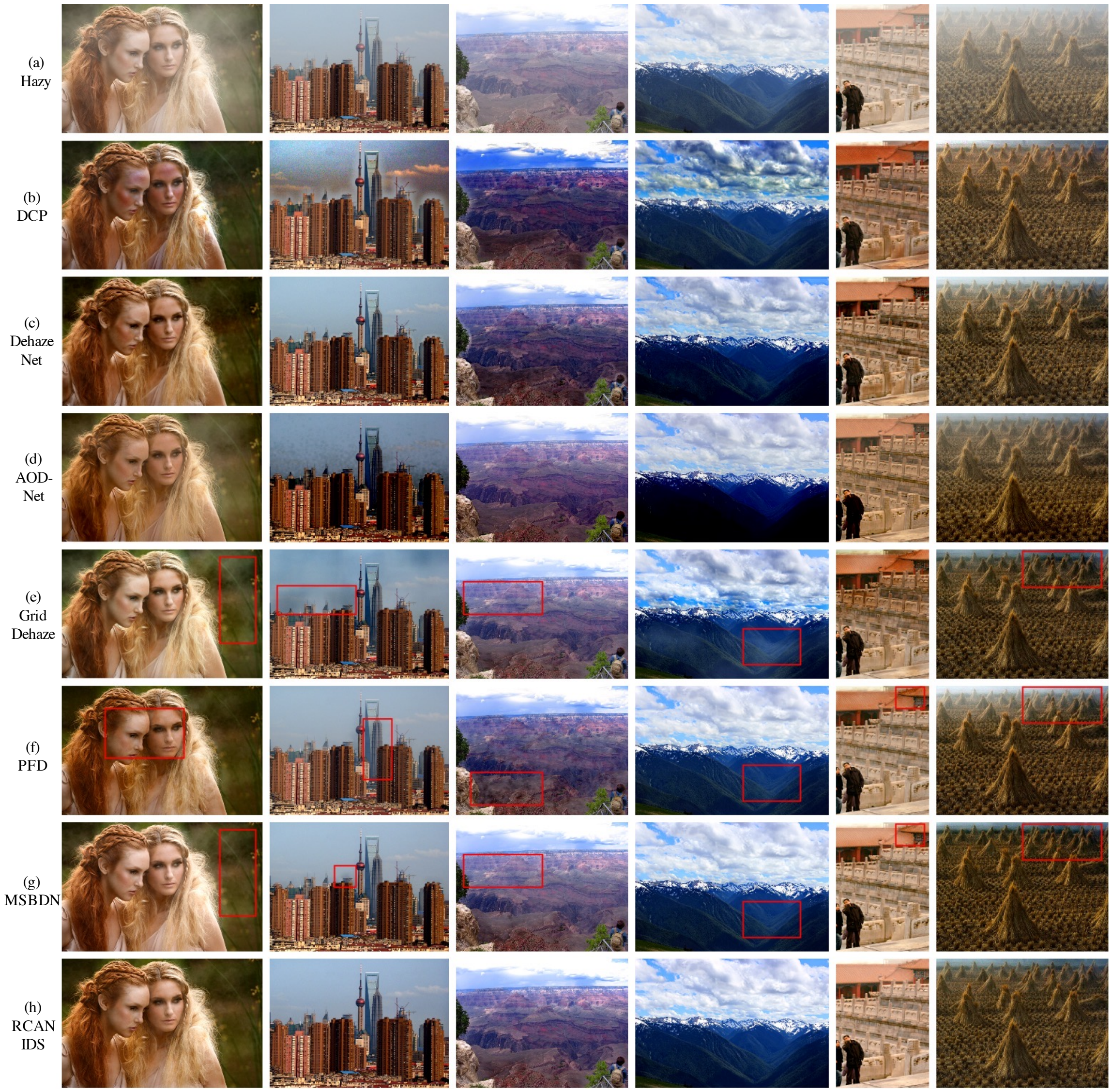


Figure 2.8: The output examples from real-world images in Fattal et.al. to compare with SOTA DNN based methods.

adopted. For hard IDS, the learning rate decays with a multiplicative factor of 0.5 every 120 epochs for a total of 700 epochs, while soft IDS is trained for 100 epochs with the learning rate reduced by half on the 60th, the 80th, and the 90th epochs. Besides, horizontal/vertical random flipping is applied for data augmentation. It is

DNN Based (GPU Running Time)								
DehazeNet	AOD-Net	GFN	GridDehaze	PFD	MSDBN	Hard IDS	Soft IDS	RCAN IDS
0.190s	0.004s	0.011s	0.22s	0.103	0.088	0.048s	0.035s	0.041s

Table 2.5: Average per-image (620 × 460) runtime (second) on SOTS-indoor.

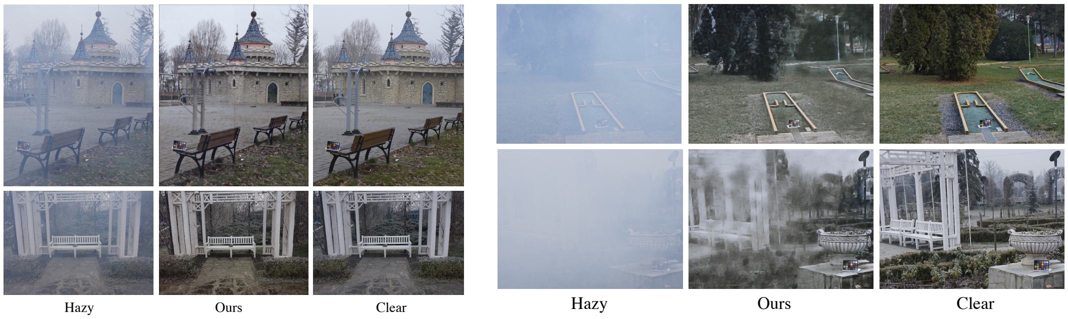
worth mentioning that after random flipping of both input and target images, the training data are still paired. Therefore, such an augmentation strategy is not harmful to supervised training but help expand the size of training data.

2.7.3 RCAN as Substitute

The proposed IDS framework is generic in nature and admits many different concrete implementations. In this work, we have focused on a particular implementation with RDNs as fundamental building blocks. However, this is by no means the best possible one. Indeed, the performance of our IDS network can be further improved by adopting more powerful substitutes of RDNs. To demonstrate this, we replace RDNs in soft IDS by residual channel attention networks (RCANs) [152] with the same number of trainable parameters. We further illustrate the effectiveness of adopting RCANs as substitute in the following experimental results.

2.7.4 Evaluation on Benchmark Dataset

We train our network from scratch on RESIDE-standard ITS, OTS and validate it on the separated testing dataset SOTS. The quantitative results and the qualitative results are shown in Table 2.4 and Figure 2.7, respectively. Here hard IDS corresponds to the deep model in Table 2.1, while soft IDS is as described in Section 2.6.4. It can



(a) (b)

Figure 2.9: Qualitative evaluation on Dense-Haze and NH-Haze dataset.

be seen from Table 2.4 that soft IDS outperforms the other methods under comparison in terms of PSNR and SSIM. In particular, the PSNR achieved by soft IDS reaches 34.74 on SOTS indoor dataset. Moreover, with the boost from RCANs substitute, RCAN IDS outperforms the others by a large margin.

As for visual quality, prior-based methods [48] overestimate the haze thickness, which results in color distortion (e.g. the color of the wall turns purple in the fifth row in Figure 2.7). Although some learning-based baseline methods [17, 67] avoid the color distortion problem, they tend to deliver unsatisfactory haze removal results for shaded regions. For example, in the seventh row of Figure 2.7, the area behind the arch should be dark; however, the restoration results produced by most baseline methods show light color instead. This is probably because of that the baseline methods fail to correctly estimate the depth information and consequently mislead by the haze effect. GFN generates decent results, and removes the haze in this area reasonably well. A possible explanation is that GFN does not rely on depth estimation for haze removal; it can also be attributed to the multi-scale approach adopted by GFN, which is an important ingredient of the IDS framework as well. Exploiting the full strength

O-HAZE			Dense-Haze		
Metrics	PSNR	SSIM	Metrics	PSNR	SSIM
DCP	12.92	0.505	DCP	10.85	0.404
AOD-Net	17.69	0.616	AOD-Net	13.30	0.469
GridDehaze	22.76	0.721	GridDehaze	14.56	0.493
MSBDN	23.28	0.743	MSBDN	15.18	0.509
Ours	**23.84**	**0.766**	**Ours**	**15.78**	**0.543**

Table 2.6: The SSIM/PSNR performance of different methods on O-Haze and Dense-Haze dataset. Our proposed methods outperform the others.

of IDS enables us to obtain better dehazing results. GridDehaze [76], PFD [29] and MSBDN [28] are the methods that can produce dehazed images comparable to ours. However, they still generates inconsistent color gradients on the venetian blinds in the fourth row. On the other hand, it can be seen in Figure 2.7 that our dehazed images can hardly be distinguished from the ground truth.

2.7.5 Evaluation on Real-world Photographs

We further show the dehazing results on real-world images in [33] to illustrate the generalization ability of IDS. In Figure 2.8, Prior-based method [48] introduces color distortion and over enhancement on images.

It is clear that DehazeNet[17], and AOD-Net [67] fail to remove haze completely, especially in the last column where heavy haze can still be seen around the haystack. Moreover, they also tend to over-enhance the images (e.g. the mountains in the fourth column). Although GridDehaze [76] , PFD [29] and MSBDN [28] work well on the synthetic dataset, its generalization performance on real images is unsatisfactory. The red boxes in Figure 2.8 locate their unsatisfactory regions. Their weaknesses include color distortion, incomplete haze removal and over enhancement. We also notice that the proposed IDS is able to not only remove haze successfully, regardless whether it is dense or light, but also restore the texture details faithfully, which further proves the effectiveness of our method.

2.7.6 Evaluation on Real-world Datasets

The evaluation is conducted on the O-Haze [7], and Dense-Haze [8] datasets. The Two real-world datasets is challenging since they contain limited training images (45 and 55 respectively) and vivid haze patterns. Therefore, the performance on the two dataset can be a good indication to the effectiveness of the proposed methods. The training on the two datasets adopts same strategies as introduced in Section 2.7.2. For fair comparison, we omit to use pre-trained weights or data augmentations that are not introduced in Section 2.7.2. We demonstrate the evaluation quantitatively and qualitatively in Table 2.6 and Figure 2.9.

Results on NTIRE2018 O-Haze. We evaluate our proposed IDS on O-Haze dataset [7] following the data split in official NTIRE2018-Dehazing challenge [6]. It can be observed in Table 2.6 that our IDS outperforms the other methods in terms of PSNR and SSIM. Figure 2.9(a) shows that our approach reconstructs faithful and sharp haze free images with good perceptual quality.

Results on NTIRE2019 Dense-Haze. In contrast to O-Haze that mostly contains light haze, Dense-Haze [8] records images with denser and more homogeneous haze layer. We follows NTIRE-2019 challenge [9] to conduct evaluation. Qualitative results in Figure 2.9(b) demonstrate that even if the background scene is occluded by thick haze, our IDS is still able to restore these region. In particular, since the second testing sample in Figure 2.9(b) is covered by severe haze, the background scene is almost invisible to human eyes. Nevertheless, our IDS surprisingly removes dense haze and reconstructs identifiable details. Quantitative comparisons in Table 2.6 illustrate that our IDS is the top performing method.

2.7.7 Runtime

Table 2.5 shows runtime comparisons on the SOTS dataset. Our method is ranked the third among DNN-based methods. It is worth mentioning that in our implementation multi-scale estimation is performed branch by branch. A significant reduction in runtime is possible via a parallel implementation of multi-scale estimation in two branches.

2.8 Conclusion

In this chapter, it is shown that the traditional direct mapping methods cannot provide accurate direct mapping for image dehazing. To solve this problem, an indirect domain shift (IDS) method is proposed by adding explicit loss functions inside a deep CNN model to guide the dehazing process. Multi-scale estimation, multi-branch diversity, and adversarial loss play important roles in this method as shown by the ablation studies. We also propose two training schemes, which have their respective advantages. Specifically, hard IDS is less demanding in terms of computational resources and alleviates the gradient vanishing problem. Besides, hard IDS is designed according to our theoretical formulation and its success provides a strong empirical indication of the correctness of our indirect domain shift mechanism. On the other hand, soft IDS is easier to implement and in general yields better performance. We show that IDS achieves remarkable improvements compared with the state-of-the-art on five dehazing datasets. Despite the success of our method, the visual performance of IDS is not completely satisfactory on Dense-Haze dataset. Since the deep learning methods often require large-scale datasets for training, we believe the performance of our method on Dense-Haze dataset can be further improved by simply acquiring more training samples. From another perspective, one interesting direction for our future work is to enhance the IDS framework to enable good generalization with limited training data.

The following chapter is reproduced from a paper on arXiv:

Huan Liu, Junsong Yun, Chen Wang, and Jun Chen. ”Pseudo Supervised Monocular Depth Estimation with Teacher-Student Network”. arXiv preprint arXiv:2110.11545. 2021 Oct 22.

Contribution Declaration: Huan Liu (the author of this thesis) is the first author and main contributor of this article (more than 95%). He proposed the method, conducted experiments and composed the article. Prof. Junsong Yuan is the supervisor of Huan Liu when he was a visiting student at State University of New York at Buffalo; Dr. Chen Wang helped polish this paper; Prof. Jun Chen is the supervisor of Huan Liu and helped polish this article.

Chapter 3

Pseudo Supervised Monocular Depth Estimation with Teacher-Student Network

3.1 Abstract

Despite recent improvement of supervised monocular depth estimation, the lack of high quality pixel-wise ground truth annotations has become a major hurdle for further progress. In this work, we propose a new unsupervised depth estimation method based on pseudo supervision mechanism by training a teacher-student network with knowledge distillation. It strategically integrates the advantages of supervised and unsupervised monocular depth estimation, as well as unsupervised binocular depth estimation. Specifically, the teacher network takes advantage of the effectiveness of binocular depth estimation to produce accurate disparity maps, which are then used as the pseudo ground truth to train the student network for monocular depth estimation.

This effectively converts the problem of unsupervised learning to supervised learning. Our extensive experimental results demonstrate that the proposed method outperforms the state-of-the-art on the KITTI benchmark.

3.2 Introduction

Estimating depth from a single image is a challenging but valuable task in both computer vision and robotics. Recently, we have witnessed the tremendous success of monocular depth estimation in assisting complicated computer vision tasks such as 3D scene reconstruction, visual optometry [125], and augmented reality [104]. This success can be largely attributed to large-scale labeled datasets and deep convolutional neural network (DCNN) models. However, it can be very costly and in some cases impossible to obtain pixel-wise ground truth annotations for supervised training. As such, great attention has been paid to unsupervised monocular depth estimation [19, 41, 100, 159] in recent years. A common approach is to formulate unsupervised monocular depth estimation as a self-supervised image reconstruction problem [38, 41].

Despite its innovativeness, this approach has two intrinsic weaknesses. 1) Compared to the supervised monocular setting, they often use the photometric loss to indirectly control the quality of disparity maps, which is less effective. 2) Compared to the unsupervised binocular setting, using one image to generate the disparity map (with the second image indirectly involved) is less effective than simultaneously exploiting the stereo pairs. Intuitively, the two weakness are intimately related to the nature of unsupervised and monocular approach and consequently inevitable. In this work, we aim to train an unsupervised monocular depth estimation network that can partially avoid these weaknesses by using a teacher-student based pseudo supervision for

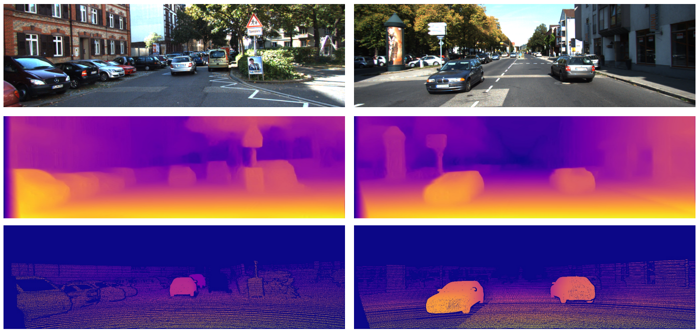


Figure 3.1: Example of the depth estimation results on KITTI 2015 stereo 200 training set by our proposed pseudo supervision mechanism. From the top to bottom are respectively the input images, our results and sparse ground truth disparities.

monocular depth estimation.

To this end, we propose a novel pseudo supervision scheme, which is leveraged to train the teacher-student network with distillation [52]. Specifically, the teacher network takes advantage of the effectiveness of unsupervised binocular depth estimation to produce accurate disparity maps. The disparity maps are then used as the pseudo ground truth to train the student network for monocular depth estimation, which converts the problem of unsupervised learning to supervised learning. This pseudo supervision mechanism enables us to exploit the benefits of both supervised learning and binocular processing for unsupervised monocular depth estimation. As a consequence, the aforementioned two weakness can be tackled to a certain extent.

However, in view of that it is not always possible to achieve perfect performance for the teacher network due to occlusion [157], in the distillation process the student network is also provided with occlusion maps, which indicate the performance gap between the teacher network's prediction (pseudo ground truth for the student) and the real ground truth. This occlusion indication allows the student to focus on dealing with the un-occluded regions. Moreover, the depth predictions in occlusion region still need to be carefully handled. To address this problem, we train the teacher network with semantic supervision to enhance the performance around the occlusion boundaries, which was verified to be effective [19, 30, 61, 131].

The main contributions of this work can be summarized as follows. 1) By taking advantages of both unsupervised binocular depth estimation and pseudo supervised monocular depth estimation, we propose a novel mechanism for unsupervised monocular depth estimation. 2) We fuse both occlusion maps and semantic representations wisely to handle the occlusion problem as well as boost the performance of student network. 3) We demonstrate through extensive experiments that our method outperforms the state-of-the-arts both qualitatively and quantitatively on the benchmark dataset[39].

3.3 Related Works

The existing monocular depth estimation methods can be roughly divided into two categories.

3.3.1 Supervised / Semi-supervised Monocular Depth Estimation

Supervised monocular depth estimation has been extensively studied in the past years. In the deep-learning framework, the problem becomes designing a neural network to learn the mapping from the RGB inputs to the depth maps. Eigen *et al.* [31] proposed a two-scale structure for global depth estimation and local depth refinement. Laina *et al.* [62] and Alhashim *et al.* [4] showed that better depth estimation results can be achieved with more powerful designs based on ResNet [49] and DenseNet [54]. There are also some works exploring the possibility of boosting the mapping ability of neural networks using statistical learning techniques. For example, Roy *et al.* [110] considered the combination of regression forests and neural networks; [66, 74, 141, 142] used conditional random fields (CRFs) and CNNs to obtain sharper depth maps with clear boundary.

Due to their alleviated reliance on large labeled real-world datasets, semi-supervised methods have also received significant attention. Nevertheless, they still require some additional information [20, 140, 163]. In particular, Guo *et al.* [46] proposed a teacher-student network for depth estimation, where the teacher network is trained in a supervised manner, albeit largely with synthetic depth data, and its knowledge is then transferred to the student network via distillation. Our work is partly motivated by the observation that the teacher network can actually be trained in a completely unsupervised manner without relying on any ground truth depth information (not even those associated with synthetic images).

3.3.2 Unsupervised Monocular Depth Estimation

In the unsupervised setting, only the RGB domain information, typically in the form of stereo images or video sequences, is provided. Many training schemes and loss functions have been proposed for unsupervised depth estimation to exploit photometric warps. Garg *et al.* [38] constructed a novel differentiable inverse warping loss function. Zhou *et al.* [158] proposed a windowed bundle adjustment framework with considering constraints from consecutive frames with clip loss. Godard *et al.* [41] introduced the notion of left-right consistency, which is imposed on both images and disparity maps.

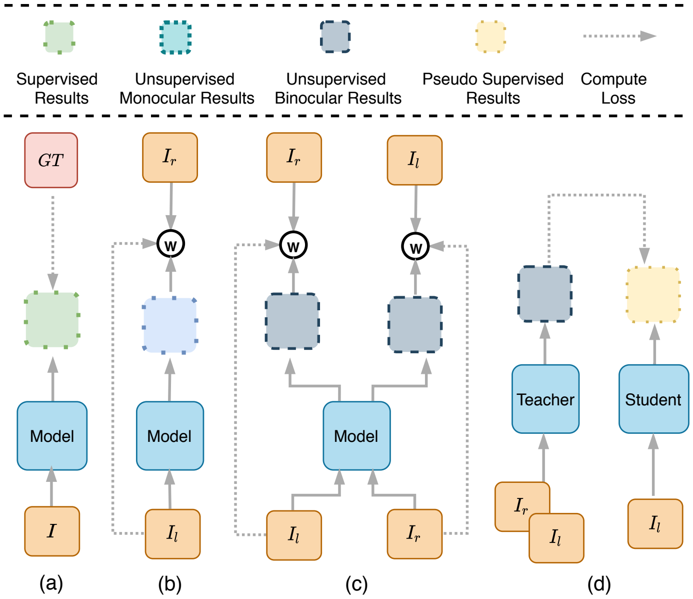

Supervised Results
Unsupervised Monocular Results
Unsupervised Binocular Results
Pseudo Supervised Results
Compute Loss
GT
I_r
I_l
w
Model
I
Teacher
Student
(a)
(b)
(c)
(d)

Figure 3.2: We show the architectures of (a) supervised/ (b) unsupervised monocular depth estimation, (c) unsupervised binocular depth estimation, and (d) our pseudo supervised mechanism.

Other consistency requirements, such as trinocular consistency [101] and bilateral consistency [137], were also investigated. In addition, there have been various attempts to take advantage of generative adversarial networks (GANs) [3, 99, 156], knowledge distillation [100], synthetic datasets [12, 90, 156, 164], or semantic information [19, 21, 55, 79, 154]. Among them, arguably most relevant to the present chapter is [100], where Pilzer *et al.* proposed a distillation mechanism based on the concept of cycle inconsistency. However, their adopted network structure is not very effective in simultaneously exploring the stereo pair and suffers from a mismatching problem [19]. In contrast, it will be seen that the proposed approach can take advantage of the efficiency of binocular processing in the training phase. Many recent works have recognized the benefit of exploiting semantic information for depth estimation via multi-task learning. Common approaches [21, 55, 79, 154] to multi-task learning typically involve neural networks with sophisticated structures. In contrast, Chen *et al.* [19] showed that it suffices to use a simple encoder-decoder network with a task identity variable embedded in the middle. Inspired by [86], we propose an alternative implementation with the task label stacked to the input images from the semantic dataset and KITTI to guide the teacher network for multi-task learning.

3.4 Proposed Method

3.4.1 Pseudo Supervised Depth Estimation Formulation

In this section, we provide a systematic comparison of several existing depth estimation formulations and show how the proposed pseudo supervision mechanism strategically integrates the desirable characteristics of different formulations.

Supervised Monocular Depth Estimation

Let I and h_{gt} denote the input RGB image and its ground truth depth map, respectively. Supervised training for monocular depth estimation aims to find a mapping F that solve the following optimization problem (Figure 3.2 (a)):

$$\begin{aligned} \arg\min_{F} \quad & error(h_e, h_{gt}), \\ \text{s.t.} \quad & h_e = F(I), \end{aligned} \tag{3.4.1}$$

where h_e is the estimated depth map of I. Given a well-specified depth target, it is possible to train a DCNN model $\hat{F}_1$, as an approximate solution to (3.4.1), that is capable of lifting I into a close neighborhood of h_{gt}. However, it can be very costly to obtain enough pixel-wise ground-truth annotations needed to specify the depth domain.

Unsupervised Depth Estimation

The unsupervised depth estimation can be classified as monocular and binocular depth estimation (stereo matching). Due to the unavailability of a directly accessible depth map, the following formulations are often considered (Figure 3.2 (b) and (c)):

$$\begin{aligned} \arg\min_{F} \quad & error(I_{el}, I_l), \\ \text{s.t.} \quad & I_{el} = \langle I_r \rangle_{d_l},\ d_l = F(I_l), \end{aligned} \tag{3.4.2}$$

$$\begin{aligned} \arg\min_{F_l, F_r} \quad & error(I_{el}, I_l) + error(I_{er}, I_r), \\ \text{s.t.} \quad & I_{el} = \langle I_r \rangle_{d_l},\ d_l = F_l(I_l, I_r), \\ & I_{er} = \langle I_l \rangle_{d_r},\ d_r = F_r(I_l, I_r). \end{aligned} \tag{3.4.3}$$

where (3.4.2) and (3.4.3) respectively refer to monocular and binocular estimation. (I_l, I_r) is a stereo pair, $\langle . \rangle$ is the warping operator, and $d_{l(r)}$ denotes the estimated left (right) disparity map. Note that $d_{l(r)}$ can be easily translated to a depth estimate given the focal length and the camera distance.

However, these solutions are in general not as good as $\hat{F}_1$ for the following reasons : 1) Using the warped image $I_{el(er)}$ with respect to $I_{l(r)}$ to indirectly control the quality of the depth estimate is less effective than comparing the depth estimate directly with the ground truth as done in the supervised setting. 2) I_l and I_r often exhibit slightly different object occlusion, rendering perfect estimation of $d_{l(r)}$ impossible. Nevertheless, $\hat{F}_3$ in principle performs better than $\hat{F}_2$ since monocular processing can be viewed as a degenerate form of binocular processing. Of course, the necessity of using stereo pairs as inputs restricts the applicability of binocular depth estimation.

Pseudo Supervision Mechanism

To strategically integrate the desirable characteristics of supervised monocular depth estimation, unsupervised monocular depth estimation, and unsupervised binocular depth estimation, we propose a pseudo supervision mechanism (Figure 3.2 (d)) as follows:

$$\begin{aligned} \arg\min_{F_s,F_t} \quad & error(d_e, d_{\tilde{gt}}), \\ \text{s.t.} \quad & d_e = F_s(I_l), d_{\tilde{gt}} = F_t(I_l, I_r), \end{aligned} \tag{3.4.4}$$

where F_t is a teacher network and F_s is a student network. The teacher network trained with stereo pairs (I_l, I_r) as in Figure 3.2 (c). Due to the advantage of binocular processing, the teacher network can be trained efficiently in an unsupervised manner and produce reasonably accurate disparity estimate. The pseudo ground truth disparity maps $d_{\tilde{gt}}$ produced by the trained teacher network $\hat{F_t}$ enable the student network to take advantage of supervised learning; moreover, in contrast to $\hat{F_t}$, the trained student network $\hat{F_s}$ is capable of performing monocular depth estimation. In order to ensure the pseudo ground truth produced by $\hat{F_t}$ with higher quality, a non-depth information (i.e. semantic maps) is integrated. The detailed implementation of the pseudo supervision mechanism is described below.

3.4.2 Training the Teacher Network

The teacher network is designed to thoroughly exploit the training data and provide the pseudo ground truth to the student network (see Figure 3.3). In addition, the teacher network is trained to learn the semantic information as well.

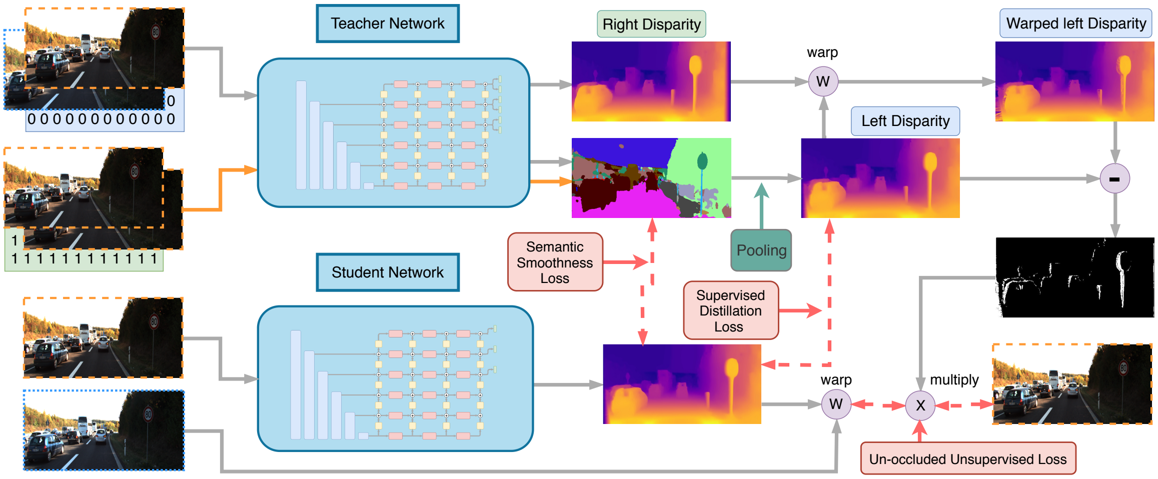


Figure 3.3: The pipeline of our proposed pseudo supervision mechanism. The teacher network is trained with alternating task-specific inputs (0 for semantic segmentation and 1 for depth estimation) while the student network is trained using the pseudo ground truth. During inference, the student take a single image and produce its disparity map accordingly.

Depth Estimation with Semantic Booster

Most depth estimation methods exploit semantic information by employing a two-branch network where semantic segmentation and depth estimation are performed separately. In contrast, inspired by [19] and [86], we design an encoder-decoder network that can switch between the aforementioned two tasks according to a task label. Given the input images I and the associated task labels c, the network outputs a task-specific prediction $Y = F_t(I, c)$. We set $c = 0$ when the network is trained for depth estimation and set $c = 1$ when the network is trained for semantic segmentation.

For semantic segmentation, we train our network supervised with ground truth semantic maps from an urban scene dataset. The loss function $\mathcal{L}_{seg}$ for this task is:

$$\mathcal{L}_{seg} = \mathcal{CE}(F_t(I, c = 0), gt), \tag{3.4.5}$$

where $\mathcal{CE}$ denotes cross-entropy loss and gt specifies the semantic ground truth label.

In contrast, for binocular depth estimation (*i.e.*, when $c = 1$), we adopt unsupervised training. Following [41], we formulate the problem as minimizing the photometric reprojection error (see Figure 3.2(c) and (3.4.3)). Specifically, given two views I_l and I_r, the network predicts their corresponding disparity maps d_l and d_r, which are used to warp the opposite views; the resulting $\tilde{I}_l \triangleq \langle I_r \rangle_{d_l}$ and $\tilde{I}_r \triangleq \langle I_l \rangle_{d_r}$ serve as the reconstructions of I_l and I_r, respectively. The loss function is a combination of $L1$ loss and single scale **SSIM** [136] loss:

$$\mathcal{L}_{re}(I, \tilde{I}) = \theta \frac{\mathbf{1 - SSIM}(I - \tilde{I})}{2} + (1 - \theta)\|I - \tilde{I}\|_1, \tag{3.4.6}$$

where θ is set to 0.5, and **SSIM** uses a 3×3 filter. We also adopt the left-right consistency loss $\mathcal{L}_{lr}$ and the disparity smoothness loss $\mathcal{L}_{sm}$ introduced in [41]:

$$\mathcal{L}_{lr}(d, \tilde{d}) = \|d - \tilde{d}\|_1, \tag{3.4.7}$$

$$\mathcal{L}_{sm}(d, I) = |\partial_x d| e^{-\|\partial_x I\|} + |\partial_y d| e^{-\|\partial_y I\|}, \tag{3.4.8}$$

where $\tilde{d}_l = \langle d_r \rangle_{d_l}$, $\tilde{d}_r = \langle d_l \rangle_{d_r}$, and ∂ is the gradient operator. Therefore, the total loss for unsupervised binocular depth estimation is $\mathcal{L}_{bi}$:

$$\begin{aligned} \mathcal{L}_{bi}(d_l, d_r, I_l, I_r) = {} & \alpha_1(\mathcal{L}_{re}(I_l, \tilde{I}_l) + \mathcal{L}_{re}(I_r, \tilde{I}_r)) \\ & + \alpha_2(\mathcal{L}_{lr}(d_l, \tilde{d}_l) + \mathcal{L}_{lr}(d_r, \tilde{d}_r)) \\ & + \alpha_3(\mathcal{L}_{sm}(d_l, I_l) + \mathcal{L}_{sm}(d_r, I_r)). \end{aligned} \tag{3.4.9}$$

Following [19], after the training process for semantic segmentation converges, we use semantics-guided disparity smooth loss within each segmentation mask to boost disparity smoothness especially on object boundaries. During training, we only predict semantic segmentation on I_l to reduce the computation load. Unlike [19], our semantic-guided smooth loss $\mathcal{L}_{semantic}$ is a simple variant of (3.4.8):

$$\mathcal{L}_{semantic}(d_l, s_l) = \mathcal{L}_{sm}(d_l, s_l), \tag{3.4.10}$$

where s denotes the predicted semantic map.

The overall loss function for the teacher network can be defined as follows:

$$\begin{aligned}\mathcal{L}_{teacher}(d_l, d_r, I_l, I_r, s_l) = {} & \gamma_1 \mathcal{L}_{bi}(d_l, d_r, I_l, I_r) \\ & + \gamma_2 \mathcal{L}_{semantic}(d_l, s_l).\end{aligned} \tag{3.4.11}$$

3.4.3 Training the Student Network

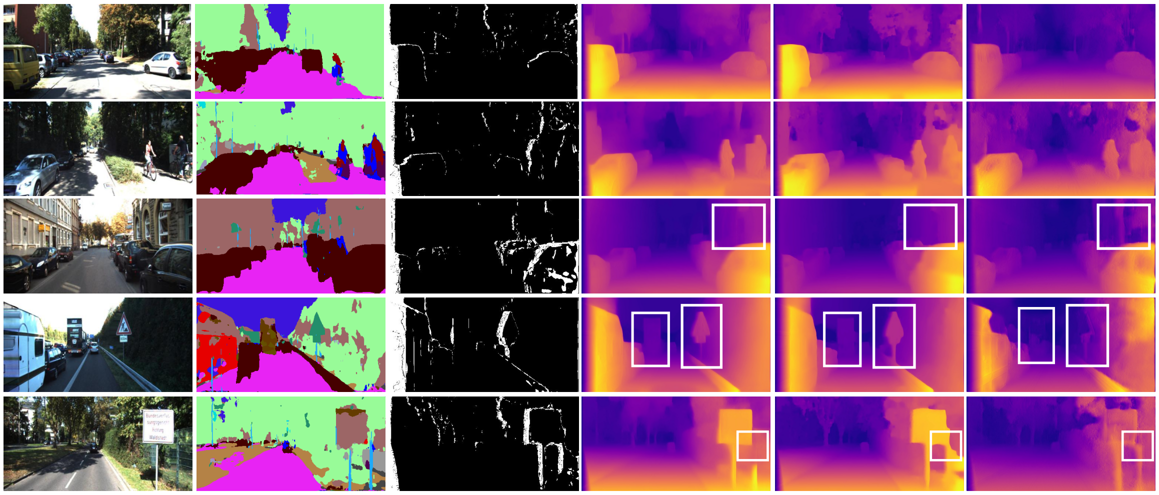
Now we proceed to discuss the training strategy for the student network as shown in Figure 3.3.

Supervised Training with Pseudo Disparity Ground Truth

The student network is trained under the supervision of the pseudo disparity ground truth provided by the teacher network. The adopted pseudo supervised distillation loss $\mathcal{L}_{sup-mo}$ is an adaptation of the reconstruction loss (3.4.6) to disparity maps:

$$\mathcal{L}_{sup-mo}(d_s, d_t) = \mathcal{L}_{re}(d_s, d_t), \tag{3.4.12}$$

where d_s and d_t are respectively the disparity estimate by the student and the pseudo disparity ground truth from the teacher.



(a) Input Image (b) Semantic Map (c) Occlusion Map (d) Ours Student (e) Ours Teacher (F) Monodepth

Figure 3.4: Illustrations of the experiment results on KITTI 2012 Eigen split. Monodepth denotes the results by Gordard *et al.*.

Method	Sup	Aux	Error (lower, better)				Accuracy (higher, better)		
			Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Eigen et al. [31]	Y	N	0.203	1.548	6.307	0.282	0.702	0.890	0.958
Guo et al. [46]	Y	D	0.096	0.641	4.059	0.168	0.892	0.967	0.986
Fu et al. [36]	Y	N	**0.072**	**0.307**	**2.727**	**0.120**	**0.932**	**0.984**	**0.994**
Garg *et al.* [38]	N	N	0.152	1.226	5.849	0.246	0.784	0.921	0.967
Pilzer et al. [100]	N	N	0.142	1.231	5.785	0.239	0.795	0.924	0.968
Zhou et al. [158]	N	N	0.135	0.992	5.288	0.211	0.831	0.942	0.976
Gordard et al. (Monodepth) [41]	N	N	0.124	1.388	6.125	0.217	0.841	0.936	0.975
Gordard et al. (Monodepth2) [42]	N	N	0.115	0.903	4.863	0.193	0.877	0.959	0.981
Ours (Student)	N	N	**0.099**	**0.901**	**4.783**	**0.178**	**0.908**	**0.970**	**0.984**
Chen et al. [19]	N	S	0.108	0.875	4.873	0.204	0.865	0.956	0.981
Lu et al.[79]	N	S	0.115	1.202	5.828	0.203	0.850	0.944	0.980
Ours (Student)	**N**	S	**0.090**	**0.853**	**4.671**	**0.167**	**0.912**	**0.972**	**0.988**
Ours (Teacher)	**N**	S	**0.059**	**0.777**	**3.868**	**0.137**	**0.959**	**0.983**	**0.991**

Table 3.1: Quantitative comparison with state-of-the-art methods on the KITTI 2015 eigen split. Elements in the supervision (Sup) column are marked by yes (Y) or no (N) to describe whether the methods adopt a supervision manner. In the Auxiliary supervision (Aux) column, N represents 'no extra supervision', D stands for 'Depth supervision' and S denotes 'semantic supervision'. Best results are in **bold** and the second best are with underline. No matter if semantic information is used or not, our proposed method outperforms all the others.

Unsupervised Training with Occlusion Maps

Since the binocular teacher network naturally fails to find a good reconstruction in occlusion region [157], the less capable monocular student network has little chance to succeed in this region. For this reason, it is sensible to direct the attention of the student network to other places where good reconstructions can be potentially found. Motivated by this, we generate an occlusion map from teacher as:

$$\mathcal{M}_{oc}(d, \tilde{d}) = \mathbb{1}(|d - \tilde{d}| \leqslant 0.01),$$

which sets the region that admits a good reconstruction (*i.e.*, the region where the reconstructed $\tilde{d}$ is close to the original map d) to 1 and sets the remaining part to 0.

Based on occlusion map, we further define an un-occluded unsupervised loss $\mathcal{L}_{un-mo}$ by masking out the difficult region:

$$\mathcal{L}_{un-mo}(d_s, I_s, \tilde{I}_s) = \mathcal{M}_{oc}\mathcal{L}_{re}(I_s, \tilde{I}_s) \tag{3.4.13}$$

where $\mathcal{L}_{re}$ and is the image reconstruction loss introduced in Section 3.4.2 (a); I_s and $\tilde{I}_s$ are respectively the monocular input and its reconstruction.

The semantic information S_t from the teacher network is also used to guide the training of the student network via loss (3.4.10) for handling occlusion boundaries. The total loss function for the student network can be defined as follow:

$$\begin{aligned}\mathcal{L}_{student}(I_s, \tilde{I}_s, d_s, d_t) = {} & \gamma_3\mathcal{L}_{sup-mo}(d_s, d_t) \\ & + \gamma_4\mathcal{L}_{un-mo}(d_s, I_s, \tilde{I}_s) \\ & + \gamma_5\mathcal{L}_{semantic}(d_s, S_t).\end{aligned} \tag{3.4.14}$$

In the inference phase, the student network F_s takes an image I_s and produces a disparity $d_s = F_s(I_s)$, from which the depth estimate D_s can be readily computed according to the formula $D_s = bf/d_s$, where b is the baseline distance between the cameras and f is the focal length of lenses.

3.5 Experiments

3.5.1 Implementation Details

Network Architecture

As shown in Figure 3.3, we shall refer to a specific encoder-decoder as Dense-Grid since the encoder is built using DenseNet161 [54] (in view of its feature extraction ability) without a linear layer while the decoder is built using GridNet [35] (in view of its feature aggregation ability) with a shape of 6×4. For the teacher network, the output end of each scale of the decoder is attached with two 3×3 convolutional layers. Depending on the task label, the first convolutional layer predicts semantic maps or left disparities (with the latter involving an extra global pooling step); the second convolutional layer predicts right disparities only. The two low-resolution disparity maps are up-sampled to full-scale to avoid texture-crop artifacts [42]. The structure of the student network is the same as that of the teacher network with the layers that predict segmentation and left disparities removed.

Regular Training Procedures and Parameters

Our method is implemented using Pytorch [97] and evaluations are conducted on the Nvidia Titan XP GPU. Guided by alternating task labels, the teacher network is trained on KITTI [39] and Cityscape [24] for depth estimation and semantic segmentation. This training phase ends after 50 epochs when both tasks converge. The segmentation map produced in the last epoch of this training phase is leveraged to train the depth estimation task under total objective loss (3.4.10). With the pseudo ground truth and occlusion maps provided by the teacher network, the student network starts training process, which takes 50 epochs.

During training, inputs are resized to 256 × 512. Data augmentation is conducted as in Gordard *et al.* [41]. We adopt the Adam optimizer with initial learning rate $\lambda = 10^4$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^5$. In the training of the student network the learning rate reduced at 30 and 40 epochs by a factor of 10, as well as the training of the teacher network. The weights of different loss components are set as following: $\gamma_1, \gamma_2, \gamma_3, \gamma_5, \alpha_1, \alpha_3 = 1.0$, $\gamma_4 = 0.05$ and $\alpha_2 = 0.5$

Over-training of Teacher Network

Over-training is usually considered undesirable since it tends to jeopardize the generalization ability of a model. However, in our current context, it is actually desirable to train overly. Indeed, with over-training, the pseudo ground truth provided by the teacher network is likely to be very close to the actual ground truth of the training data (see Table 3.2), which enables the student network to take advantage of pseudo supervised learning. Moreover, the fact that teacher network overfits the training data has no impact on the generalization ability of the student network because we train our student regularly without over-training. (Note that the generalization ability of the teacher is not a concern). To achieve this, we train our teacher network for depth task with additional 20 epochs. Without specifying, the student network performances reported in this chapter are along with the over-trained teacher.

Method	Abs Rel	Sq Rel	RMSE	RMSE log
Teacher (over training)	**0.061**	**0.407**	**2.635**	**0.132**
Teacher (regular training)	0.074	0.545	3.021	0.172

Table 3.2: Experimental results on KITTI 2012 Eigen split training set. Over-trained teacher can produce depth with lower error.

3.5.2 Performance on KITTI

Evaluations are conducted on KITTI 2012 and 2015 Eigen split [31]. Evaluation metrics used in this work are the same as those in [41] for fair comparison.

Quantitative Results

Table 3.1 shows a quantitative comparison of several state-of-the-art depth estimation methods and the proposed one on KITTI 2015. Due to its binocular nature, the teacher network has a significant advantage over the monocular methods, which is clearly reflected in performance evaluations (the evaluation results of the teacher network reported in Table 3.1 are collected without over-training). Not surprisingly, the student network is less competitive than the teacher network; nevertheless, it still outperforms the other methods under comparison in terms of accuracy and error metrics. We additionally compare the performance of our proposed method with Guo *et al.* [46]. For fair comparison, the model in [46] is trained with auxiliary ground truth depth and unsupervised fine-tuning on KITTI. Our student is trained with semantic maps (without ground truth depth). From Table 3.3, we can observe that without any supervision directly relevant to depth, our student still outperforms the Guo *et al.* [46].

Method	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Guo et al. [46] (with depth)	0.874	0.959	0.982
Ours student (with semantic)	**0.912**	**0.972**	**0.988**

Table 3.3: Comparing with Guo *et al.*. on KITTI 2015 eigen split.

Qualitative Results

To further illustrate the effectiveness of the pseudo supervision mechanism, we show some qualitative results in Figure 3.4 on KITTI 2012. It can be seen that the disparity maps produced by the student network are comparatively the best in terms of visual quality and accuracy. For example, the edges of traffic signs and cars are clearer, and objects are detected with lower failure rate. It is also interesting to note that the disparity maps produced by the teacher network (which is over-trained) suffer from several problems (e.g., failure to distinguish the traffic sign and the background in the last row of Figure 3.4). That is to say, although the teacher network does not have a good generalization ability on the test dataset due to over-training, it is able to provide high-quality pseudo ground truth to train a student network.

3.5.3 Ablation Study

We perform ablation studies to demonstrate the effectiveness of each component in our proposed framework. Special attention is paid to three aspects: a) the benefit of incorporating semantic information in training the teacher, b) the advantage of joint utilization of pseudo ground truth (PGT), occlusion maps, and semantic information in training the student, c) inherent advantage of the proposed pseudo supervision mechanism.

Method	Improvement			Error (lower, better)			
	PGT	Occ	Semantic	Abs Rel	Sq Rel	RMSE	RMSE log
Student	✗	✗	✗	0.127	1.215	5.520	0.268
	✓	✗	✗	0.122	0.919	5.093	0.211
	✓	✓	✗	0.119	0.959	5.056	0.210
	✓	✗	✓	0.117	0.888	**4.949**	0.205
	✓	✓	✓	**0.115**	**0.885**	**4.956**	**0.202**
Teacher	✗	✗	✗	0.089	0.973	4.423	0.190
	✗	✗	✓	**0.077**	**0.672**	**3.950**	**0.174**
Monodepth Res50 Original				0.133	1.142	5.533	0.230
Pseudo Supervised Monodepth				0.129	1.112	5.236	0.217

Table 3.4: Ablation studies on KITTI 2012 Eigen split.

Ablation Study for Training Teacher.

We compare the cases with and without semantic booster. It can be seen from Table 3.4 that the performance of the teacher network improves significantly with the inclusion of semantic information.

Ablation Study for Training Student

We consider using different combinations of pseudo ground truth (PGT), occlusion maps (Occ), and semantic information to train the student network. As shown by Table 3.4, each element contributes positively to the performance of the student network, and the full combination outperforms any partial ones.

Inherent Advantage

We re-implement our pseudo supervision mechanism using the ResNet-based structure proposed by Gordard *et al.* [41] in lieu of our Dense-Grid structure. It can be seen from Table 3.4 that this re-implementation yields better performance as compared to the Monodepth network *et al.* with exactly the same ResNet-based structure.

3.6 Conclusion

In this chapter, we propose a pseudo supervision mechanism to realize unsupervised monocular depth estimation by strategically exploiting the benefits of supervised monocular depth estimation and unsupervised binocular depth estimation. We have also shown how to make effective use of performance-gap indicator, and semantic booster in the implementation of the pseudo supervision mechanism. The experimental results indicate that the proposed unsupervised monocular depth estimation method performs competitively against the state-of-the-art. As to future work, apart from refining the proposed depth estimation method, we also aim to further enrich and strengthen the theoretical framework of pseudo supervision and explore its application to other computer vision problems.

The following chapter is a reproduction of an Institute of Electrical and Electronics Engineers (IEEE) copyrighted, published paper:

Huan Liu, Zijun Wu, Liangyan Li, Sadaf Salehkalaibar, Jun Chen, and Keyan Wang. ”Towards Multi-domain Single Image Dehazing via Test-time Training”. Accepted by IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022.

Contribution Declaration: Huan Liu (the author of this thesis) is the first author and main contributor of this article (more than 90%). He proposed the method, conducted experiments and composed the article. Zijun Wu and Liangyan Li helped conduct evaluation of the proposed method; Dr. Salehkalaibar helped polish this paper; Prof. Jun Chen is the supervisor of Huan Liu; Prof. Keyan Wang participated in the discussion at the beginning.

Chapter 4

Towards Multi-domain Single Image Dehazing via Test-time Training

4.1 Abstract

Recent years have witnessed significant progress in the area of single image dehazing, thanks to the employment of deep neural networks and diverse datasets. Most of the existing methods perform well when the training and testing are conducted on a single dataset. However, they are not able to handle different types of hazy images using a dehazing model trained on a particular dataset. One possible remedy is to perform training on multiple datasets jointly. However, we observe that this training strategy tends to compromise the model performance on individual datasets. Motivated by this observation, we propose a test-time training method which leverages a helper network to assist the dehazing model in better adapting to a domain of interest. Specifically, during the test time, the helper network evaluates the quality of the dehazing results, then directs the dehazing network to improve the quality by adjusting

its parameters via self-supervision. Nevertheless, the inclusion of the helper network does not automatically ensure the desired performance improvement. For this reason, a meta-learning approach is employed to make the objectives of the dehazing and helper networks consistent with each other. We demonstrate the effectiveness of the proposed method by providing extensive supporting experiments.

4.2 Introduction

Single image dehazing is a classic but still active research topic in low-level computer vision, which aims to restore clean images from the degraded hazy counterparts. Recently, many deep learning approaches [17, 26, 37, 67, 75, 76, 102, 108, 138, 147, 149] have been proposed to address this problem by training a neural network to approximate the mapping from hazy images to haze-free ground truths. As more and more dehazing datasets have been released, such as RESIDE [68], O-Haze [7] and NH-Haze [10], these methods are able to demonstrate their outstanding ability in handling different haze patterns. However, one important issue is left behind for consideration, i.e., handling different types of hazy images by a single network. To be specific, current methods are usually trained on the training split of a particular dataset and tested on the corresponding testing split. For example, the test accuracy on RESIDE indoor test set [68] is obtained by validating a dehazing model trained on the RESIDE indoor training set. Such an evaluation strategy allows the neural network to focus on a specific domain but evades the important problem of learning a general model across datasets. A seemingly simple remedy is to train a single dehazing model on all available datasets jointly. Intuitively, with the increase of data, the network can benefit from considering more kinds of haze patterns, leading to boosted

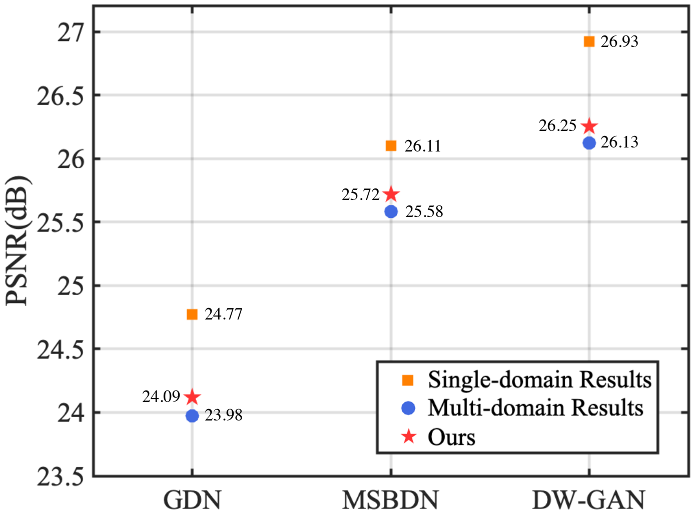


Figure 4.1: Average PSNR values of GDN, MSBDN and DW-GAN across four datasets. It can be observed that the dehazing methods perform better if the training and validation are conducted on a single dataset.

performance on every single dataset [5].

Somewhat surprisingly, we find that this naive solution actually compromises the dehazing performance on individual datasets. Indeed, it can be seen from Figure 4.1 that the dehazing models perform better when the training and testing are conducted on a single dataset (as opposed to all datasets combined). This unusual fact contradicts the common belief that the increase in data usually leads to improved performance. One possible explanation is that each dataset has a specific distribution which might be significantly different from that of another dataset (see Figure 4.2). The representative examples from the four datasets under consideration are shown in Figure 4.3, where one can observe that both the haze pattern and background scene of the four datasets are significantly different from each other. In the RESIDE indoor and outdoor datasets, the haze pattern is homogeneous but the background scenes are vastly different (indoor vs. outdoor environments). In the O-Haze and NH-Haze datasets, background scenes are consistent (outdoor environments) but the

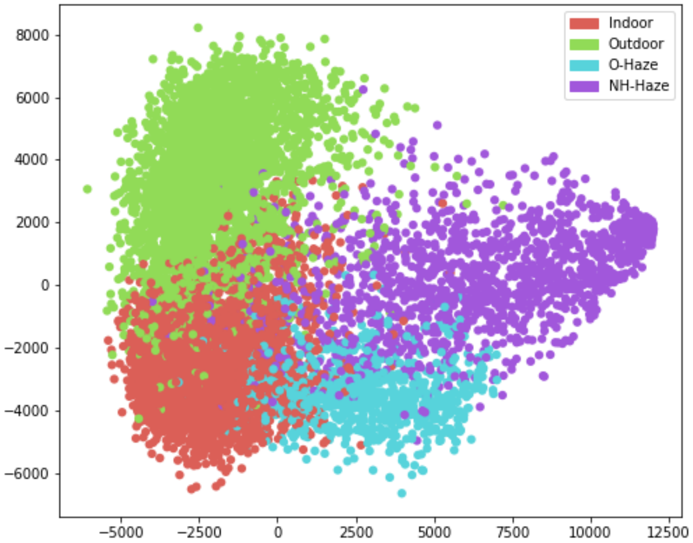


Figure 4.2: Visualization of image features using t-SNE. Image features are obtained using a ResNet18 pretrained on ImageNet. The fact that the features are clustered around four different centers shows clear discrepancies between the distributions of these datasets.

haze patterns have remarkable distinctions. Against this backdrop, learning a general dehazing model on multiple datasets can be categorized as a multi-domain learning (MDL) problem.

In this chapter, we propose a method that can enable a single dehazing model to cope with multiple domains. Here, each domain is formed by a dataset with a distinctive haze pattern and scene. Our goal is to find a model that can minimize the risk on the collection of domains for the dehazing task. Note that our problem definition is significantly different from the related field of domain adaptation and multi-task learning in the sense that the former aims to minimize the risk on a specific target domain while the latter performs optimization on a collection of tasks paired with a single domain. In principle, one can address the reformulated multi-domain dehazing problem by following the common practice in MDL. However, this requires designing sophisticated neural network structures with domain-specific modules, which is a highly non-trivial and cumbersome task in general.

To alleviate the design burden, we propose a novel MDL approach for single image dehazing by helping a given dehazing network to adapt to a specific domain when it is needed. To achieve this, we propose a method to adjust the dehazing network during the testing phase. In this method, the parameters of the network are optimized using a self-supervised loss function which is basically provided by another entity called the *helper network*. This network is designed to learn diverse haze patterns using paired hazy and haze-free images (across multiple domains) and output a reconstructed version of the hazy image that is fed into it. At the test time, the helper network uses its knowledge to assess the quality of the output of the dehazing network, which is a dehazed image. In other words, this image together with its corresponding hazy counterpart are given to the helper network as its inputs. If the output of the helper network is close to the hazy image, then a small reconstruction loss is expected. However, if the dehazed image is defective, then a large reconstruction loss may be derived at the helper network. Considering the fact that the quality of dehazed image can be represented by the reconstruction loss of the helper network, we update the parameters of the dehazing network by minimizing this loss function.

Now a natural question arise: How to guarantee that the end-to-end performance of the dehazing network is eventually optimized by minimizing the reconstruction loss of the helper network. In order to ensure the consistency between the objectives of two networks, we adopt the meta-learning approach [34]. Here, the goal of meta-learning is to adjust the parameters of the dehazing network by minimizing the reconstruction loss of the helper network so that the dehazing output based on the adjusted parameters better matches the ground-truth haze-free image.

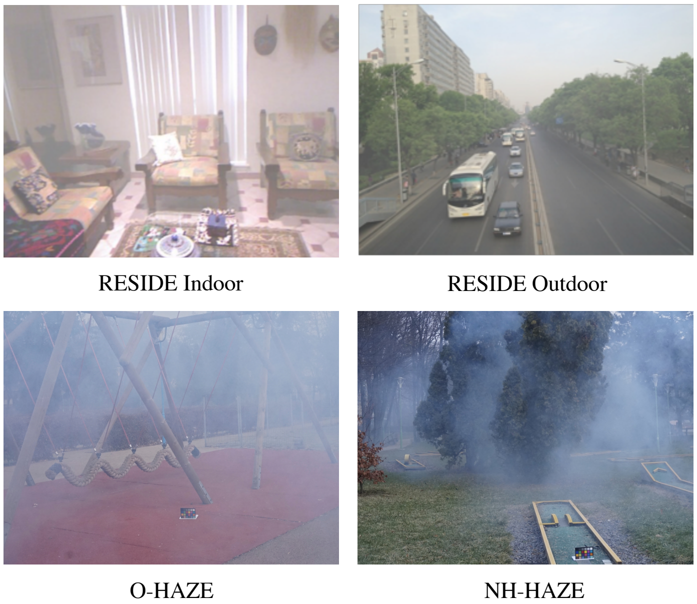


Figure 4.3: Representative examples from RESIDE indoor/outdoor, O-Haze and NH-Haze.

Our contributions can be summarized as follows. Firstly, we point out a largely unnoticed phenomenon in single image dehazing, namely, a model trained on multiple datasets exhibits compromised performance on individual datasets. This leads to the formulation of designing a dehazing model for distribution-wise distinctive datasets as a MDL problem. Secondly, we put forward a solution to this problem by introducing a test-time training approach for better adapting the dehazing network to every single observation. Finally, we provide extensive experiments to demonstrate the effectiveness of our proposed method in addressing the multi-domain dehazing problem.

4.3 Related Works

4.3.1 Single Image Dehazing

Most of the conventional single image dehazing methods [14, 33, 48, 161] are based on the estimation of parameters in the atmospheric scattering model (ASM) using statistical priors. However, they are not robust in dealing with complex real scenes. Recently, there has been a significant progress in the single image dehazing by using the deep learning approach. Although, [17, 67, 94, 107, 121, 145] still rely on the ASM, they propose to adopt a neural network to first estimate the transmission and then restore it. Due to the limitations of the ASM which make it not to be an effective method in modeling complicated haze patterns, other works [18, 28, 53, 71, 76, 78, 102, 103, 108, 139, 150] are designed using the end-to-end deep neural networks to directly learn the mapping from hazy images to haze-free counterparts. Another line of works [22, 69, 117] mainly focus on enabling the deep learning system to deal with natural hazy images. For example, in [117], a model is trained on multiple synthetic domains and the performance is evaluated on a specific real dataset. Our work is different from [117] in the sense that the performance of our proposed network is verified over different domains.

4.3.2 Multi-domain Learning

Multi-domain learning (MDL) aims to enable a model with the ability to minimize the risk across multiple domains. Usually, the model parameters can be divided into two distinctive parts according to their functionalities. Specifically, while one part focuses on learning the shared representations across different domains, the other part learns the domain-specific mapping relations [57, 88, 105, 116, 146]. Recent works consider developing a general system without explicitly learning the cross-domain or domain-specific representations. For example, [119] proposes a single model-based method to address problems in medical imaging. It uses the meta-learning to dynamically estimate hyperparameters in the loss functions. Notice the difference between the meta objectives of [119] and ours, i.e., our meta objective is designed to learn the consistency across losses. [135] introduces a universal object detector consisting of a single network using domain attention modules. These modules can activate the model parameters that are responsible for a particular domain. This approach still relies on a precise network design. However, our proposed approach is model-agnostic and can be used in a plug-and-play manner.

4.3.3 Meta-learning for Image Restoration

Meta-learning, also known as *learning to learn*, has attracted attention in the computer vision community, recently. Especially, the model-agnostic meta-learning (MAML) [34] is widely utilized in image restoration to improve the generalization ability of deep neural networks. For example, [95, 120] adopt MAML for super-resolution. The meta objective is to learn a model that can quickly adapt to novel scenes. [23] proposes to use the meta-auxiliary learning [122] for the test-time dynamic scene deblurring. Besides the obvious difference in the treated problems, our work offers two general insights regarding test-time training not present in [23]. 1) test-time training can be realized by building a helper network, detached from the main network (possibly off-the-shelf), to provide self-supervision during the test time. This idea is broadly applicable. It lifts the burden of jointly addressing the primary and auxiliary tasks in one framework and clears the way for wide adoption of test-time training. 2) The helper network should be designed by considering the special characteristics of the problem at hand (e.g., ASM is unique to dehazing) to maximize benefits of test-time training.

4.4 Methodology

Assume that we have a collection of N dehazing domains $\{D_i\}_{i=1}^{N}$ with M paired hazy and haze-free images $\{I_i, J_i\}_{i=1}^{M}$. We aim to train a dehazing model f_{θ_d} that is able to perform well on all domains. However, as mentioned before, we find through experiments that the model trained on a single domain can usually outperform the model trained on multiple domains.

In this section, we present one possible solution to address this problem. Firstly, we train the dehazing network f_{θ_d} using all image pairs from N domains by minimizing the following commonly used loss function:

$$\mathcal{L}_{dehaze}(\hat{J}, J) = \mathcal{L}_{smooth}(\hat{J}, J) + \gamma \mathcal{L}_{Per}(\hat{J}, J), \tag{4.4.1}$$

where $\hat{J}$ and J represent the dehazed and haze-free images, respectively, and $\mathcal{L}_{smooth}$ and $\mathcal{L}_{Per}$ represent smooth L_1 and perceptual losses [56], respectively. The parameter γ is used to get a weighted combination of the two loss functions Note that, the dehazing network can be any existing well-designed model.

Secondly, we will develop a helper network g_{θ_h} to learn the haze patterns in the following Section 4.4.1. This network gets a pair of hazy and haze-free images as its input and generates a reconstructed version of the hazy image at the output. It is

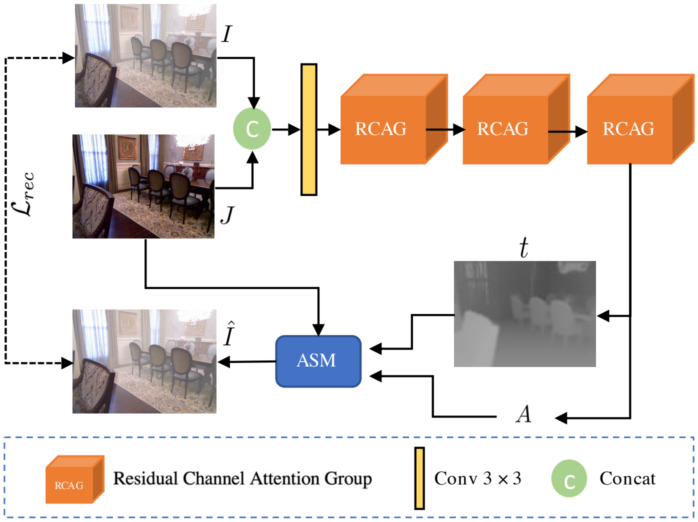


Figure 4.4: The helper network for learning the haze patterns. It consists of 3 stacked residual channel attention groups adopted from RCAN.

basically employed to determine the quality of the dehazed image $\hat{J}$ which is generated by the dehazing network. That is, if $\hat{J}$ is close to J, then a small reconstruction loss is expected at the helper network. During the test time, the dehazing network can update its parameters by minimizing the reconstruction loss of the helper network as will be discussed in Section 4.4.2. Although, this method helps the dehazing network to get an improved dehazing loss on the particular hazy inputs, however, an improved performance is not generally guaranteed. To address this problem, we finally propose a meta-learning approach in the following Section 4.4.3 to associate the dehazing and reconstruction losses with each other. Once the meta-training is complete, our setup is able to conduct the test time training which enables the dehazing network in producing clearer dehazing results in a self-supervised manner.

4.4.1 Learning the Haze Patterns

As discussed before, given a dehazed image $\hat{J}$, our goal is to effectively determine its quality and guide the dehazing network to produce a clearer counterpart. To achieve this, we build a helper network to explicitly learn the haze patterns across multiple domains and use this pre-learned knowledge to determine the quality of the dehazed images. The network structure is shown in Figure 4.4, where it takes paired hazy and haze-free images as the inputs and it outputs two key components of a haze pattern, i.e., the transmission $t(x)$ and global atmospheric light A. Then, we use the modified atmospheric scattering model (ASM) to reconstruct the hazy image by the following:

$$\hat{I}(x) = J(x)t(x) + A(1 - t(x)), \tag{4.4.2}$$

where $\hat{I}(x)$ and $J(x)$ are the reconstructed hazy and haze-free images at pixel x, respectively. Note that in the conventional ASM, $t(x)$ is defined as $t(x) = e^{-\beta d(x)}$, where $d(x)$ denotes the scene depth and the parameter β is a constant which controls the thickness of haze. However, in our proposed modification to this model, the parameter β is no longer assumed to be a constant. As shown in Figure 4.4, the transmission $t(x)$ is derived through a neural network. Using this method, it is possible to derive the transmission even when the haze is non-homogeneous. An illustration of learned transmission is shown in Figure 4.5. It is also worth emphasizing that one purpose of adopting the ASM model for the hazy image reconstruction is to avoid a trivial solution that the neural network can directly paste the input hazy image to the output side, without processing the hazy and haze-free images together.

Since the neural network and ASM model are fully differentiable, we can optimize the network on the combined domain using the loss (see Figure 4.4):

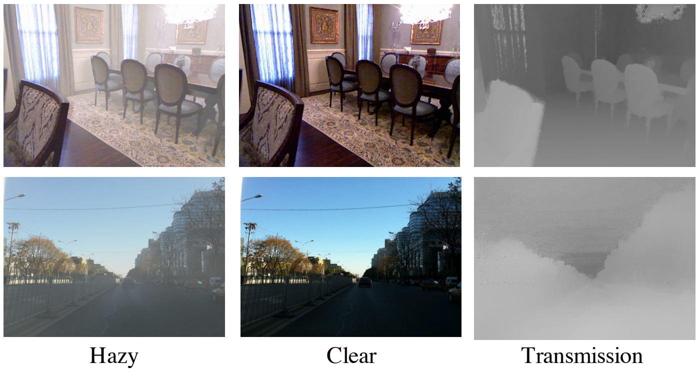


Figure 4.5: Illustrations of learned transmission map.

$$\mathcal{L}_{rec}(\hat{I}, I) = \mathcal{L}_{smooth}(\hat{I}, I) + \gamma \mathcal{L}_{Perc}(\hat{I}, I), \tag{4.4.3}$$

where I and $\hat{I}$ denote the hazy and reconstructed hazy images, respectively.

Once the helper network is trained to converge, it is able to reconstruct the hazy image by jointly employing hazy and haze-free images.

4.4.2 Dehazing Using Haze Reconstruction

During the test time, the dehazing network outputs a dehazed image $\hat{J}$ which might be different from the ground truth haze-free image J. So, feeding $\hat{J}$ and I (the hazy image) to the helper network as the inputs, it outputs a haze pattern which can result in a defective reconstructed hazy image I^{+}. And therefore, the corresponding loss $\mathcal{L}_{rec}(I^{+}, I)$ is larger than $\mathcal{L}_{rec}(\hat{I}, I)$ where $\hat{I}$ is the output of the helper network when I and J (the hazy and haze-free images) are fed into it.

Inspired by this, we perform test-time training on the dehazing network to minimize $\mathcal{L}_{rec}(I^{+}, I)$. Specifically, we update the dehazing network in few steps using $\mathcal{L}_{rec}(I^{+}, I)$ by the following:

$$\hat{\theta}_d \leftarrow \theta_d - \lambda_1 \nabla_{\theta_d} \mathcal{L}_{Rec}(f_{\theta_h}(f_{\theta_d}(I), I), I), \tag{4.4.4}$$

where λ_1 denotes the learning rate. Here, $\hat{\theta}_d$ is the updated weights of the dehazing model according to the reconstruction loss. Note that, the test-time training of the dehazing network is purely self-supervised by using the hazy image I, i.e., it does not require any manual labeling. Ideally, by minimizing $\mathcal{L}_{rec}(I^+, I)$, it can be expected that the image $\hat{J}$ produced by the updated dehazing network gets closer to the ground truth J over time. Therefore, we can finally produce improved dehazed images using the updated $\hat{\theta}_d$.

Despite the idea of developing the helper network to determine if the output dehazing results are clean counterparts of input hazy images, one might have a question that "is it true that the dehazing network can always benefit from the supervision of the reconstruction loss?" Unfortunately, we will show in our ablation studies at Section 4.5.5 that minimizing $\mathcal{L}_{rec}(I^+, I)$ is not always equivalent to minimizing $\mathcal{L}_{dehaze}(\hat{J}, J)$, which means that even if sometimes the dehazing network's produced output $\hat{J}$ steps far away from the ground truth J, it may be adopted by the helper network to reconstruct a hazy image $\hat{I}$ which is closer to I. The problem might be that the two losses are not consistent with each other and they lack explicit connections.

4.4.3 Learning Meta-objective

Inspired by the recent meta-learning approach [23, 122], where the test-time training is conducted via an auxiliary loss, we are further motivated to propose a meta-learning method across models. The goal of the meta-training is to learn the dehazing model parameters so that the dehazing loss is spontaneously minimized by optimizing the parameters based on the reconstruction loss.

Before the meta-training, we pre-train the dehazing and helper networks (θ_d and θ_h). They are independently trained by Eq. (4.4.1) and Eq. (4.4.3), respectively. Given a paired training data (I_i, J_i), we update the dehazing network using the reconstruction loss as follows:

$$\hat{\theta}_d = \theta_d - \lambda_1 \nabla_{\theta_d} \mathcal{L}_{rec}(f_{\theta_h}(f_{\theta_d}(I_i), I_i), I_i) \tag{4.4.5}$$

Intuitively, this update enables the dehazing network to produce results that can be adopted by the helper network to get an improved reconstructed hazy image. Considering the fact that we intend to use $\hat{\theta}_d$ to minimize the dehazing loss, we update the dehazing network by encouraging the performance of dehazing network to be maximized if the helper network can employ the dehazed image to get an improved reconstructed hazy image.

To that end, our meta objective can be formally defined as:

$$\arg\min_{\theta_d} \mathcal{L}_{dehaze}(f_{\hat{\theta}_d}(I_i), J_i). \tag{4.4.6}$$

Note that the dehazing loss is computed using the dehazed image $f_{\hat{\theta}_d}(I_i)$ produced by updated dehazing network $f_{\hat{\theta}_d}$, while the optimization is performed on θ_d. Eq. (4.4.6) can be achieved using the gradient descent as follows:

$$\theta_d \leftarrow \theta_d - \lambda_2 \nabla_{\theta_d} \mathcal{L}_{dehaze}(f_{\hat{\theta}_d}(I_i), J_i) \tag{4.4.7}$$

where λ_2 denotes the learning rate. The overall meta-learning procedure is summarized in the Algorithm. 1.

```
Algorithm 1: Meta Training
  Require: Pre-trained networks: $f_{\theta_d}$, $g_{\theta_h}$
  Require: Learning rates $\lambda_1$ and $\lambda_2$
  Output : Meta-learned model parameter $\theta_d$
  while not converge do
      Sample a batch of training data in $\{I_i, J_i\}_{i=1}^{M}$;
      for each $I_i$ do
          Compute updated parameters $\hat{\theta_d}$:
          $\hat{\theta_d} = \theta_d - \lambda_1 \nabla_{\theta_d} \mathcal{L}_{rec}(g_{\theta_h}(f_{\theta_d}(I_i), I_i), I_i)$
      Update:
      $\theta_d \leftarrow \theta_d - \lambda_2 \nabla_{\theta_d} \mathcal{L}_{dehaze}(f_{\hat{\theta}_d}(I_i), J_i)$
```

Methods		Indoor	Outdoor	O-Haze	NH-Haze	# Params	Runtime (s)
GDN [76]	Single-domain	27.79/0.953	28.93/0.972	23.23/0.808	19.14/0.710	3.84M	0.028
	Multi-domain	26.67/0.951	27.18/0.962	23.13/0.747	18.93/0.716	0.96M	0.028
	Ours	**26.83/0.952**	**27.26/0.961**	**23.21/0.747**	**19.04/0.716**	1.34M	1.043
MSBDN [28]	Single-domain	28.89/0.956	30.76/0.977	24.95/0.824	19.82/0.747	12.56M	0.153
	Multi-domain	28.53/0.961	30.31/0.973	23.97/0.764	19.51/0.725	3.14M	0.153
	Ours	**28.68/0.961**	**30.42/0.974**	**24.14/0.766**	**19.62/0.726**	3.52M	1.828
DW-GAN [37]	Single-domain	29.65/0.963	31.75/0.978	24.50/0.793	21.83/0.769	206.04M	0.076
	Multi-domain	28.84/0.941	31.21/0.974	24.02/0.789	20.44/0.763	51.51M	0.076
	Ours	**28.95/0.942**	**31.39/0.974**	**24.13/0.789**	**20.53/0.762**	51.89M	1.621

Table 4.1: Quantitative comparison of the dehazing results on multiple datasets using different training schemes. The term "single-domain" denotes that the method is trained on a single dataset and evaluated on the relative one; the term "multi-domain" represents that the network is trained using the combined dataset; the term "ours" denotes the results adopting the proposed test-time training. Accuracies are presented in the form of PSNR/SSIM.

Finally, after the meta-training, the updated dehazing and helper networks are ready to use. We can follow the procedure in Section 4.4.2 to conduct the test-time training.

4.5 Experimental Results

4.5.1 Datasets and Evaluation Metrics

Our experiments are conducted on widely used dehazing datasets, including O-Haze [7], NH-Haze [10] and RESIDE indoor/outdoor [68]. O-haze contains 40 image pairs, where the first 35 pairs are used for the training and the rest 5 pairs are adopted for the testing. NH-Haze consists of two variants that are released in 2020 and 2021. We form our NH-Haze dataset by combining both of them. For NH-Haze 2020, we adopt the official train, test split. As the validation and test data of NH-Haze 2021 is not released publicly, we take the first 22 pairs for the training and the other 3 pairs for the testing. Finally, our NH-Haze has a total of 67 training pairs and 8 testing pairs. RESIDE dataset is a benchmark for single image dehazing. We follow DADN [117] to form the training set by selecting 3000 indoor pairs and 3000 outdoor pairs and cropping them to the size of 256×256. For the testing, we adopt the Synthetic Objective Testing Set (SOTS) of RESIDE. The quantitative evaluation metrics used in this chapter are PSNR and SSIM [136].

4.5.2 Implementation details

We first pre-train the selected dehazing networks and our helper network on the combined dataset, which consists of the aforementioned O-Haze, NH-Haze and RESIDE indoor/outdoor datasets. The initial learning rate is set to 10^{-4} for the training of all networks except the GridDehazeNet[76], which is set to 10^{-3}. During the meta-training, the learning rates λ_1 and λ_2 in Eq. (4.4.5) and (4.4.7) are fixed to be 1.25×10^{-5} and 2.5×10^{-5}, respectively. The Adam optimizer[58] is used in both pre-training and meta-training with the default values of $\beta_1 = 0.9$ and $\beta_2 = 0.99$. In the test-time training phase, we perform 5 gradient updates on each hazy image and report the final accuracy. All our experiments are conducted on Nvidia V100 GPUs.

4.5.3 Degradation from Multi-domain Learning

As we denoted, a neural network trained on multiple domains is usually suboptimal when tested on each individual domain. Here, we provide quantitative results to investigate this phenomenon in single image dehazing. Our experiments are conducted using three popular learning-based methods, i.e., GDN [76], MSBDN [28] and DW-GAN [37]. To be specific, we first implement the single-domain training and testing, where a dehazing network is trained on a single dataset and tested on the relative one. Then, we take the same network to conduct the multi-domain learning, where the dehazing network is trained on the combined dataset that consists of all datasets introduced in Section 4.5.1. The results of the single-domain learning and multi-domain learning are shown in Table 4.1 denoted by "single-domain" and "multi-domain". It can be observed that both PSNR and SSIM of the multi-domain learned method are smaller than that of the single-domain learned. This fact indicates that simply collecting data for the dataset augmentation is not always useful.

4.5.4 Test-time Training on the State-of-the-art

To illustrate the effectiveness of our proposed method, we conduct quantitative and qualitative experiments using GDN [76], MSBDN [28] and DW-GAN [37] as the dehazing network. The experiments aim to show that our helper network can be helpful in boosting the performance of the existing approaches. Note that, our method is employed for the dehazing networks trained on multiple domains.

Quantitative Improvements: Table 4.1 summarizes the PSNR and SSIM measures on all four datasets. The performance of a dehazing network using our proposed test-time training is reported under the term “ours”. Thanks to different structures of the three networks, we can observe variations in the improvement of PSNR. For example, in the indoor testing set, our method can improve the PSNR of multi-domain learned GDN by 0.16dB, and improvements can be observed for MSBDN and DW-GAN, where the PSNR is increased by 0.15dB and 0.11dB, respectively. It can be easily checked throughout the table that our proposed test-time training can always improve the performance of the network trained on multiple domains. This further indicates the judicious model-agnostic property of our proposed test-time training method.

Qualitative Improvements: Figure 4.6 presents the dehazing results on O-HAZE and NH-HAZE. Here, we unfold the test-training process to provide a better understanding of our method. There are multiple problems shown in the initial results that can be fixed by conducting the proposed test-time training. In the first and second rows of Figure 4.6, we can notice that severe artifacts are added to the sky region of the dehazed images. Surprisingly, these artifacts can be removed gradually by few gradient updates. In the third and last rows, the results before updates are still hazy. However, our method is able to remove the haze from the initial results. Finally, other instances of color distortion are shown in the fourth and fifth rows, where after 5 updates, the dehazing network can produce more elegant images.

Algorithm Efficiency: Here, we investigate the efficiency of the proposed method in terms of the number of parameters. The last column of Table 4.1 reports the number of parameters that is required to dehaze on four domains. Thanks to the

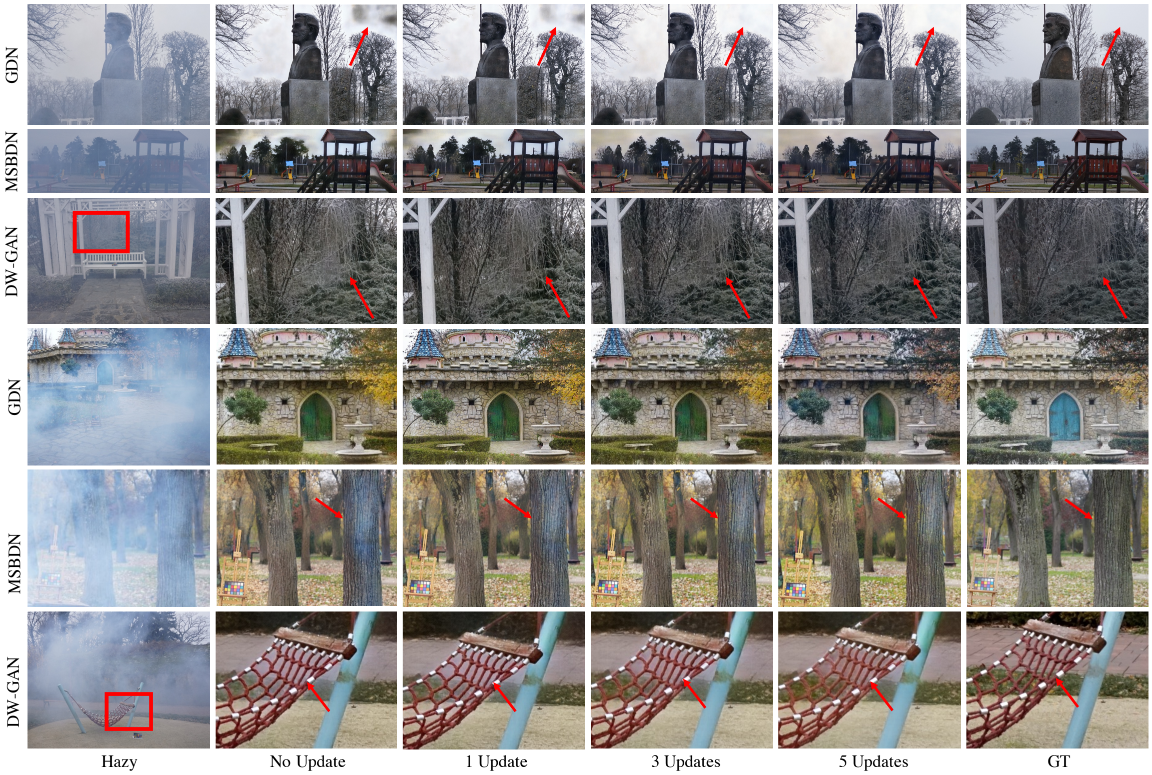


Figure 4.6: Qualitative results on O-Haze and NH-Haze datasets. The image samples of the first three rows are from O-Haze, and the others are from NH-Haze. After few gradient updates, the proposed test-time training can improve the image quality.

lightweight design of our helper network, by integrating our method with the current dehazing networks, the total number of parameters is still comparable to that of a single dehazing network. Especially when our method uses DW-GAN, there is a negligible increase in the number of parameters, however the PSNR is boosted by an average of 0.12dB. Moreover, considering the fact that we have four domains, if a model is separately trained on each domain, the deployment of four models is extremely memory-consuming as four collections of parameters need to be stored; this fact can be verified by observing the total number parameters for "signal-domain" in Table 4.1. However, we also find that using test-time training is slower than its one-shot

Methods	Indoor	Outdoor	O-Haze	NH-Haze
GDN	26.67/0.951	27.18/0.962	23.13/0.747	18.93/0.716
(a)	26.69/0.950	27.16/0.961	22.89/0.743	18.65/0.712
(b)	26.71/0.951	27.19/0.962	22.94/0.745	18.71/0.713
(c)	**26.83/0.952**	**27.26/0.961**	**23.21/0.747**	**19.04/0.716**

Table 4.2: Ablation studies on our method. Numbers are presented in the form of PSNR/SSIM. (a), (b) and (c) denote three different methods introduced in Section 4.5.5

inference counterpart. This issue can be alleviated via a more efficient implementation of test-time training.

4.5.5 Ablation Studies

All our ablation studies are conducted using GDN [76] (baseline). In order to illustrate the effectiveness of the meta-learning approach, we then introduce three experimental setting to reveal this fact. They are presented as the following: (a) **GDN+Helper**: where the helper network is directly used to provide the test-time supervision; (b) **GDN+Helper+joint-training**: the training of GDN simultaneously employs both Eq. (4.4.1) and Eq. (4.4.3) as the loss function, i.e., the GDN is optimized in a manner such that both dehazing and hazy reconstruction losses are minimized; (c) **GDN+Helper+meta-learning**: where we use our proposed method. The quantitative results of the three methods are presented in Table 4.2. It can be observed that without meta-training to associate the objectives of dehazing and helper networks, the helper network cannot assist the dehazing network to further converge on an unseen image.

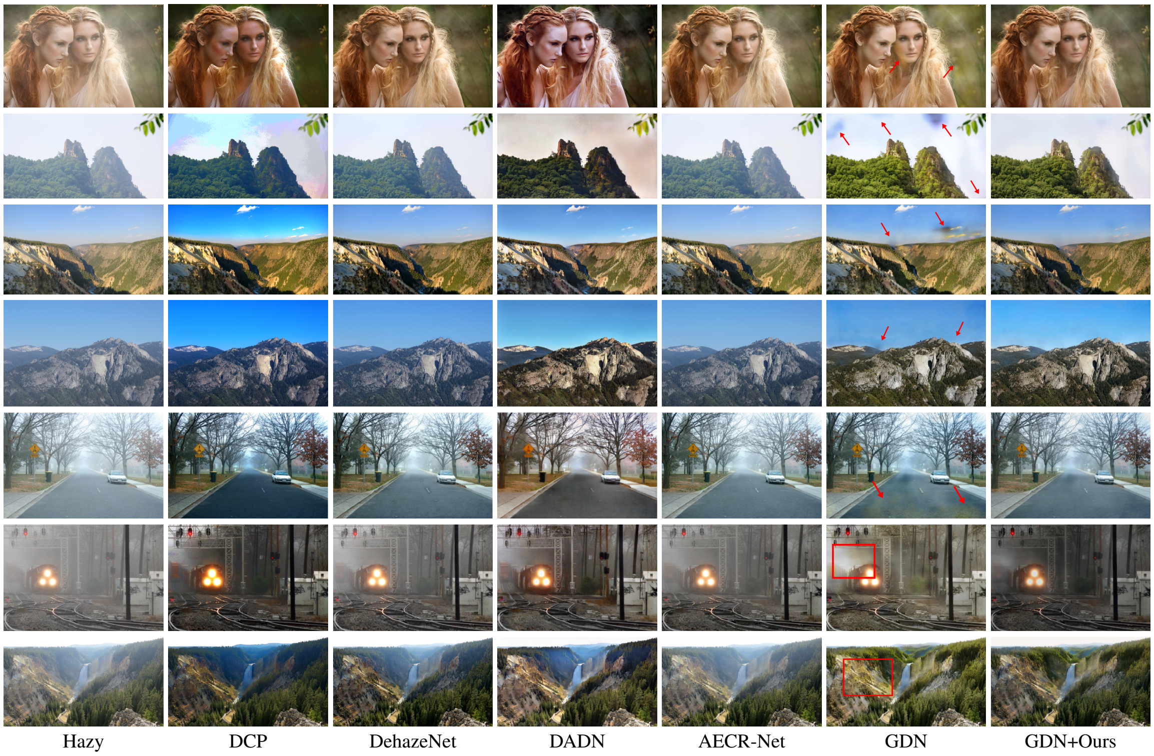


Figure 4.7: Comparison of the dehazing methods on real hazy images from fattal et.al.

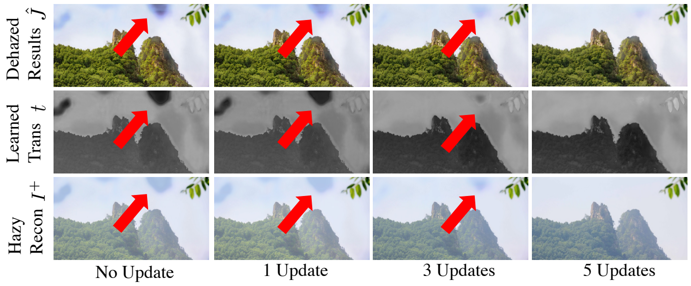


Figure 4.8: Breakdown of dehazing on a real hazy image.

4.5.6 Out-of-domain Validation on Real Scenes

For the above experiments, we assume that the hazy and haze-free images are from the same domain, while ignoring the out-of-domain (OOD) problem. Here, we take the real data as an example to validate the domain generalization ability of our proposed method. To conduct comparison, we choose four state-of-the-art single image dehazing algorithms, i.e., DCP [48], DehazeNet [17], DADN [117] and AECR-NET[139]. Note that, the training of these methods follows a common setting to use RESIDE indoor and outdoor datasets. In addition, GDN [76] is trained using the combined dataset as mentioned in Section 4.5.1.

There are two notes that should be mentioned, see Figure. 4.7. First, since GDN is trained on the combined dataset (including O-HAZE and NH-Haze datasets), its dehazed images are usually cleaner and visually pleasing. For example, the third row shows a mountain covered by haze. Here, GDN completely removes the haze from the mountain, while others cannot remove the haze effectively and suffer from color distortion. Despite this success, GDN also generates severe artifacts in the sky region. This reminds us that although multi-domain learning is beneficial in some places but it still needs further research to be employed for removing haze from all kinds of scenes. Second, comparing our outputs (GDN+Our) with those of vanilla GDN, it can be easily observed that the artifacts are gone and our dehazed images look more natural. Another example is in the last row, where GDN paints the mountains to be yellow, while our result presents a more natural color. Besides, Figure 4.8 gives a breakdown of dehazing on a real image that is presented in the second row of Figure 4.7. We can observe that the reconstructed hazy image reveals the potential issues with the dehazed image. By minimizing the hazy reconstruction loss at test time, the dehazed results and hazy reconstructions are improved simultaneously.

4.6 Conclusion

In this chapter, we reveal a critical problem that has not been considered in single image dehazing, that is, a dehazing network trained on multiple domains can perform worse than that trained only on a single domain. Based on this observation, we formulate the problem into a multi-domain learning setup, where a single model should be designed carefully to perform well on multiple domains. To address this issue, we propose a helper network to provide self-supervision to the dehazing network and improve its performance during the test time. A meta-learning approach has also been introduced to handle the problem that the supervision signal from the helper network cannot always help the dehazing network to gain an improved performance. Extensive experiments and analyses strongly support both our observation and the proposed method.

The following chapter is a reproduction of an published paper:

Huan Liu*, George Zhang*, Jun Chen, and Ashish Khisti. ”Lossy Compression with Distribution Shift as Entropy Constrained Optimal Transport. International Conference on Learning Representations, 2022.

Contribution Declaration: Huan Liu (the author of this thesis) is the first author and main contributor of this article (more than 70%). He proposed the method, conducted experiments and composed the article. George Zhang helped write and made substantial changes to a few sections of the paper; Prof. Khisti helped polish this paper and participated in the discussion; Prof. Jun Chen is the supervisor of Huan Liu.

Chapter 5

Lossy Compression with Distribution Shift as Entropy Constrained Optimal Transport

5.1 Abstract

We study an extension of lossy compression where the reconstruction distribution is different from the source distribution in order to account for distributional shift due to processing. We formulate this as a generalization of optimal transport with an entropy bottleneck to account for the rate constraint due to compression. We provide expressions for the tradeoff between compression rate and the achievable distortion with and without shared common randomness between the encoder and decoder. demonstrate We study the examples of binary, uniform and Gaussian sources (in an asymptotic setting) in detail and demonstrate that shared randomness can

strictly improve the tradeoff. For the case without common randomness and squared-Euclidean distortion, we show that the optimal solution partially decouples into the problem of optimal compression and transport and also characterize the penalty associated with fully decoupling them. We provide experimental results by training deep learning end-to-end compression systems for performing denoising on SVHN and super-resolution on MNIST, and demonstrate consistency with our theoretical results.

5.2 Introduction

Using deep neural networks for lossy image compression has proven to be effective, with rate-distortion performance capable of dominating general-purpose image codecs like JPEG, WebP or BPG [1, 84, 109]. More recently, many of these works have included generative aspects within the compression to synthesize realistic elements when the rate is otherwise too low to represent fine-grained details [2, 85, 128]. Though this has been found to deteriorate rate-distortion performance, it has generally resulted in more perceptually-pleasing image reconstruction by reducing artifacts such as pixelation and blur. Using a distributional constraint as a proxy for perceptual measure, several works have subsequently formalized this in a mathematical framework known as the rate-distortion-perception tradeoff [15, 16, 80, 81, 127, 144, 148]. As is conventional in lossy compression, these works address the scenario in which both low distortion, whereby each individual image reconstruction resembles the ground truth image, and closeness in distribution in which it is not easy to discriminate between image samples from the data-generating distribution and reconstruction distribution, are desirable.

The underlying ideal in conventional compression systems is to have perfect reconstruction with respect to some ground truth input. However this is not the case in applications such as denoising, deblurring, or super-resolution (SR), which require restoration from a degraded input image. In fact, in these cases a ground truth may not even be available. In such applications naturally the reconstruction distribution must match the original source rather than the degraded input distribution. A large body of literature has been devoted to various image restoration tasks, including several methods based on deep learning including both supervised (e.g., [15]) and unsupervised learning methods (e.g., [134]). Although most of the literature exclusively treats compression and restoration separately, in many application they can co-occur. For example, the encoder which records a degraded image may not be co-located with the decoder, but must transmit a compressed version of the image over a digital network. In turn, the decoder must perform both decompression and restoration simultaneously.

To that end, we study an extension of lossy compression in which the reconstruction distribution is different than the source distribution to account for distributional shift due to processing. The problem can be described as a transformation from some source domain to a new target domain under a rate constraint, which generalizes optimal transport. This readily extends other works which view image restoration under the perception-distortion tradeoff [15] or under optimal transport [134]. It also provides a generalization of the rate-distortion-perception problem [16] where the reconstruction distribution must be close to the input distribution. Following [126, 127], we also utilize *common randomness* as a tool for compression in our setting. Our results are summarized as follows:

- We provide a formulation of lossy compression with distribution shift as a generalization of optimal transport with an entropy constraint and identify the tradeoff between the compression rate and minimum achievable distortion both with and without common randomness at the encoder and decoder. We identify conditions under which the structure of the optimal solution partially decouples the problems of compression and transport, and discuss their architectural implications. We study the examples of binary, uniform and Gaussian sources (in asymptotic regime) in detail and demonstrate the utility of our theoretical bounds.

- We train deep learning end-to-end compression systems for performing super-resolution on MNIST and denoising on SVHN. Our setup is *unsupervised* and to the best of our knowledge the first to integrate both compression and restoration at once using deep learning. We first demonstrate that by having common randomness at the encoder and decoder the achievable distortion-rate tradeoffs are lower than when such randomness is not present. Furthermore, we provide experimental validation of the architectural principle suggested by our theoretical analysis.

5.3 Theoretical Formulation

We consider a setting where an input $X \sim p_X$ is observed at the encoder, which is a degraded (e.g., noisy, lower resolution, etc) version of the original source. It must be restored to an output $Y \sim p_Y$ at the decoder, where p_Y denotes the target distribution of interest. For example, if X denotes a noise-corrupted image and Y denotes the associated clear reconstruction, then p_Y can be selected to match the distribution of the original source. We will assume p_X and p_Y are probability distributions over $\mathcal{X}, \mathcal{Y} \subseteq \mathbb{R}^n$ and require X and Y to be close with respect to some fidelity metric, which will be measured using a non-negative cost function $d(x, y)$ over $\mathcal{X} \times \mathcal{Y}$. We will refer to $d(\cdot, \cdot)$ as the distortion measure and assume that it satisfies $d(x, y) = 0$

if and only if $x = y$. We further assume that X cannot be directly revealed to the decoder, but instead must be transmitted over a bit interface with an average rate constraint. Such a scenario occurs naturally in many practical systems when the encoder and decoder are not co-located such as communication systems or storage systems. As one potential application, when aerial photographs are produced for remote sensing purposes, blurs are introduced by atmospheric turbulence, aberrations in the optical system and relative motion between camera and ground. In such scenarios unsupervised restoration is preferred as it is often intractable to accurately model such degradation processes and collection of paired training data can be time consuming or require significant human intervention. Unsupervised image restoration has been studied recently in [83, 93, 134, 151]. These works also fix the reconstruction distribution $Y \sim p_Y$ and propose to minimize a distortion metric between the output and the degraded input as in our present work, but do not consider compression.

5.3.1 Optimal Transport and Extensions

Definition 1 (Optimal Transport)**.** *Let $\Gamma(p_X, p_Y)$ be the set of all joint distributions $p_{X,Y}$ with marginals p_X and p_Y. The classical optimal transport problem is defined as*

$$D(p_X, p_Y) = \inf_{p_{X,Y} \in \Gamma(p_X, p_Y)} d(X, Y), \tag{5.3.1}$$

where we refer to each $p_{X,Y} \in \Gamma(p_X, p_Y)$ as a transport plan.

Operationally the optimal transport plan in (5.3.1) minimizes the average distortion between the input and output while keeping the output distribution fixed to p_Y. This may generate a transport plan with potentially unbounded entropy, which may not be amenable in a rate-constrained setting. We therefore suggest a generalization to Definition 1 which constrains the entropy of the transport plan. It turns out that having common randomness at the encoder and decoder can help in this setting, so we will distinguish between when it is available and unavailable.

Definition 2 (Optimal Transport with Entropy Bottleneck — no Common Randomness)**.** *Let $M_{\mathrm{ncr}}(p_X, p_Y)$ denote the set of joint distributions $p_{X,Z,Y}$ compatible with the given marginal distributions p_X, p_Y satisfying $p_{X,Z,Y} = p_X p_{Z|X} p_{Y|Z}$. The optimal transport from p_X to p_Y with an entropy bottleneck of R and without common randomness is defined as*

$$\begin{aligned} D_{\mathrm{ncr}}(p_X, p_Y, R) \triangleq & \inf_{p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)} \mathbb{E}[d(X, Y)] \\ & s.t. \quad H(Z) \leq R, \end{aligned} \tag{5.3.2}$$

where $H(\cdot)$ denotes the Shannon entropy of a random variable.

We note that when the rate constraint R is sufficiently large such that one can select $Z = X$ or $Z = Y$ in (5.3.2), then $D_{\mathrm{ncr}}(p_X, p_Y, R) = D(p_X, p_Y)$ in (5.3.1). More generally, $D(p_X, p_Y)$ serves as a lower bound for $D_{\mathrm{ncr}}(p_X, p_Y, R)$ for any $R > 0$. Definition 2 also has a natural operational interpretation in our setting. We can view the encoder as implementing the conditional distribution $p_{Z|X}$ to output a representation Z given the input X, and the decoder as implementing the conditional distribution $p_{Y|Z}$ to output the reconstruction Y given the representation Z. The entropy constraint $H(Z) \leq R$ essentially guarantees that the representation Z can be losslessly transmitted at a rate close to R[1].

[1] The source coding theorem guarantees that any discrete random variable Z can be losslessly compressed using a variable length code with average length of no more than $H(Z) + 1$ bits.

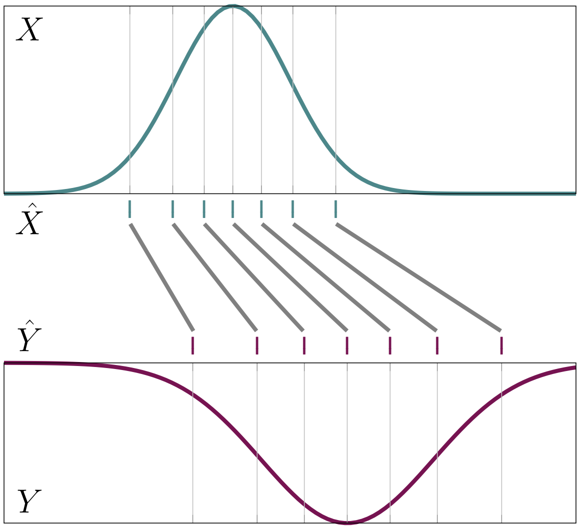


Figure 5.1: Illustration of Theorem 1 (no common randomness). Given source distribution p_X, target reconstruction distribution p_Y and rate R, we can find quantizations $\hat{X}$ of X and $\hat{Y}$ of Y and consider transport between them.

It turns out that when we specialize to the squared Euclidean distance, we can without loss of optimality impose a more structured architecture for implementing the encoder and the decoder. Let $W_2^2(\cdot,\cdot)$ be the squared quadratic Wasserstein distance, by setting $d(X,Y) = ||X-Y||^2$ in Definition 1.

Theorem 1. *Let*

$$\begin{aligned} D_{\text{mse}}(p_X, p_Y, R) \triangleq \inf_{p_{\hat{X}|X}, p_{\hat{Y}|Y}} \; & \mathbb{E}[\|X-\hat{X}\|^2] + \mathbb{E}[\|Y-\hat{Y}\|^2] + W_2^2(p_{\hat{X}}, p_{\hat{Y}}) \\ s.t. \quad & \mathbb{E}[X|\hat{X}] = \hat{X}, \quad \mathbb{E}[Y|\hat{Y}] = \hat{Y}, \quad H(\hat{X}) \le R, \quad H(\hat{Y}) \le R, \end{aligned} \tag{5.3.3}$$

and

$$\begin{aligned} D_{\text{mse}}(p_X, R) \triangleq \inf_{p_{\hat{X}|X}} \; & \mathbb{E}[\|X-\hat{X}\|^2] \\ s.t. \quad & H(\hat{X}) \le R. \end{aligned} \tag{5.3.4}$$

Moreover, let

$$\overline{D}_{\mathrm{ncr}}(p_X, p_Y, R) \triangleq D_{\mathrm{mse}}(p_X, R) + D_{\mathrm{mse}}(p_Y, R) + W_2^2(p_{\hat{X}^*}, p_{\hat{Y}^*}), \tag{5.3.5}$$

$$\underline{D}_{\mathrm{ncr}}(p_X, p_Y, R) \triangleq D_{\mathrm{mse}}(p_X, R) + D_{\mathrm{mse}}(p_Y, R), \tag{5.3.6}$$

where $p_{\hat{X}^*}$ *and* $p_{\hat{Y}^*}$ *are the marginal distributions induced by the minimizers* $p_{\hat{X}^*|X}$ *and* $p_{\hat{Y}^*|Y}$ *that attain* $D_{\mathrm{mse}}(p_X, R)$ *and* $D_{\mathrm{mse}}(p_Y, R)$*, respectively (assuming the existence of such minimizers). Then under the squared Eucledian distortion measure,*

$$D_{\mathrm{ncr}}(p_X, p_Y, R) = D_{\mathrm{mse}}(p_X, p_Y, R). \tag{5.3.7}$$

In addition, we have

$$\overline{D}_{\mathrm{ncr}}(p_X, p_Y, R) \geq D_{\mathrm{ncr}}(p_X, p_Y, R) \geq \underline{D}_{\mathrm{ncr}}(p_X, p_Y, R), \tag{5.3.8}$$

and both inequalities are tight when $p_X = p_Y$*.*

Theorem 1 deconstructs Z into the quantizations $\hat{X}$ of X and $\hat{Y}$ of Y, and decomposes the overall distortion in (5.3.2) in terms of the losses due to quantization, transport, and dequantization in (5.3.3). It also suggests a natural architecture that partially decouples compression and transport without loss of optimality. First, the sender uses the distribution $p_{\hat{X}|X}$ to produce the compressed representation $\hat{X}$ from X. This is then passed through a "converter" $p_{\hat{Y}|\hat{X}}$ to transform $\hat{X}$ to an optimal representation $\hat{Y}$ of Y. Finally, the receiver maps $\hat{Y}$ back to Y using the conditional distribution $p_{Y|\hat{Y}}$. This is illustrated in Figure 5.1. The entropy constraint $H(\hat{X}) \leq R$ in (5.3.2) essentially guarantees that $\hat{X}$ can be losslessly transmitted to the decoder where the converter can be applied to map $\hat{X}$ to $\hat{Y}$ before outputting Y. Alternately the constraint $H(\hat{Y}) \leq R$ guarantees that the converter could also be implemented at the encoder and then $\hat{Y}$ can be compressed and transmitted to the decoder. Finally note that our proposed architecture is symmetric[2] with respect to the encoder and the decoder and in particular the procedure to transport Y to X would simply be the inverse of transporting X to Y, and indeed the distortion incurred by dequantizing $p_{Y|\hat{Y}}$ is the same as the distortion incurred by quantizing $p_{\hat{Y}|Y}$.

For the special case of same source and target distribution, we have $D_{\text{mse}}(p_X, p_X, R) = 2D_{\text{mse}}(p_X, R)$, implying that the rate required to achieve distortion D under no output distribution constraint (and with the output alphabet relaxed to $\mathbb{R}^n$) achieves distortion $2D$ under the constraint that Y equals X in distribution. This recovers the result of Theorem 2 in [144] for the one-shot setting. More generally, (5.3.8) shows that we may lower bound $D_{\text{mse}}(p_X, p_Y, R)$ by the distortion incurred when compressing X and Y individually, each at rate R, through ignoring the cost of transport. On the other hand, the upper bound corresponds to choosing the optimal rate-distortion representations $\hat{X}^*, \hat{Y}^*$ for X, Y, then considering transport between them. The advantage of this approach is that knowledge of the other respective distribution is not necessary for design. Although not optimal in general, we will, in fact, provide an example where this is optimal in Section 5.3.2.

Finally, the following result implies that under mild regularity conditions, the optimal converter $p_{\hat{Y}|\hat{X}}$ can be realized as a (deterministic) bijection, and in the scalar case it can basically only take the form as illustrated in Figure 5.1.

[2] We say that the problem is symmetric if it is invariant under reversing p_X, p_Y with a new distortion measure defined by reversing the arguments of $d(\cdot, \cdot)$.

Theorem 2. *Assume that $D_{\mathrm{ncr}}(p_X, p_Y, R)$ is a strictly decreasing function in a neighborhood of $R = R^*$ and $D_{\mathrm{ncr}}(p_X, p_Y, R^*)$ is attained by $p_{X,Z,Y}$. Let $\hat{X} \triangleq \mathbb{E}[X|Z]$ and $\hat{Y} \triangleq \mathbb{E}[Y|Z]$. Then*

$$H(\hat{X}) = H(\hat{Y}) = R^*, \tag{5.3.9}$$

$$\mathbb{E}[\|\hat{X} - \hat{Y}\|^2] = W_2^2(p_{\hat{X}}, p_{\hat{Y}}), \tag{5.3.10}$$

and there is a bijection between $\hat{X}$ and $\hat{Y}$.

We remark that in general computing the optimal transport map is not straightforward. For the case of binary sources we can compute an exact characterization for D_{ncr} as discussed in Section 5.3.2. Furthermore as discussed in Appendix A.1.6, $W_2^2(p_{\hat{X}}, p_{\hat{Y}})$ can be computed in closed form when $\hat{X}$ and $\hat{Y}$ are scalar valued, which can be used to obtain upper bounds on D_{ncr}. In our experimental results in Section 5.4 we use deep learning based methods to learn approximately optimal mappings.

So far we have focused on the setting when there is no shared common randomness between the encoder and the decoder. We will now consider the setting when a shared random variable denoted by U is present at the encoder and decoder. We assume that the variable U is independent of the input X so that the decoder has no apriori information of the input. In practice the sender and receiver can agree on a pseudo-random number generator ahead of time and some kind of seed could be transmitted, after which both sides can generate the same U. We further discuss how shared randomness is used in practice in the experimental section.

Definition 3 (Optimal Transport with Entropy Bottleneck — with Common Randomness)**.** *Let $M_{\mathrm{cr}}(p_X, p_Y)$ denote the set of joint distributions $p_{U,X,Z,Y}$ compatible with*

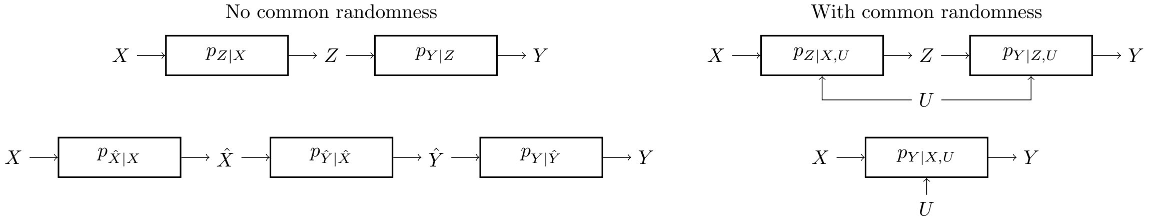


Figure 5.2: Architectures. Top left: Definition 2. Bottom left: Theorem 1. Top right: Definition 3. Bottom right: Theorem 3. Entropy coding of intermediate representations $Z, \hat{X}, \hat{Y}$ is not shown. For Theorem 3, the division between sender and receiver is across an encoder $C = f(X, U)$ and decoder $Y = g(C, U)$ performing entropy coding along $p_{Y|X,U}$.

the given marginal distributions p_X, p_Y and satisfying $p_{U,X,Z,Y} = p_U p_X p_{Z|X,U} p_{Y|Z,U}$, where p_U represents the distribution of shared randomness. The optimal transport from p_X to p_Y with entropy bottleneck R and common randomness is defined as

$$\begin{aligned} D_{\mathrm{cr}}(p_X, p_Y, R) \triangleq & \inf_{p_{U,X,Z,Y} \in M_{\mathrm{cr}}(p_X, p_Y)} \mathbb{E}[d(X,Y)] \\ & s.t. \quad H(Z|U) \leq R. \end{aligned} \tag{5.3.11}$$

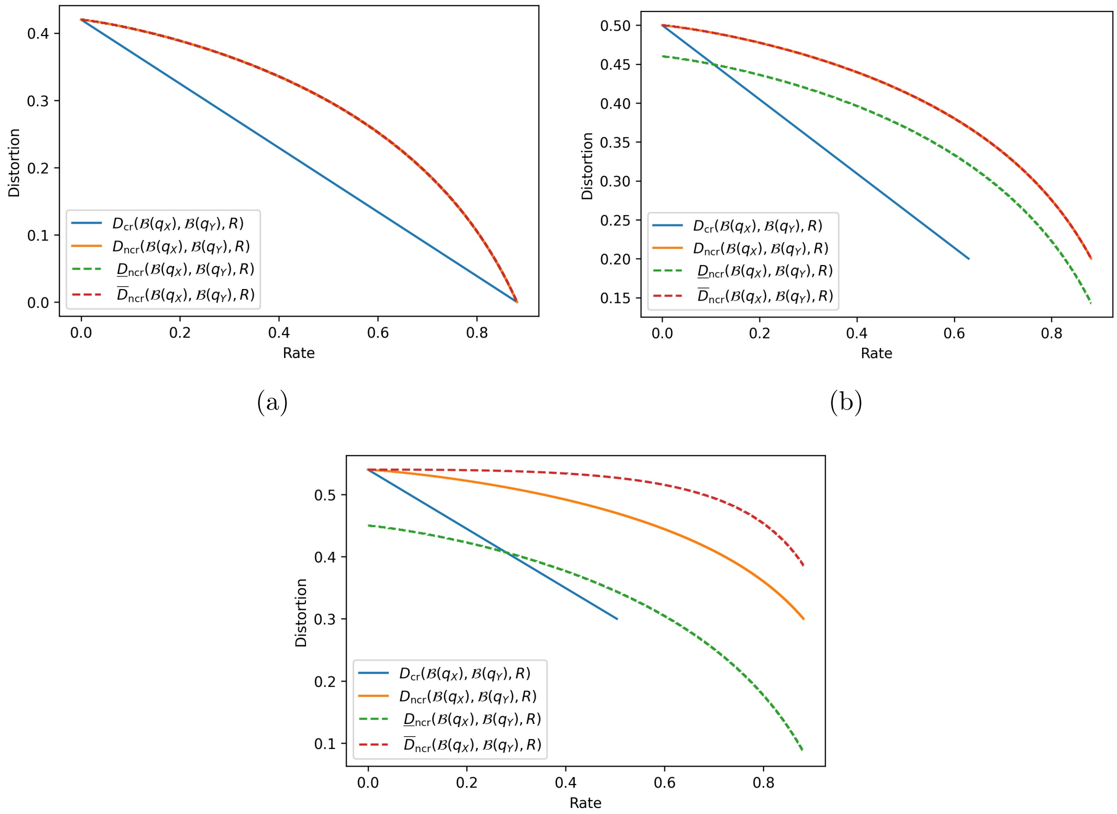
Note that we optimize over p_U (the distribution associated with shared randomness), in addition to $p_{Z|X,U}$ and $p_{Y|Z,U}$ in (5.3.11). Furthermore, $D_{\mathrm{cr}}(p_X, p_Y, R) \leq D_{\mathrm{ncr}}(p_X, p_Y, R)$ in general, as we do not have access to shared randomness in Definition 2. Also from the same argument that was made following Definition 2, we have that $D_{\mathrm{cr}}(p_X, p_Y, R) \geq D(p_X, p_Y)$ in Definition 1. As with Definition 2, we can also provide a natural operational interpretation. In particular, given the input X and common randomness U the encoder can output a compressed representation Z using the conditional distribution $p_{Z|X,U}$. The representation Z can be losslessly compressed approximately to an average rate of R again by exploiting the shared randomness U. Finally the decoder, given Z and U can output the reconstruction Y using the conditional distribution $p_{Y|Z,U}$. An interesting difference with Definition 2 is that the setup is no longer symmetric between encoder and decoder, as X is independent of U but Y is not. The following result provides a simplification to the architecture in Definition 3.

Theorem 3. *Let $Q_{cr}(p_X, p_Y)$ denote the set of joint distributions $p_{U,X,Y}$ compatible with the given marginals p_X, p_Y satisfying $p_{U,X,Y} = p_U p_X p_{Y|U,X}$ as well as $H(Y|U,X) = 0$. Then*

$$\begin{aligned} D_{\mathrm{cr}}(p_X, p_Y, R) = &\inf_{p_{U,X,Y} \in Q_{cr}(p_X, p_Y)} \mathbb{E}[d(X,Y)] \\ &\quad s.t. \quad H(Y|U) \leq R. \end{aligned} \tag{5.3.12}$$

Before discussing the implications of Theorem 3 we remark on a technical point. Because the Shannon entropy is defined only for discrete random variables, U must be chosen in a way such that $Y|U = u$ is discrete for each u, even for continuous (X, Y). This is known to be possible, e.g., [70] have provided a general construction for a U with this property, with additional quantitative guarantees to ensure that U is informative of Y. In the finite alphabet case we show in Appendix A.1.3 that optimization of U can be formulated as a linear program.

We next discuss the implication of Theorem 3. First note that the problem can be modelled with only $p_{Y|U,X}$ producing a reconstruction Y without the need for the intermediate representation Z, much like the conventional optimal transport in Definition 1. The condition $H(Y|U,X) = 0$ also implies that the transport plan is deterministic when conditioned on the shared randomness, which plays the role of stochasticity. Furthermore in this architecture the encoder should compute the representation Y given the source X and the shared random-variable U (which corresponds to the transport problem) and then compress it losslessly at a rate close



(a)

(b)

(c)

Figure 5.3: Binary case distortion-rate tradeoffs. (a) $q_X = q_Y = 0.3$, where $\overline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),R)$ and $\underline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),R)$ coincide with $D_{\mathrm{ncr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),R)$; (b) $q_X = 0.3, q_Y = 0.5$, where $\overline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),R)$ is tight but $\underline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),R)$ is loose; (b) $q_X = 0.3, q_Y = 0.6$, where both bounds are loose. Moreover, it can be seen from all these examples that common randomness can indeed help improve the distortion-rate tradeoff.

to $H(Y|U)$ (which corresponds to the compression problem). The receiver only needs to decompress and reconstruct Y. This is in contrast to the case without common randomness in Theorem 1 where the reconstruction Y must be generated at the decoder.

5.3.2 Numerical Examples

We present how the results in Theorem 1 & 3 can be evaluated for some specific source models. We first consider the example of Binary sources. Let $X \sim \mathcal{B}(q_X)$ and $Y \sim \mathcal{B}(q_Y)$ be two Bernoulli random variables with $q_X, q_Y \in (0,1)$, and let $d(\cdot,\cdot)$ be the Hamming distortion measure $d_H(\cdot,\cdot)$ (i.e., $d_H(x,y) = 0$ if $x = y$ and $d_H(x,y) = 1$ otherwise), which coincides with the squared error distortion in Theorem 1 for binary variables. The explicit expressions of $D_{\mathrm{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$, $D_{\mathrm{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$ as well as $\overline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$ and $\underline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$ are provided by Theorem 4 in Appendix A.1.4, from which the following observations can be made. In general, we have $D_{\mathrm{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R) > D_{\mathrm{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$, i.e., common randomness strictly improves the distortion-rate tradeoff (except at some extreme point(s)). Moreover, as long as $\mathcal{B}(q_X)$ and $\mathcal{B}(q_Y)$ are biased toward the same symbol (namely, $q_X, q_Y \leq 1/2$ or $q_X, q_Y \geq 1/2$), the upper bound $\overline{D}_{\mathrm{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$ is tight, which implies that blindly using optimal quantizer and dequantizer in the conventional rate-distortion sense incurs no penalty, and the cross-domain knowledge is only needed for optimal transport from the quantizer output and the dequantizer input. Some illustrative examples are shown in Figure 5.3.

In Appendix A.1.6 we consider the case when X and Y are continuous valued sources from a uniform distribution and establish an upper bound on $D_{\mathrm{ncr}}(\cdot)$ that is shown to be tight as the rate $R \to \infty$. For Gaussian distributions in the asymptotic optimal transport setting (see Appendix A.1.7 for relevant definitions and results) we present results qualitatively similar to the binary case in Appendix A.1.8.

5.4 Experimental Results

We use two sets of results to illustrate that the principles derived from our theoretical results are applicable to practical compression with deep learning. Importantly, we assume an *unsupervised* setting in which we have only unpaired noisy and clean images available to us, as in [134]. Our first experiment is, to the best of our knowledge, the first in which restoration and compression are performed jointly using deep learning. We will furthermore demonstrate the utility of common randomness in this setting. Stochasticity was necessary in our theoretical results both with and without common randomness. In practice, we will use generative models to induce domain shift, and stochasticity is necessary to train an effective (rate-constrained) generative model at low rates to produce variety in the reconstructions. The first set of experiments compare the rate-distortion tradeoffs achieved by quantization schemes with and without common randomness using a rate constrained architecture designed for image denoising and super resolution tasks.

The second set of experiments are designed on the principle of Theorem 1. In addition to the generator trained from our first experiment, we will construct a helper network to allow us to estimate the decomposition (5.3.3). This is then compared with the direct loss between the noisy image and rate-constrained denoising reconstruction. If the losses are close, this would suggest that the decomposition is not only without loss of optimality but also effective in practice.

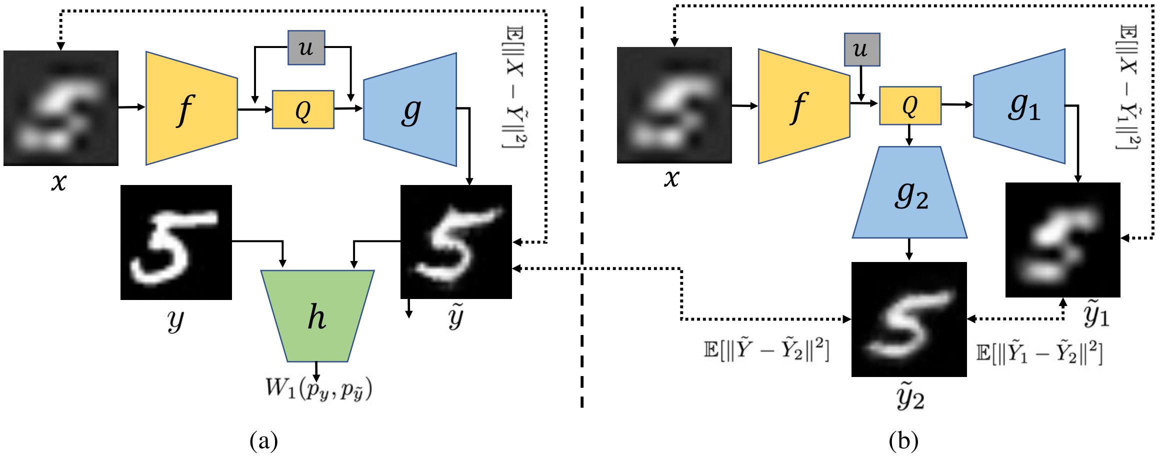

u
f
Q
g
$\mathbb{E}[\|X - \tilde{Y}\|^2]$
x
y
h
$\tilde{y}$
$W_1(p_y, p_{\tilde{y}})$
(a)
u
f
Q
g_1
g_2
$\mathbb{E}[\|X - \tilde{Y}_1\|^2]$
x
$\tilde{y}_1$
$\mathbb{E}[\|\tilde{Y} - \tilde{Y}_2\|^2]$
$\mathbb{E}[\|\tilde{Y}_1 - \tilde{Y}_2\|^2]$
$\tilde{y}_2$
(b)

Figure 5.4: Illustration of our experimental setup. (a) shows the end-to-end learning system with common randomness, where the encoder and decoder have access to the same randomness u. (b) presents the network setup for verifying the architecture principle given in Theorem 1.

5.4.1 Rate-Distortion Comparison with Common Randomness

Let p_X be a degraded source distribution that we wish to restore and p_Y be the target distribution. Our goal is to compress X so that the reconstruction semantically resembles X within target distribution p_Y. For our application, we will use MSE loss as a fidelity criterion. Let f be an encoder, Q a quantizer, and g a decoder. For a given rate R with common randomness U available, we have a problem of the form

$$\begin{aligned} \min_{f,g,Q} \quad & \|X - g(Q(f(X,U)))\|_2^2 \\ \text{s.t.} \quad & p_{g(Q(f(X,U)))} = p_Y, \quad H(Q(f(X,U))|U) \leq R, \end{aligned}$$

which uses parameterized neural networks to implement (5.3.11). We also fix Q such that a hard constraint on the rate is satisfied and assume f and g are sufficiently expressive to map to these fixed quantization points. Let $\tilde{Y} \triangleq g(Q(f(X, U)), U)$. We will use a penalty on the Wasserstein-1 distance between $p_{\tilde{Y}}$ and p_Y in accordance with the Wasserstein GAN [11] framework, so that our system is a stochastic rate-constrained autoencoder with GAN regularization. Specifically, we follow the network shown in Figure 5.4(a) which in addition to f, Q, and g contains critic h.

For the realization of common randomness in Definition 3, we adopt the universal quantization scheme of [126, 162]. Given trained f and g and degraded image X, we generate restored image $\tilde{Y}$ through

$$\tilde{Y} = g(Q(f(X) + U) - U), \tag{5.4.1}$$

where U is the stochastic noise shared by the sender and receiver. Details about the quantization are provided in Appendix A.2.2. To find an appropriate f and g, we use the relaxed objective

$$L_1 = \mathbb{E}[\|X - \tilde{Y}\|^2] + \lambda W_1(p_Y, p_{\tilde{Y}}), \tag{5.4.2}$$

which is the sum of the MSE and Wasserstein-1 losses weighted by λ. By optimizing our network using this objective, we see two favorable properties. First, the Wasserstein-1 loss ensures the distribution of output is close to that of target images, i.e. $p_{\tilde{Y}} \approx p_Y$ for sufficiently large λ. Moreover, the MSE loss that pushes the output $\tilde{Y}$ to input X ensures that the output structurally resembles X. Consequently, the training objective allows the output $\tilde{Y}$ to be clear and preserves content from input.

To generate a rate-distortion trade-off curve, we modify the encoder to produce a different number of symbols ranging from low bit rate to high bit rate and record the

MSE distortion loss between noisy inputs and denoised outputs. Figure 5.5(a) and Figure 5.5(c) show the curves for image super-resolution and image denoising. We also show some qualitative results in Figure 5.5(b) and 5.5(d). As the rate increases, the generated high-quality images are clearer.

As exemplified by the numerical results in Section 5.3.2, common randomness can help reduce the rate that is needed for reconstruction given a specific distortion. Equivalently, given a fixed rate, a system with common randomness can perform better than one without common randomness. To demonstrate this in practice, we conduct the following experiment. We remove the common randomness setup from the framework in Section 5.4.1 and alternatively add two independent noises U_1 and U_2 to the encoder and decoder sides. Concretely, under the new setting, (5.4.1) becomes

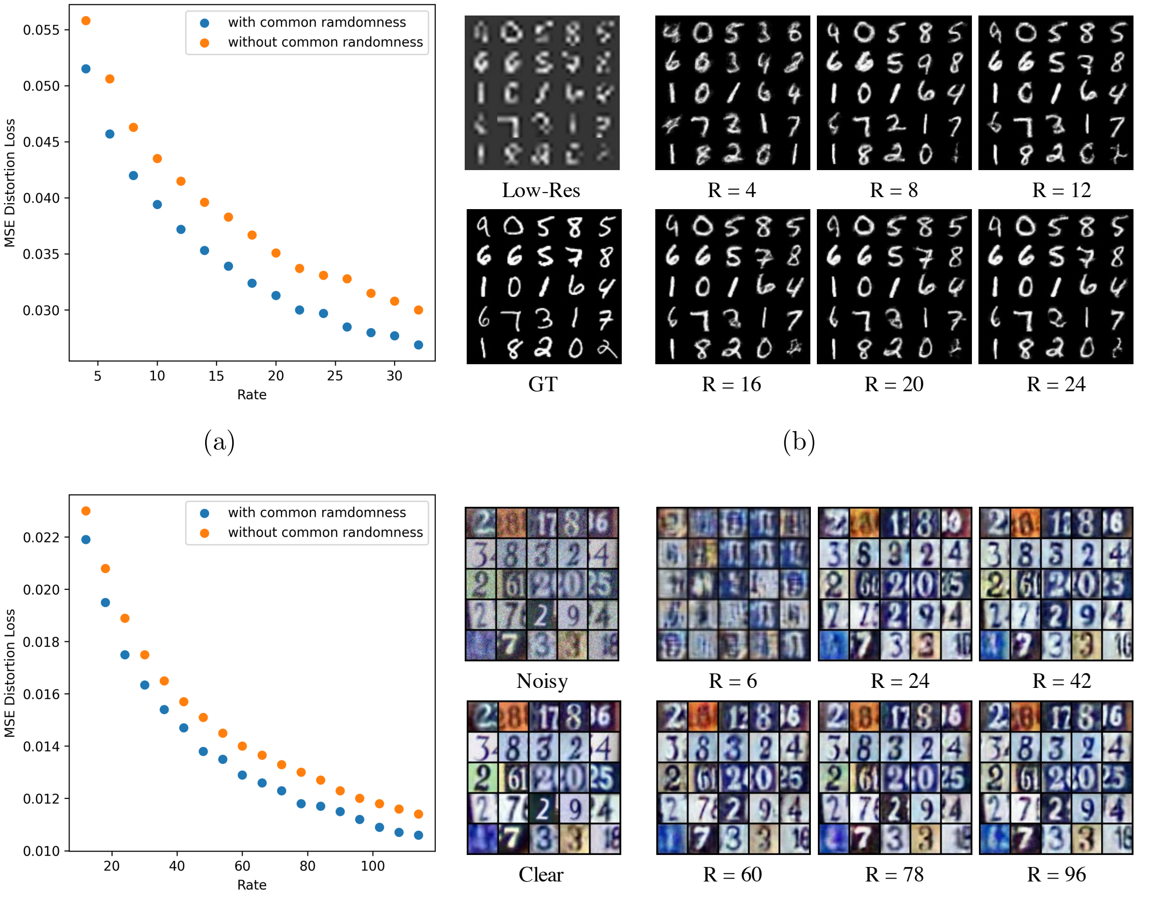
$$\tilde{Y} = g(Q(f(X) + U_1) - U_2) \tag{5.4.3}$$

Then we conduct training using the objective (5.4.2) as in the common randomness. The tradeoff curve without common randomness for both tasks are shown in Figure 5.5(a) and 5.5(c) with orange dots. Performance of the framework is better when there is common randomness.

5.4.2 Architectural Principle

In the case without common randomness, Theorem 1 implies that (under the rate constraint) the overall distortion $\mathbb{E}[\|X - Y\|^2]$ can be decomposed to the summation of the three distortion terms

$$\mathbb{E}[\|X - Y\|^2] = \mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + W_2^2(p_{\hat{X}}, p_{\hat{Y}}), \tag{5.4.4}$$



(a) (b)

(c) (d)

Figure 5.5: (a)(b) The experimental results of 4 times image super-resolution. (c)(d) The experimental results of image denoising. The noise pattern is synthesized by additive Gaussian noise with standard deviation set to 20. (a)(c) Rate-distortion trade-offs. Blue points are the MSE distortion loss for a particular rate under the setting of using common randomness, while orange points illustrate the same trade-off without using common randomness. For both tasks, at any rate, the performance of using common randomness is better than the case without common randomness. (b)(d) Examples for outputs from several models with different rates. As the rate increases, the outputs become clearer.

where $\hat{X}$ and $\hat{Y}$ are some representations of X and Y under MSE distortion. The chosen rate-distortion representations $\hat{X}$ for X and $\hat{Y}$ for Y must not only be representative of X and Y, but also enjoy low cost of transport between one another. We now seek to estimate the overhead of this decomposition in practice.

However, due to the nature of the deep learning framework, the distortion measure between images and compressed representations cannot be explicitly measured. Thus, we alternatively develop a two-branch network to compare the summation of the three distortion components (5.4.4) to the overall distortion. First, we take trained f and g from the previous experiment and freeze their weights. Given noisy input X, we encode it through f, then decoder g_1 is trained to minimize the distortion with X, and decoder g_2 is trained to minimize the distortion with $\tilde{Y} = f(g(X))$ which distributionally approximates a clean restoration (we use $\tilde{Y}$ instead of ground truth because we assume an unsupervised setting). Let

$$\tilde{Y_1} = g_1(f(X)), \quad \tilde{Y_2} = g_2(f(X)).$$

The idea here is that $\tilde{Y_1}$ is a rate-constrained reconstruction of X and $\tilde{Y_2}$ is a rate-constrained reconstruction of $\tilde{Y}$, both of which are produced from compressing X using f. We assume this is reasonable as in light of Theorem 2, there is no loss of optimality in doing so given sufficiently expressive neural networks. The overall decomposed loss is then given by

$$L_2 = \underbrace{\mathbb{E}[\|X - \tilde{Y_1}\|^2]}_{\text{(a)}} + \underbrace{\mathbb{E}[\|\tilde{Y} - \tilde{Y_2}\|^2]}_{\text{(b)}} + \underbrace{\mathbb{E}[\|\tilde{Y_1} - \tilde{Y_2}\|^2]}_{\text{(c)}}, \tag{5.4.5}$$

in which $\tilde{Y_1}$ approximates $\hat{X}$ and $\tilde{Y_2}$ approximates $\hat{Y}$. Training is performed jointly over g_1 and g_2.

One additional point is that g_1 and g_2 can also be trained separately, although in this case we can no longer assume that f can be reused without loss of optimality,

Table 5.1: Results of architecture principle in Theorem 1. The end-to-end loss and decomposed loss are very close for different rates.

	Super-resolution				Denoising			
Rate	4	10	20	30	12	30	60	90
End-to-end Loss	0.0558	0.0435	0.0351	0.0308	0.0230	0.0175	0.0140	0.0123
Decomposed Loss	0.0586	0.0453	0.0349	0.0309	0.0243	0.0192	0.0158	0.0139

as this objective would not be equivalent to minimizing (5.4.5) (there is no control over (c)). We develop an additional experiment in which we optimize encoder-decoder pairs f_1, g_1 to minimize (a) and f_2, g_2 to minimize (b), where now $\tilde{Y}_1$ is produced using f_1, g_1 and $\tilde{Y}_2$ using f_2, g_2. In this setting, we aim to approximate the rate-distortion optimal $\tilde{X}^*$ and $\tilde{Y}^*$ corresponding to $\overline{D}_{\text{ncr}}(p_X, p_Y, \cdot)$ from Theorem 1 using $\tilde{Y}_1$ and $\tilde{Y}_2$, and in doing this separate optimization it is clear that we will drive down (a) and (b) but increase (c). As it turns out, the resultant values (a) and (b) obtained during joint optimization are not much worse than the values from separate optimization. This provides evidence that in practice, the optimal rate-distortion representations (i.e. under objective (5.3.4)) can be leveraged for the general objective (5.4.5) without much loss of optimality, which further suggests that the encoder f_1 can be potentially trained without knowledge of p_Y without much performance loss. These can be viewed in Table 5.2.

5.5 Related Works

Sinkhorn distances [25] are a formulation of optimal transport with a penalty term corresponding to the mutual information between the source and target distributions. This has been studied in information theory literature (e.g. [13, 132]). For source coding in particular, [113, 114] consider common randomness with constrained output

Table 5.2: Comparison of separate vs. joint training for (5.4.5). See the above paragraph for explanation.

Super-resolution

Rate	4		10		20		30	
Method	Joint	Separate	Joint	Separate	Joint	Separate	Joint	Separate
$\mathbb{E}[\|X-\tilde{Y}_1\|^2]$	0.0355	0.0356	0.0223	0.0214	0.0136	0.0113	0.0092	0.0083
$\mathbb{E}[\|\tilde{Y}-\tilde{Y}_2\|^2]$	0.0227	0.0216	0.0222	0.0206	0.0191	0.0191	0.0172	0.0155

Denoising

Rate	12		30		60		90	
Method	Joint	Separate	Joint	Separate	Joint	Separate	Joint	Separate
$\mathbb{E}[\|X-\tilde{Y}_1\|^2]$	0.0191	0.0190	0.0146	0.0145	0.0117	0.0123	0.0104	0.0107
$\mathbb{E}[\|\tilde{Y}-\tilde{Y}_2\|^2]$	0.0050	0.0046	0.0044	0.0040	0.0038	0.0035	0.0032	0.0030

distribution. [15] evaluated a number of deep image restoration techniques and somewhat counter-intuitively demonstrated a tradeoff between optimizing for distortion and "perceptual quality", i.e. realism. This is explained by the fact that the output of the conventional rate-distortion objective can differ significantly from the source distribution. [134] model the shift in distribution due to degradation as an optimal transport problem. However this work does not consider compression and their results are qualitatively different from ours. Meanwhile, output-constrained lossy compression has also been shown to improve perceptual quality [128], leading to the rate-distortion-perception framework [16].

5.6 Conclusion

We consider the setting of lossy compression in which we compress across different source and target distributions. We formulate this as an entropy-constrained optimal transport problem and provide expressions for characterizing the tradeoff between compression rate and the minimum achievable distortion with and without shared common randomness. We also develop a number of architectural principles through our theoretical results and provide experimental validations by training deep learning models for super-resolution and denoising tasks over compressed representations. On the theory side it will be interesting to consider the case where there are either rate constraints on the amount of shared common randomness between the encoder and decoder or consider the case when the shared randomness is correlated with the source input, which can arise in many practical applications. On the practical side it will be interesting to experimentally study the a broader set of *cross-domain* tasks where our theory could be applicable.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Four deep learning based approaches have been researched in this thesis for addressing several low-level vision problems, such as image dehazing, depth estimation and cross-domain lossy compression.

The first work proposes a new network structure and a training method based on the introduced indirect domain shift mechanism. In contrast to the widely used end-to-end training method, the first work shows that end-to-end training cannot establish accurate mapping on the paired dataset and proposes an indirect domain shift method. The indirect domain shift method consists of two parts. The first part is based on the idea of using a multi-scale and two-branch structure to build a neural network for achieving successful image dehazing. The second part is motivated by the observation that supervision should be placed inside the network, and two branches should have different loss functions for diversity. Adversarial loss is also included to guarantee that the network's outputs have the same distribution as that of clean images. Extensive ablation studies show that the proposed indirect domain shift is practical, and each part of this method plays an important role. The comparison with the several competitive methods further proves the effectiveness of our indirect domain shift methods.

The second work suggests that the problem of current unsupervised depth estimation methods lies in the failure to take full advantage of stereo data. To solve the problem and maintain the monocular setting, a pseudo supervision mechanism is proposed by integrating both unsupervised monocular depth estimation and unsupervised binocular depth estimation. Besides, a semantic booster and occlusion estimation are further introduced to improve depth estimation accuracy. Extensive experiments including ablation studies show the mechanism's effectiveness and illustrate that the proposed framework achieves unprecedented improvements compared with the state-of-the-art methods.

In the third work, a critical problem in single image dehazing is disclosed, that is, a dehazing network trained on multiple domains performs worse than that trained on a single domain. Based on this observation, a multi-domain dehazing problem is formulated. To address this problem, a test-time training method is proposed alongside a helper network to assist the dehazing model in further adapting to a specific scene. Concretely, the helper network is developed to evaluate the quality of dehazing results and provide effective self-supervision to the dehazing network during test time. Afterwards, the dehazing model can benefit from the feedback of the helper and update itself towards better dehazing results. In order to ensure the supervision from the helper network is always helpful for training the dehazing network in test time, a meta-learning approach is introduced to address the issue. Extensive experiments demonstrate the effectiveness of the proposed method.

The last work considers the novel task of cross-domain lossy compression, in which compression and decompression are conducted on different distributions. This task is formulated as an entropy-constrained optimal transport problem. The theoretical results provide an architectural principle when common randomness is unavailable and further suggest that common randomness can help reduce the compression rate. The experimental results also demonstrate the utility of common randomness and indicate that the architectural decomposition in the case of no common randomness allows us to use rate-distortion optimal encoders (which do not require knowledge of the target distribution) without much penalty.

6.2 Future Work

There is still space for further improving the proposed approaches in handling low-level vision tasks. Firstly, the methods in chapter 2 and chapter 3 are developed aiming to achieve successful image dehazing on the paired dataset. However, collecting paired hazy and haze-free images is extremely hard and even impossible. Interesting future work can be considered to dehaze real-world hazy images using pure unsupervised learning. Secondly, the depth estimation method proposed in chapter 3 simply masks the occlusion region between stereo images, which can result in inaccurate depth estimation in these regions. It is preferred if future work can be conducted to explore deep into the occlusion region. Thirdly, due to the resource constraint, the current experiments are conducted using four representative datasets and three popular dehazing networks. It is preferable to have a more comprehensive evaluation of our method by considering a larger collection of datasets and dehazing networks. Finally, the experiments shown in chapter 3 are based on two simple datasets, i.e., MNIST and SVHN. For future work, applying the algorithm to large-scaled datasets is necessary for further algorithm validation.

Appendix A

Appendix - Lossy Compression with Distribution Shift as Entropy Constrained Optimal Transport

A.1 Theoretical Results

A.1.1 Distortion-Rate vs Rate-Distortion Formulation

In addition to Definitions 2 and 3, we can equivalently define

$$\begin{aligned} R_{\mathrm{ncr}}(p_X, p_Y, D) \triangleq & \inf_{p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)} H(Z) \\ & \text{s.t.} \quad \mathbb{E}[d(X,Y)] \leq D, \end{aligned} \tag{A.1.1}$$

$$\begin{aligned} R_{\mathrm{cr}}(p_X, p_Y, D) \triangleq & \inf_{p_{U,X,Z,Y} \in M_{\mathrm{cr}}(p_X, p_Y)} H(Z|U) \\ & \text{s.t.} \quad \mathbb{E}[d(X,Y)] \leq D. \end{aligned} \tag{A.1.2}$$

$D_{\mathrm{ncr/cr}}(p_X, p_Y, R)$ and $R_{\mathrm{ncr/cr}}(p_X, p_Y, D)$ are monotonically decreasing in R and D, respectively, so they are the inverse of each other. Sometimes it is more convenient to work with this rate-distortion formulation.

A.1.2 Proofs of Theoretical Results

Proof of Theorem 1. For any $p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)$ with $H(Z) \leq R$,

$$\begin{aligned}\mathbb{E}[\|X-Y\|^2] &= \mathbb{E}[\|X-\mathbb{E}[X|Z]\|^2] + \mathbb{E}[\|Y-\mathbb{E}[Y|Z]\|^2] + \mathbb{E}[\|\mathbb{E}[X|Z]-\mathbb{E}[Y|Z]\|^2] \\ &\geq D_{\mathrm{mse}}(p_X, p_Y, R),\end{aligned}$$

where the last inequality follows from the definition of $D_{\mathrm{mse}}(p_X, p_Y, R)$ and the fact that

$$\max\{H(\mathbb{E}[X|Z]), H(\mathbb{E}[Y|Z]\} \leq H(Z) \leq R.$$

As a consequence, we must have $D_{\mathrm{ncr}}(p_X, p_Y, R) \geq D_{\mathrm{mse}}(p_X, p_Y, R)$. On the other hand, for any $p_{\hat{X}|X}$, $p_{\hat{Y}|Y}$ with $\mathbb{E}[X|\hat{X}] = \hat{X}$, $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$, $H(\hat{X}) \leq R$, and $H(\hat{Y}) \leq R$, we can construct a joint distribution $p_{X,\hat{X},\hat{Y},Y}$ such that $X \leftrightarrow \hat{X} \leftrightarrow \hat{Y} \leftrightarrow Y$ form a Markov chain, $p_{X,\hat{X}} = p_X p_{\hat{X}|X}$, $p_{Y,\hat{Y}} = p_Y p_{\hat{Y}|Y}$, and $p_{\hat{X},\hat{Y}}$ satisfying $\mathbb{E}[\|\hat{X}-\hat{Y}\|^2] = W_2^2(p_{\hat{X}}, p_{\hat{Y}})$. Note that

$$\begin{aligned}\mathbb{E}[\|X-Y\|^2] &= \mathbb{E}[\|X-\hat{X}\|^2] + \mathbb{E}[\|Y-\hat{Y}\|^2] + \mathbb{E}[\|\hat{X}-\hat{Y}\|^2] \\ &= \mathbb{E}[\|X-\hat{X}\|^2] + \mathbb{E}[\|Y-\hat{Y}\|^2] + W_2^2(p_{\hat{X}}, p_{\hat{Y}}). \qquad \text{(A.1.3)}\end{aligned}$$

Let $Z \triangleq \hat{X}$. It can be verified that $p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)$ and $H(Z) = H(\hat{X}) \leq R$, which, together with (A.1.3), implies $D_{\mathrm{ncr}}(p_X, p_Y, R) \leq D_{\mathrm{mse}}(p_X, p_Y, R)$. This completes the proof of (5.3.7).

Dropping the term $W_2^2(p_{\hat{X}}, p_{\hat{Y}})$ in (5.3.3) yields

$$D_{\mathrm{ncr}}(p_X, p_Y, R) \geq \tilde{D}_{\mathrm{mse}}(p_X, R) + \tilde{D}_{\mathrm{mse}}(p_Y, R),$$

where

$$\begin{aligned} \tilde{D}_{\mathrm{mse}}(p_X, R) \triangleq \inf_{p_{\hat{X}|X}} \ & \mathbb{E}[\|X - \hat{X}\|^2] \\ \text{s.t.} \quad & \mathbb{E}[X|\hat{X}] = \hat{X}, \quad H(\hat{X}) \leq R. \end{aligned}$$

and $\tilde{D}_{\mathrm{mse}}(p_Y, R)$ is definely analogously. On the other hand, choosing $p_{\hat{X}|X} = p_{\hat{X}'|X}$ and $p_{\hat{Y}|Y} = p_{\hat{Y}'|Y}$ in (5.3.3) gives

$$D_{\mathrm{ncr}}(p_X, p_Y, R) \leq \tilde{D}_{\mathrm{mse}}(p_X, R) + \tilde{D}_{\mathrm{mse}}(p_Y, R) + W_2^2(p_{\hat{X}'}, p_{\hat{Y}'}),$$

where $p_{\hat{X}'|X}$ and $p_{\hat{Y}'|Y}$ are the minimizers that attain $\tilde{D}_{\mathrm{mse}}(p_X, R)$ and $\tilde{D}_{\mathrm{mse}}(p_Y, R)$ respectively while $p_{\hat{X}'}$ and $p_{\hat{Y}'}$ are their induced marginal distributions. It is clear that $p_{\hat{X}'|X}$ and $p_{\hat{Y}'|Y}$ coincide with $p_{\hat{X}^*|X}$ and $p_{\hat{Y}^*|Y}$ respectively as the constraints $\mathbb{E}[X|\hat{X}] = \hat{X}$ and $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$ are automatically satisfied by $p_{\hat{X}^*|X}$ and $p_{\hat{Y}^*|Y}$. This proves (5.3.8). For the special case $p_X = p_Y$, we have $p_{\hat{X}^*|X} = p_{\hat{Y}^*|Y}$ and consequently the upper bound and the lower bound in (5.3.8) coincide.

Note that due to the involvement of conditional expectation, $\hat{X}$ is not necessarily defined over $\mathcal{X}$ if $\mathcal{X}$ is a strict subset of $\mathbb{R}^n$ (for the same reason, $\hat{Y}$ is not necessarily defined over $\mathcal{Y}$). In other words, the output of the quantizer is not consrained to the input alphabet and needs to be relaxed to $\mathbb{R}^n$. As such, $D_{\text{mse}}(p_X, R)$ should be interpreted as the one-shot distortion-rate function with the reconstruction alphabet being $\mathbb{R}^n$, which is in general strictly below its counterpart with the reconstruction alphabet being $\mathcal{X}$ (also known as the distortion-rate-perception function with an inactive perception constraint) if $\mathcal{X}$ is a strictly subset of $\mathbb{R}^n$. This subtle issue, which is often overlooked in the literature, arises when one deals with discrete X and Y (see the binary example in Section 5.3.2 and Appendix A.1.4). □

Proof of Theorem 2. We have that $\max\{H(\hat{X}), H(\hat{Y})\} \leq R^*$. If one of them, say $H(Y)$, is less than R^*, this will lead to a contradiction by the following argument. Note that

$$D^* = \mathbb{E}[\|X - Y\|^2] = \mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2],$$

which depends on $p_{X,\hat{X},\hat{Y},Y}$ only through $p_{X,\hat{X}}$, $p_{\hat{X},\hat{Y}}$, and $p_{Y,\hat{Y}}$. We can construct a new joint distribution $p_{X,\hat{X}',\hat{Y}',Y}$ such that $p_{X,\hat{X}'} = p_{X,\hat{X}}$, $p_{\hat{X}',\hat{Y}'} = p_{\hat{X},\hat{Y}}$, $p_{Y,\hat{Y}'} = p_{Y,\hat{Y}}$, and $X \leftrightarrow \hat{X}' \leftrightarrow \hat{Y}' \leftrightarrow Y$ form a Markov chain. Denote $\hat{X}'$ by Z'. It is clear that the induced $p_{X,Z',Y}$ belongs to $M_{\text{ncr}}(p_X, p_Y)$, preserves $\mathbb{E}[\|X - Y\|^2]$, and $H(Z') < R^*$, which is contradictory with the fact that $D_{\text{ncr}}(p_X, p_{\hat{X}}, R)$ is a strictly decreasing function in a neighborhood of $R = R^*$ since R can be set slightly below R^* without violating the constraint $H(Z') \leq R$. This proves (5.3.9), which futher implies the existence of a bijection between $\hat{X}$ and $\hat{Y}$.

It remains to prove (5.3.10). If (5.3.10) does not hold, then we can find some $p_{\hat{X}'',\hat{Y}''}$ with $p_{\hat{X}''} = p_{\hat{X}}$ and $p_{\hat{Y}''} = p_{\hat{Y}}$ such that $\mathbb{E}[\|\hat{X}'' - \hat{Y}''\|^2] < \mathbb{E}[\|\hat{X} - \hat{Y}\|^2]$. Leverage this $p_{\hat{X}'',\hat{Y}''}$ to construct a new joint distribution $p_{X,\hat{X}'',\hat{Y}'',Y}$ such that $p_{X,\hat{X}''} = p_{X,\hat{X}}$, $p_{Y,\hat{Y}''} = p_{Y,\hat{Y}}$, and $X \leftrightarrow \hat{X}'' \leftrightarrow \hat{Y}'' \leftrightarrow Y$ form a Markov chain. Denote $\hat{X}''$ by Z''. It is clear that the induced $p_{X,Z'',Y}$ belongs to $M_{\mathrm{ncr}}(p_X, p_Y)$, $H(Z'') = R^*$, and

$$\begin{aligned}\mathbb{E}[\|X-Y\|^2] &= \mathbb{E}[\|X-\hat{X}''\|^2] + \mathbb{E}[\|Y-\hat{Y}''\|^2] + \mathbb{E}[\|\hat{X}''-\hat{Y}''\|^2]\\ &< \mathbb{E}[\|X-\hat{X}\|^2] + \mathbb{E}[\|Y-\hat{Y}\|^2] + \mathbb{E}[\|\hat{X}-\hat{Y}\|^2]\\ &= D^*,\end{aligned}$$

which is contradictory with the fact that $R_{\mathrm{ncr}}(p_X, p_Y, D)$ is a strictly decreasing function in a neighborhood of $D = D^* \triangleq D_{\mathrm{ncr}}(p_X, p_{\hat{X}}, R^*)$ since D can be set slightly below D^* without violating the constraint $\mathbb{E}[\|X-Y\|^2] \leq D$. So in conclusion, the converter is a one-to-one mapping, which induces an optimal coupling that attains $W_2^2(p_{\hat{X}}, p_{\hat{Y}})$. □

Proof of Theorem 3. Choosing $p_{U,X,Y}$ from $W_2^2(p_X, p_Y)$ and setting $Z = Y$ shows that

$$\begin{aligned}D_{\mathrm{cr}}(p_X, p_Y, R) \leq &\inf_{p_{U,X,Y} \in W(p_X,p_Y)} \mathbb{E}[d(X,Y)]\\ &\text{s.t.} \quad H(Y|U) \leq R.\end{aligned}$$

So it remains to prove that this upper bound is tight. In the light of the functional representation lemma, for any (U, X, Z, Y) with $p_{U,X,Z,Y} \in M_{\mathrm{cr}}(p_X, p_Y)$, there exist V_1, independent of (U, X), and V_2, independent of (U, X, V_1), as well as determintic mappings ϕ_1 and ϕ_2 such that $Z = \phi_2(U, X, V_1)$ and $Y = \phi_2(Z, V_2)$. Let $U' \triangleq$

(U, V_1, V_2). Clearly, $p_{U',X,Y} = p_{U'}p_Xp_{Y|U',X}$ and $H(Y|U',X) = 0$. Moreover, we have

$$\begin{aligned} H(Z|U) &\geq H(Z|U') \\ &\geq H(Y|U'), \end{aligned}$$

where the last inequality is due to the fact that Y is determined by (Z, U'). Therefore,

$$\begin{aligned} D_{\mathrm{cr}}(p_X, p_Y, R) \geq &\inf_{p_{U',X,Y} \in W(p_X, p_Y)} \mathbb{E}[d(X,Y)] \\ &\text{s.t.} \quad H(Y|U') \leq R. \end{aligned}$$

This completes the proof of (5.3.12).

Note that each realization of U is associated with a deterministic function from $\mathcal{X}$ to $\mathcal{Y}$. As a consequence, the problem boils down to optimizing the probablity distribution defined over this collection of functions. For the finite alphabet case, there are totally $|\mathcal{Y}|^{|\mathcal{X}|}$ such functions. In fact, a simple application of the support lemma shows that only $|\mathcal{Y}| + 1$ functions need to be assigned with a positive probability. □

A.1.3 Linear Program Formulation for Common Randomness

In the finite alphabet case, we can formulate Theorem 3 as follows:

$$\begin{aligned} D_{\mathrm{cr}}(p_X, p_Y, R) = \min_{p_U} &\sum_{u \in \mathcal{U}} p_U(u)\mathbb{E}[d(X, f_u(X))] \\ \text{s.t.} \quad &\sum_{u \in \mathcal{U}} p_U(u) H(f_u(X)) \leq R, \\ &\sum_{u \in \mathcal{U}} p_U(u)\mathbb{P}(f_u(X) = y) = p_Y(y), \quad y \in \mathcal{Y}, \end{aligned}$$

where p_U is defined over $\mathcal{U} \triangleq \{1, 2, \cdots, |\mathcal{Y}|^{|\mathcal{X}|}\}$, and $\{f_u : u \in \mathcal{U}\}$ is the set of all distinct functions from $\mathcal{X}$ to $\mathcal{Y}$. By the support lemma (Appendix C on page 631 of [32]), only $|\mathcal{Y}| + 1$ functions need to be assigned with a positive probability.

A.1.4 Binary Case

Let $D^{(B)}_{\min} \triangleq |q_X - q_Y|$ and $D^{(B)}_{\max} \triangleq q_X + q_Y - 2q_X q_Y$. Note that $D^{(B)}_{\min}$ is the total variation distance between $\mathcal{B}(q_X)$ and $\mathcal{B}(q_Y)$, which is the minimum $\mathbb{E}[d_H(X, Y)]$ achievable by coupling X and Y. On the other hand, we have $\mathbb{E}[d_H(X, Y)] = D^{(B)}_{\max}$ for X, Y independent. It is clear that $D^{(B)}_{\min}$ and $D^{(B)}_{\max}$ are the infimum and supremum of $D_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$ (as well as $D_{\text{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$), respectively.

Theorem 4. *Assume Hamming distortion measure. Under no common randomness, we have*

$$D_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R) = \begin{cases} -\frac{2(1-q_X)(1-q_Y)}{1-H_b^{-1}(R)} + 2 - q_X - q_Y, & q_X + q_Y \leq 1, \\ -\frac{2q_X q_Y}{1-H_b^{-1}(R)} + q_X + q_Y, & q_X + q_Y > 1, \end{cases} \tag{A.1.4}$$

for $R \in [0, \min\{H_b(q_X), H_b(q_Y)\}]$*, and* $= D_{\min}^{(B)}$ *for* $R > \min\{H_b(q_X), H_b(q_Y)\}$*. Moreover,*

$$\begin{aligned}&\overline{D}_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)\\ &= \begin{cases} -\frac{2(1-q_X)(1-q_Y)}{1-H_b^{-1}(R)} + 2 - q_X - q_Y, & q_X, q_Y \leq \frac{1}{2},\\ -\frac{2q_X q_Y}{1-H_b^{-1}(R)} + q_X + q_Y, & q_X, q_Y \geq \frac{1}{2},\\ -\frac{(1-q_X)^2+q_Y^2-(q_Y-q_X+H_b^{-1}(R))^2}{1-H_b^{-1}(R)} + H_b^{-1}(R) - \frac{2q_Y(1-q_X)H_b^{-1}(R)}{(1-H_b^{-1}(R))^2}, & q_X < \frac{1}{2}, q_Y > \frac{1}{2},\\ -\frac{q_X^2+(1-q_Y)^2-(q_X-q_Y+H_b^{-1}(R))^2}{1-H_b^{-1}(R)} + H_b^{-1}(R) - \frac{2q_X(1-q_Y)H_b^{-1}(R)}{(1-H_b^{-1}(R))^2}, & q_X > \frac{1}{2}, q_Y < \frac{1}{2}, \end{cases}\end{aligned} \tag{A.1.5}$$

for $R \in [0, \min\{H_b(q_X), H_b(q_Y)\}]$*, and*

$$\underline{D}_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R) = \begin{cases} -\frac{(1-q_X)^2+(1-q_Y)^2}{1-H_b^{-1}(R)} + 2 - q_X - q_Y, & q_X, q_Y \leq \frac{1}{2},\\ -\frac{q_X^2+q_Y^2}{1-H_b^{-1}(R)} + q_X + q_Y, & q_X, q_Y \geq \frac{1}{2},\\ -\frac{(1-q_X)^2+q_Y^2}{1-H_b^{-1}(R)} + 1 - q_X + q_Y, & q_X < \frac{1}{2}, q_Y > \frac{1}{2},\\ -\frac{q_X^2+(1-q_Y)^2}{1-H_b^{-1}(R)} + 1 + q_X - q_Y, & q_X > \frac{1}{2}, q_Y < \frac{1}{2}, \end{cases} \tag{A.1.6}$$

for $R \in [0, \min\{H_b(q_X), H_b(q_Y)\}]$*. Here,* $H_b^{-1}(R)$ *denotes the inverse of the binary entropy function on* $[0, 1/2]$*. With common randomness,*

$$D_{\text{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R) = -\frac{2(1-q_X)q_X R}{H_b(q_X)} + D_{\max}^{(B)} \tag{A.1.7}$$

for $R \in [0, \rho H_b(q_X)]$*, and* $= D_{\min}^{(B)}$ *for* $R > \rho H_b(q_X)$*. Here,* $\rho \triangleq \min\{q_Y/q_X, (1 -$

$q_Y)/(1-q_X)\}$.

Proof of (A.1.7). There are totally 4 distinct functions from $\{0,1\}$ to $\{0,1\}$:

$$f_1(x)=x,\quad f_2(x)=1-x,\quad f_3(x)=0,\quad f_4(x)=1,\quad x\in\{0,1\}.$$

Therefore, we have

$$\begin{aligned}
&\sum_{u\in\mathcal{U}} p_U(u)H(f_u(X))=H_b(q_X)(p_U(1)+p_U(2)),\\
&\sum_{u\in\mathcal{U}} p_U(u)\mathbb{E}[d_H(X,f_u(X))]=p_U(2)+q_Xp_U(3)+(1-q_X)p_U(4),\\
&\sum_{u\in\mathcal{U}} p_U(u)\mathbb{P}(f_u(X)=1)=p_U(1)q_X+(1-q_X)p_U(2)+p_U(4).
\end{aligned}$$

In light of Theorem 3,

$$\begin{aligned}
R_{\mathrm{cr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),D)=\min_{p_U(1),\cdots,p_U(4)}\ & H_b(q_X)(p_U(1)+p_U(2))\\
\text{s.t.}\quad & p_U(2)+q_Xp_U(3)+(1-q_X)p_U(4)\le D, && \text{(A.1.8)}\\
& q_Xp_U(1)+(1-q_X)p_U(2)+p_U(4)=q_Y, && \text{(A.1.9)}\\
& p_U(1)+p_U(2)+p_U(3)+p_U(4)=1, && \text{(A.1.10)}\\
& p_U(1),p_U(2),p_U(3),p_U(4)\ge 0. && \text{(A.1.11)}
\end{aligned}$$

Note that

$$
\begin{aligned}
&p_U(2)+q_Xp_U(3)+(1-q_X)p_U(4)\\
&=p_U(2)+q_Xp_U(3)+(1-q_X)(1-p_U(1)-p_U(2)-p_U(3)) && \text{(A.1.12)}\\
&=-(1-q_X)p_U(1)+q_Xp_U(2)+(2q_X-1)p_U(3)+1-q_X, && \text{(A.1.13)}
\end{aligned}
$$

where (A.1.12) is due to (A.1.10). Moreover, it follows by (A.1.9) and (A.1.10) that

$$p_U(3)=-(1-q_X)p_U(1)-q_Xp_U(2)+1-q_Y. \tag{A.1.14}$$

Substituting (A.1.14) into (A.1.13) and invoking the fact that $p_U(2)\geq 0$ gives

$$
\begin{aligned}
&p_U(2)+q_Xp_U(3)+(1-q_X)p_U(4)\\
&=-2(1-q_X)q_X(p_U(1)+p_U(2))+4(1-q_X)q_Xp_U(2)+D_{\max}^{(B)}\\
&\geq-2(1-q_X)q_X(p_U(1)+p_U(2))+D_{\max}^{(B)},
\end{aligned}
$$

which, together with (A.1.8), implies

$$p_U(1)+p_U(2)\geq\frac{1}{2(1-q_X)q_X}(D_{\max}^{(B)}-D).$$

As a consequence, we must have

$$R_{\mathrm{cr}}(\mathcal{B}(q_X),\mathcal{B}(q_Y),D)\geq\frac{H_b(q_X)}{2(1-q_X)q_X}(D_{\max}^{(B)}-D).$$

One can readily verify that this lower bound is tight as it is attained by p_U^* with

$$\begin{aligned}
p_U^*(1) &= \frac{1}{2(1-q_X)q_X}(D_{\max}^{(B)} - D),\\
p_U^*(2) &= 0,\\
p_U^*(3) &= -\frac{1}{2q_X}(D_{\max}^{(B)} - D) + 1 - q_Y,\\
p_U^*(4) &= -\frac{1}{2(1-q_X)}(D_{\max}^{(B)} - D) + q_Y,
\end{aligned}$$

which satisfies (A.1.8)–(A.1.11) for $D \in [D_{\min}^{(B)}, D_{\max}^{(B)}]$. The expression of $D_{\text{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$ can be obtained by taking the inverse of $R_{\text{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), D)$. □

Proof of (A.1.4). We will rely on some results which will come after this proof.

Note that Hamming distortion coincides with squared error distortion when $\mathcal{X} = \mathcal{Y} = \{0, 1\}$. So Theorem 1, Lemma 1, and Lemma 2 are applicable here. In particular, in light of Lemmas 1 and 2, for any $R \geq 0$ and $\epsilon > 0$, there exists a joint distribution $p_{X\hat{X}\hat{Y}Y}$ compatible with the given marginal distributions p_X and p_Y such that $\hat{X}$ and $\hat{Y}$ are deterministically related finite-support random variables with $H(\hat{X}) = H(\hat{Y}) \leq R$ and $X \leftrightarrow \hat{X} \leftrightarrow \hat{Y} \leftrightarrow Y$ form a Markov chain; moreover, $\hat{X} = \mathbb{E}[X|\hat{X}]$, $\hat{Y} = \mathbb{E}[Y|\hat{Y}]$, and

$$\mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2] \leq D_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R) + \epsilon. \quad \text{(A.1.15)}$$

Without loss of generality, we assume $\hat{X}$ and $\hat{Y}$ take value from $\{\hat{x}_i\}_{i=1}^N$ and $\{\hat{y}_i\}_{i=1}^N$, respectively, and $\hat{Y} = \psi(\hat{X})$, where ψ is a bijection from $\{\hat{x}_i\}_{i=1}^N$ to $\{\hat{y}_i\}_{i=1}^N$ with $\hat{y}_i = \psi(\hat{x}_i)$, $i = 1, \cdots, N$. Let $\theta_i \triangleq p_{\hat{X}}(\hat{x}_i)$, or equivalently, $\theta_i \triangleq p_{\hat{Y}}(\hat{y}_i)$, $i =$

$1, \cdots, N$. Note that $\hat{X} = \mathbb{E}[X|\hat{X}]$ and $\hat{Y} = \mathbb{E}[Y|\hat{Y}]$ if and only if $p_{X|\hat{X}}(1|\hat{x}_i) = \hat{x}_i$ and $p_{Y|\hat{Y}}(1|\hat{y}_i) = \hat{y}_i$ for $\theta_i > 0$, $i = 1, \cdots, N$. So the constraints $\sum_{i=1}^{N} p_{\hat{X}}(\hat{x}_i) p_{X|\hat{X}}(1|\hat{x}_i) = q_X$ and $\sum_{i=1}^{N} p_{\hat{Y}}(\hat{y}_i) p_{Y|\hat{Y}}(1|\hat{y}_i) = q_Y$ can be written equivalently as $\sum_{i=1}^{N} \theta_i \hat{x}_i = q_X$ and $\sum_{i=1}^{N} \theta_i \hat{y}_i = q_Y$. Moreover, it is easy to verify that

$$\begin{aligned}
&\mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2] \\
&= \sum_{i=1}^{N} \theta_i (1 - \hat{x}_i)\hat{x}_i + \sum_{i=1}^{N} \theta_i (1 - \hat{y}_i)\hat{y}_i + \sum_{i=1}^{n} \theta_i (\hat{x}_i - \hat{y}_i)^2 \\
&= \sum_{i=1}^{N} \theta_i (\hat{x}_i + \hat{y}_i - 2\hat{x}_i\hat{y}_i).
\end{aligned}$$

In light of Theorem 1 and (A.1.15), the following optimization problem (P) yields an upper bound on $D_{\mathrm{ncr}}(p_X, p_Y, R)$ with a gap at most ϵ:

$$\begin{aligned}
&\min_{(\theta_i, \hat{x}_i, \hat{y}_i,)_{i=1}^{N}} \sum_{i=1}^{N} \theta_i (\hat{x}_i + \hat{y}_i - 2\hat{x}_i\hat{y}_i) \qquad \text{(P)} \\
&\text{s.t.} \quad \sum_{i=1}^{N} \theta_i \log \frac{1}{\theta_i} \leq R, \quad \sum_{i=1}^{N} \theta_i = 1, \quad \sum_{i=1}^{N} \theta_i \hat{x}_i = q_X, \quad \sum_{i=1}^{N} \theta_i \hat{y}_i = q_Y, \\
&\qquad \theta_i \geq 0, \quad \hat{x}_i \in [0, 1], \quad \hat{y}_i \in [0, 1], \quad i = 1, \cdots, N.
\end{aligned}$$

Given $(\theta_i, \hat{y}_i)_i^N$, (P) degenerates to a linear programming problem with respect $(\hat{x}_i)_{i=1}^N$ over hyperrectangle $[0, 1]^N$ subject to the constraint $\sum_{i=1}^{N} \theta_i \hat{x}_i = q_X$, for which the minimum is attained at a point on an edge of $[0, 1]^N$. Therefore, it suffices to consider $(\hat{x}_i)_{i=1}^N$ with at most one element different from 0 and 1. By a similar argument, it can be shown that there is no loss of optimality in assuming that at most one of $\hat{y}_i$, $i = 1, \cdots, N$, takes value other than 0 or 1. Due to the merge of different elements, the one-to-one relationship might not be preserved. Nevertheless, by Lemma

2, we just need to consider deterministically related $\hat{X}$ and $\hat{Y}$ with support size at most 3. Applying the linear programming argument to (P) with $N = 3$ shows that, at the cost of potentially compromising the one-to-one relationship, at most one element in the support of $\hat{X}$ as well as the support of $\hat{Y}$ need to take value different from 0 and 1. In the case that the bijection is lost, $\hat{X}$ or $\hat{Y}$ must have a reduced support size. One can restore the bijection by invoking Lemma 2, then use the linear programming argument to assign extreme values to all but at most one element in the support. Following this line of reasoning, we can conclude that the attention can be restricted to deterministically related $\hat{X}$ and $\hat{Y}$ with support size at most 3 and at most one element in the support different from 0 and 1. Moreover, the following configurations can be excluded.

1. Support size = 3 and the existence of pairs $(\hat{x}, \hat{y})$ and $(\hat{x}', \hat{y}')$ for some $\hat{x} > \hat{x}'$ and $\hat{y} < \hat{y}'$ ($(\hat{x}, \hat{y})$ is said to be a pair if $\hat{X} = \hat{x} \Leftrightarrow \hat{Y} = \hat{y}$): Since

$$\begin{aligned}
&(\hat{x} - \hat{y})^2 + (\hat{x}' - \hat{y}')^2 - (\hat{x} - \hat{y}')^2 - (\hat{x}' - \hat{y})^2 \\
&= -2\hat{x}\hat{y} - 2\hat{x}'\hat{y}' + 2\hat{x}\hat{y}' + 2\hat{x}'\hat{y} \\
&= 2(\hat{x} - \hat{x}')(\hat{y}' - \hat{y}) \\
&> 0,
\end{aligned}$$

   it follows that $\mathbb{E}[\|\hat{X} - \hat{Y}\|^2]$ can be strictly reduced by moving the same amount of probability from $\{\hat{X} = \hat{x}, \hat{Y} = \hat{y}\}$ to $\{\hat{X} = \hat{x}, \hat{Y} = \hat{y}'\}$ and from $\{\hat{X} = \hat{x}', \hat{Y} = \hat{y}'\}$ to $\{\hat{X} = \hat{x}', \hat{Y} = \hat{y}\}$. This modification does not affect $p_{\hat{X}}$ and $p_{\hat{Y}}$, and consequently $H(\hat{X})$, $H(\hat{Y})$, $\mathbb{E}[\|X - \hat{X}\|^2]$, $\mathbb{E}[\|Y - \hat{Y}\|^2]$ remain the same. So the distortion-rate performance of this configuration is strictly suboptimal.

2. Support size $= 2$ and the existence of pairs $(\hat{x}, \hat{y})$ and $(\hat{x}', \hat{y}')$ for some $\hat{x} > \hat{x}'$ and $\hat{y} < \hat{y}'$: Same as configuration 1).

3. Support size $= 2$ and existence of pairs $(\hat{x}, 1)$ and $(0, \hat{y})$ for some $\hat{x} \in (0, 1)$ and $\hat{y} \in (0, 1)$: It follows by $\mathbb{E}[X|\hat{X}] = \hat{X}$ and $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$ that

$$\begin{aligned} p_{\hat{X},\hat{Y}}(\hat{x}, 1) &= \frac{q_X}{\hat{x}}, \\ p_{\hat{X},\hat{Y}}(0, \hat{y}) &= 1 - \frac{q_X}{\hat{x}}, \\ \hat{y} &= 1 - \frac{1 - q_Y}{1 - \frac{q_X}{\hat{x}}}. \end{aligned}$$

Clearly, $H(\hat{X}) = H(\hat{Y}) = H_b(\frac{q_X}{\hat{x}})$. Since $\hat{x} \in (0, 1)$ and $\hat{y} \in (0, 1)$, we must have $q_X < \frac{q_X}{\hat{x}} < q_Y$, which implies $H(\hat{X}) = H(\hat{Y}) > \min\{H(X), H(Y)\}$. Furthermore, it can be verified that

$$\begin{aligned} &\mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2] \\ &= \frac{q_X}{\hat{x}}\hat{x}(1 - \hat{x}) + \left(1 - \frac{q_X}{\hat{x}}\right)\hat{y}(1 - \hat{y}) + \frac{q_X}{\hat{x}}(1 - \hat{x})^2 + \left(1 - \frac{q_X}{\hat{x}}\right)\hat{y}^2 \\ &= q_Y - q_X. \end{aligned}$$

However, this end-to-end distortion is obviously achievable when $R = \min\{H(X), H(Y)\}$. So the rate-distortion performance of this configuration is strictly suboptimal.

4. Support size $= 2$ and the existence of pairs $(\hat{x}, 0)$ and $(1, \hat{y})$ for some $\hat{x} \in (0, 1)$ and $\hat{y} \in (0, 1)$: Same as configuration 3).

In view of the excluded configurations, we are left with the case where $p_{\hat{X}\hat{Y}}$ assigns all probabilities to $\{\hat{X} = 0, \hat{Y} = 0\}$, $\{\hat{X} = 1, \hat{Y} = 1\}$, and $\{\hat{X} = \hat{x}, \hat{Y} = \hat{y}\}$ for some $\hat{x} \in [0,1]$ and $\hat{y} \in [0,1]$. So it suffices to consider the $N = 3$ version of (P) with $\hat{x}_1 = \hat{y}_1 = 0$, $\hat{x}_3 = \hat{y}_3 = 1$, $\hat{x}_2 = \hat{x}$, and $\hat{y}_2 = \hat{y}$. The constraints $\sum_{i=1}^{3} \theta_i = 1$, $\sum_{i=1}^{3} \theta_i \hat{x}_i = q_X$, and $\sum_{i=1}^{3} \theta_i \hat{y}_i = q_Y$ imply

$$\begin{aligned}
&\theta_1 = 1 - q_X - (1-\hat{x})\theta,\\
&\theta_3 = q_X - \hat{x}\theta,\\
&\hat{y} = \frac{q_Y - q_X}{\theta} + \hat{x}.
\end{aligned}$$

In this way, we get a simplified optimization problem (P'):

$$\begin{aligned}
&\min_{\theta,\hat{x}} \; 2\hat{x}(1-\hat{x})\theta + (1-2\hat{x})(q_Y - q_X) \qquad \text{(P')}\\
&\text{s.t.} \quad (1 - q_X - (1-\hat{x})\theta)\log\frac{1}{1 - q_X - (1-\hat{x})\theta} + \theta\log\frac{1}{\theta} + (q_X - \hat{x}\theta)\log\frac{1}{q_X - \hat{x}\theta} \le R,\\
&\qquad \hat{x} \in [0,1], \quad \theta \in [0,1], \quad (1-\hat{x})\theta \in [q_Y - q_X, 1 - q_X], \quad \hat{x}\theta \in [q_X - q_Y, q_X].
\end{aligned}$$

Note that (P') does not depend on ϵ and consequently yields the exact characterization of $D_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)$. Therefore, $R_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), D)$ is characterized by the following optimization problem (P"):

$$\begin{aligned}
&\min_{\theta,\hat{x}} (1 - q_X - (1-\hat{x})\theta)\log\frac{1}{1 - q_X - (1-\hat{x})\theta} + \theta\log\frac{1}{\theta} + (q_X - \hat{x}\theta)\log\frac{1}{q_X - \hat{x}\theta} \qquad \text{(P")}\\
&\text{s.t.} \quad 2\hat{x}(1-\hat{x})\theta + (1-2\hat{x})(q_Y - q_X) \le D,\\
&\qquad \hat{x} \in [0,1], \quad \theta \in [0,1], \quad (1-\hat{x})\theta \in [q_Y - q_X, 1 - q_X], \quad \hat{x}\theta \in [q_X - q_Y, q_X].
\end{aligned}$$

Given $\hat{x}$, the objective function of (P") is concave in θ and consequently its minimum is attained at an endpoint of $[\underline{\theta}, \overline{\theta}]$, where

$$\begin{aligned}\underline{\theta} &\triangleq \max\left\{0, \frac{q_Y - q_X}{1-\hat{x}}, \frac{q_X - q_Y}{\hat{x}}\right\},\\ \overline{\theta} &\triangleq \min\left\{1, \frac{1-q_X}{1-\hat{x}}, \frac{q_X}{\hat{x}}, \frac{D-(1-2\hat{x})(q_Y-q_X)}{2\hat{x}(1-\hat{x})}\right\}.\end{aligned}$$

Without loss of generality, we assume $q_Y \geq q_X$ and $q_X + q_Y \leq 1$. The following statements can be easily verified.

1. For $\hat{x} \in [0, \frac{D+q_X-q_Y}{2(1-q_Y)}]$,

$$\begin{aligned}\underline{\theta} &= \frac{q_Y - q_X}{1-\hat{x}},\\ \overline{\theta} &= \frac{1-q_X}{1-\hat{x}}.\end{aligned}$$

2. For $\hat{x} \in (\frac{D+q_X-q_Y}{2(1-q_Y)}, \frac{q_X+q_Y-D}{2q_Y}]$,

$$\begin{aligned}\underline{\theta} &= \frac{q_Y - q_X}{1-\hat{x}},\\ \overline{\theta} &= \frac{D-(1-2\hat{x})(q_Y-q_X)}{2\hat{x}(1-\hat{x})}.\end{aligned}$$

3. For $\hat{x} \in (\frac{q_X+q_Y-D}{2q_Y}, \frac{q_X}{q_Y}]$,

$$\begin{aligned}\underline{\theta} &= \frac{q_Y - q_X}{1-\hat{x}},\\ \overline{\theta} &= \frac{q_X}{\hat{x}}.\end{aligned}$$

4. For $\hat{x} > \frac{q_X}{q_Y}$, $[\underline{\theta}, \overline{\theta}]$ is empty.

Note that when $\hat{x} \in [0, \frac{q_X}{q_Y}]$ and $\theta = \underline{\theta}$,

$$\begin{aligned}
&(1 - q_X - (1 - \hat{x})\theta) \log \frac{1}{1 - q_X - (1 - \hat{x})\theta} + \theta \log \frac{1}{\theta} + (q_X - \hat{x}\theta) \log \frac{1}{q_X - \hat{x}\theta} \\
&= (1 - q_Y) \log \frac{1}{1 - q_Y} + \underline{\theta} \log \frac{1}{\underline{\theta}} + (q_X - \hat{x}\underline{\theta}) \log \frac{1}{q_X - \hat{x}\underline{\theta}} \\
&\geq H_b(q_Y) \\
&= H(Y).
\end{aligned}$$

So it suffices to consider the case $\hat{x} \in [0, \frac{q_X}{q_Y}]$ and $\theta = \overline{\theta}$, for which (P") is reduced to the following form:

$$\min_{\hat{x} \in [0, \frac{q_X}{q_Y}]} \eta(\hat{x}),$$

where

$$\eta(\hat{x}) \triangleq \begin{cases}
\frac{1-q_X}{1-\hat{x}} \log \frac{1-\hat{x}}{1-q_X} + \frac{q_X-\hat{x}}{1-\hat{x}} \log \frac{1-\hat{x}}{q_X-\hat{x}}, & \hat{x} \in [0, \frac{D+q_X-q_Y}{2(1-q_Y)}], \\
\frac{2\hat{x}-q_X+(1-2\hat{x})q_Y-D}{2\hat{x}} \log \frac{2\hat{x}}{2\hat{x}-q_X+(1-2\hat{x})q_Y-D} & \\
+\frac{D-(1-2\hat{x})(q_Y-q_X)}{2\hat{x}(1-\hat{x})} \log \frac{2\hat{x}(1-\hat{x})}{D-(1-2\hat{x})(q_Y-q_X)} & \\
+\frac{q_X+(1-2\hat{x})q_Y-D}{2(1-\hat{x})} \log \frac{2(1-\hat{x})}{q_X+(1-2\hat{x})q_Y-D}, & \hat{x} \in (\frac{D+q_X-q_Y}{2(1-q_Y)}, \frac{q_X+q_Y-D}{2q_Y}], \\
\frac{\hat{x}-q_X}{\hat{x}} \log \frac{\hat{x}}{\hat{x}-q_X} + \frac{q_X}{\hat{x}} \log \frac{\hat{x}}{q_X}, & \hat{x} \in (\frac{q_X+q_Y-D}{2q_Y}, \frac{q_X}{q_Y}].
\end{cases}$$

Note that $\eta(\hat{x})$ is a continuous function over $[0, \frac{q_X}{q_Y}]$. Moreover, $\eta(\hat{x})$ decreases monotonically from $H_b(q_X)$ to $H_b(\frac{D_{\max}^{(B)}-D}{2-q_X-q_Y-D})$ as $\hat{x}$ varies from 0 to $\frac{D+q_X-q_Y}{2(1-q_Y)}$. Since $\eta(\hat{x})$ is a concave function of $\frac{q_X}{\hat{x}}$ for $\hat{x} \in [\frac{q_X+q_Y-D}{2q_Y}, \frac{q_X}{q_Y}]$, it follows that

$$\min_{\hat{x}\in[\frac{q_X+q_Y-D}{2q_Y},\frac{q_X}{q_Y}]} \eta(\hat{x}) = \min\left\{\eta(\frac{q_X+q_Y-D}{2q_Y}), \eta(\frac{q_X}{q_Y})\right\} = \min\left\{H_b(\frac{2q_Xq_Y}{q_X+q_Y-D}), H_b(q_Y)\right\}.$$

So we have

$$\begin{aligned}
&\min_{\hat{x}\in[0,\frac{D+q_X-q_Y}{2(1-q_Y)}]\cup[\frac{q_X+q_Y-D}{2q_Y},\frac{q_X}{q_Y}]} \eta(\hat{x}) \\
&= \min\left\{H_b(\frac{D_{\max}^{(B)}-D}{2-q_X-q_Y-D}), H_b(\frac{2q_Xq_Y}{q_X+q_Y-D}), H_b(q_Y)\right\} \\
&= \min\left\{H_b(\frac{D_{\max}^{(B)}-D}{2-q_X-q_Y-D}), H_b(\frac{2q_Xq_Y}{q_X+q_Y-D})\right\} \qquad \text{(A.1.16)} \\
&= \min\left\{\eta(\frac{D+q_X-q_Y}{2(1-q_Y)}), \eta(\frac{q_X+q_Y-D}{2q_Y})\right\} \\
&\geq \min_{\hat{x}\in[\frac{D+q_X-q_Y}{2(1-q_Y)},\frac{q_X+q_Y-D}{2q_Y}]} \eta(\hat{x}),
\end{aligned}$$

where (A.1.16) is due to the fact that $H_b(q_Y) \geq H_b(q_X) \geq H_b(\frac{D_{\max}^{(B)}-D}{2-q_X-q_Y-D})$ (with the first inequality being a consequence of $q_X \leq q_Y$ and $q_X + q_Y \leq 1$). So the problem boils down to solving

$$\min_{\hat{x}\in[\frac{D+q_X-q_Y}{2(1-q_Y)},\frac{q_X+q_Y-D}{2q_Y}]} \eta(\hat{x}).$$

It can be verified that the minimum is attained at $\hat{x} = \frac{D+q_X-q_Y}{2(1-q_Y)}$. This completes the proof of (A.1.4). A graphical illustration of the entropy-constrained optimal transport plan for the binary case can be found in Figure A.1.

Proof of (A.1.5). Note that

$$\begin{aligned}
D_{\mathrm{mse}}(\mathcal{B}(q_X),R) &= \frac{1}{2}D_{\mathrm{ncr}}(\mathcal{B}(q_X),\mathcal{B}(q_X),R)\\
&= \begin{cases} -\frac{(1-q_X)^2}{1-H_b^{-1}(R)}+1-q_X, & q_X\le\frac{1}{2},\\ -\frac{q_X^2}{1-H_b^{-1}(R)}+q_X, & q_X>\frac{1}{2}, \end{cases} \quad R\in[0,H_b(q_X)],\\
D_{\mathrm{mse}}(\mathcal{B}(q_Y),R) &= \frac{1}{2}D_{\mathrm{ncr}}(\mathcal{B}(q_Y),\mathcal{B}(q_Y),R)\\
&= \begin{cases} -\frac{(1-q_Y)^2}{1-H_b^{-1}(R)}+1-q_Y, & q_Y\le\frac{1}{2},\\ -\frac{q_Y^2}{1-H_b^{-1}(R)}+q_Y, & q_Y>\frac{1}{2}, \end{cases} \quad R\in[0,H_b(q_Y)].
\end{aligned}$$

Moreover, we have

$$\begin{cases} p_{\hat{X}^*}(\frac{q_X-H_b^{-1}(R)}{1-H_b^{-1}(R)})=1-H_b^{-1}(R),\\ p_{\hat{X}^*}(1)=H_b^{-1}(R), \end{cases} \quad R\in[0,H_b(q_X)],q_X\le\frac{1}{2},$$

$$\begin{cases} p_{\hat{X}^*}(0)=H_b^{-1}(R),\\ p_{\hat{X}^*}(\frac{q_X}{1-H_b^{-1}(R)})=1-H_b^{-1}(R), \end{cases} \quad R\in[0,H_b(q_X)],q_X\ge\frac{1}{2},$$

$$\begin{cases} p_{\hat{Y}^*}(\frac{q_Y-H_b^{-1}(R)}{1-H_b^{-1}(R)})=1-H_b^{-1}(R),\\ p_{\hat{Y}^*}(1)=H_b^{-1}(R), \end{cases} \quad R\in[0,H_b(q_Y)],q_Y\le\frac{1}{2},$$

$$\begin{cases} p_{\hat{Y}^*}(0)=H_b^{-1}(R),\\ p_{\hat{Y}^*}(\frac{q_Y}{1-H_b^{-1}(R)})=1-H_b^{-1}(R), \end{cases} \quad R\in[0,H_b(q_Y)],q_Y\ge\frac{1}{2}.$$

So

$$
\begin{aligned}
&W_2^2(p_{\hat{X}^*}, p_{\hat{Y}^*})\\
&=\begin{cases}
\frac{(q_X-q_Y)^2}{1-H_b^{-1}(R)}, & q_X, q_Y \leq \frac{1}{2} \text{ or } q_X, q_Y \geq \frac{1}{2},\\
\frac{(q_Y-q_X+H_b^{-1}(R))^2}{1-H_b^{-1}(R)} + H_b^{-1}(R) - \frac{2q_Y(1-q_X)H_b^{-1}(R)}{(1-H_b^{-1}(R))^2}, & q_X < \frac{1}{2}, q_Y > \frac{1}{2},\\
\frac{(q_X-q_Y+H_b^{-1}(R))^2}{1-H_b^{-1}(R)} + H_b^{-1}(R) - \frac{2q_X(1-q_Y)H_b^{-1}(R)}{(1-H_b^{-1}(R))^2}, & q_X > \frac{1}{2}, q_Y < \frac{1}{2}.
\end{cases}
\end{aligned}
$$

Based on the above expressions, one can easily verify (A.1.5) and (A.1.6). In particular, it is worth noting that when $q_X, q_Y \leq \frac{1}{2}$ or $q_X, q_Y \geq \frac{1}{2}$,

$$\overline{D}_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R) = D_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R), \quad R \in [0, \min\{H_b(q_X), H_b(q_Y)\}],$$

i.e., there is no penalty for using optimal quantizer and dequantizer in the conventional rate-distortion sense.

Remark: It is easy to verify that

$$
\begin{aligned}
D_{\text{cr}}(\mathcal{B}(q_X), R) &\triangleq \min_{q_Y\in[0,1]} D_{\text{cr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)\\
&=\begin{cases}
q_X\left(1-\frac{R}{H_b(q_X)}\right), & R\in[0,H_b(q_X)], q_X \leq \frac{1}{2},\\
(1-q_X)\left(1-\frac{R}{H_b(q_X)}\right), & R\in[0,H_b(q_X)], q_X > \frac{1}{2},
\end{cases}\\
D_{\text{ncr}}(\mathcal{B}(q_X), R) &\triangleq \min_{q_Y\in[0,1]} D_{\text{ncr}}(\mathcal{B}(q_X), \mathcal{B}(q_Y), R)\\
&=\begin{cases}
q_X - H_b^{-1}(R), & R\in[0,H_b(q_X)], q_X \leq \frac{1}{2},\\
1-q_X - H_b^{-1}(R), & R\in[0,H_b(q_X)], q_X > \frac{1}{2},
\end{cases}
\end{aligned}
$$

which are respectively the conventional one-shot distortion-distortion function (or equivalently, one-shot distortion-rate-perception function with an inactive perception constraint) for $\mathcal{B}(q_X)$ with and without common randomness. In general, $D_{\mathrm{ncr}}(\mathcal{B}(q_X), R)$ is different from $D_{\mathrm{mse}}(\mathcal{B}(q_X), R)$ (the former is strictly above the latter). The reason is as follows: even though the output distribution constraint is removed in the definition of $D_{\mathrm{ncr}}(\mathcal{B}(q_X), R)$, the output alphabet remains to be $\{0, 1\}$; in contrast, for $D_{\mathrm{mse}}(\mathcal{B}(q_X), R)$, the output alphabet is relaxed to $\mathbb{R}$. □

A.1.5 Auxiliary Results

Lemma 1 (Finite Support Approximation)**.** *For any $R \geq 0$ and $\epsilon > 0$, there exists a joint distribution $p_{X,\hat{X},\hat{Y},Y}$ compatible with the given marginal distributions p_X and p_Y such that $X \leftrightarrow \hat{X} \leftrightarrow \hat{Y} \leftrightarrow Y$ form a Markov chain and $\hat{X}$ is a finite-support random variable with $H(\hat{X}) \leq R$ (or $\hat{Y}$ is a finite-support random variable with $H(\hat{Y}) \leq R$); moreover, $\mathbb{E}[X|\hat{X}] = \hat{X}$, $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$, and*

$$\mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2] \leq D_{\mathrm{ncr}}(p_X, p_Y, R) + \epsilon.$$

Proof. In light of Theorem 1, we can find $p_{X,\hat{X},\hat{Y},Y}$ such that $X \leftrightarrow \hat{X} \leftrightarrow \hat{Y} \leftrightarrow Y$ form a Markov chain, $H(\hat{X}) \leq R$, $\mathbb{E}[X|\hat{X}] = \hat{X}$, $\mathbb{E}[Y|\hat{Y}]$, and

$$\mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2] \leq D_{\mathrm{ncr}}(p_X, p_Y, R) + \frac{\epsilon}{2}. \tag{A.1.17}$$

The proof is complete if $\hat{X}$ is a finite-support random variable. So it suffices to consider the case where $\hat{X}$ takes value from some countably infinite set $\{\hat{x}_i\}_{i=1}^{\infty}$. Since $\mathbb{E}[\|X\|^2] < \infty$ and $\mathbb{E}[\|Y\|^2] < \infty$, it follows that there exists a positive integer N such that

$$\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}\}\mathbb{E}[\|X\|^2+\|Y\|^2|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]\leq\frac{\epsilon}{4}.$$

Let $\hat{X}'\triangleq\hat{X}$ if $\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}$ and $\hat{X}'\triangleq\mathbb{E}[X|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]$ if $\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}$. Note that

$$\begin{aligned}
&\mathbb{E}[\|X-\hat{X}'\|^2]+\mathbb{E}[\|\hat{X}'-\hat{Y}\|^2]\\
&=\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}\}\mathbb{E}[\|X-\hat{X}'\|^2+\|\hat{X}'-\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}]\\
&\quad+\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}\}\mathbb{E}[\|X-\hat{X}'\|^2+\|\hat{X}'-\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]\\
&=\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}\}\mathbb{E}[\|X-\hat{X}\|^2+\|\hat{X}-\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}]\\
&\quad+\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}\}\mathbb{E}[\|X-\hat{X}'\|^2+\|\hat{X}'-\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]\\
&\leq\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}\}\mathbb{E}[\|X-\hat{X}\|^2+\|\hat{X}-\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}]\\
&\quad+\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}\}\mathbb{E}[\|X-\hat{X}'\|^2+2\|\hat{X}'\|^2+2\|\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]\\
&\leq\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}\}\mathbb{E}[\|X-\hat{X}\|^2+\|\hat{X}-\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=1}^{N-1}]\\
&\quad+\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}\}\mathbb{E}[\|X\|^2+\|\hat{X}'\|^2+2\|\hat{Y}\|^2|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]\\
&\leq\mathbb{E}[\|X-\hat{X}\|^2+\|\hat{X}-\hat{Y}\|^2]+2\mathbb{P}\{\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}\}\mathbb{E}[\|X\|^2+\|Y\|^2|\hat{X}\in\{\hat{x}_i\}_{i=N}^{\infty}]\\
&\leq\mathbb{E}[\|X-\hat{X}\|^2]+\mathbb{E}[\|\hat{X}-\hat{Y}\|^2]+\frac{\epsilon}{2}.
\end{aligned}\tag{A.1.18}$$

Now define a new joint distribution $p_{X,\hat{X}'',\hat{Y}'',Y}$ such that $p_{X,\hat{X}''}=p_{X,\hat{X}'}$, $p_{\hat{X}'',\hat{Y}''}=p_{\hat{X}',\hat{Y}}$, $p_{Y,\hat{Y}''}=p_{Y,\hat{Y}}$, and $X\leftrightarrow\hat{X}''\leftrightarrow\hat{Y}''\leftrightarrow Y$ form a Markov chain. It is clear that $\hat{X}''$ is a finite-support random variable with $H(\hat{X}'')=H(\hat{X}')\leq H(\hat{X})\leq R$, $\mathbb{E}[X|\hat{X}'']=\hat{X}''$,

$\mathbb{E}[Y|\hat{Y}''] = \hat{Y}''$, and

$$\begin{aligned}
&\mathbb{E}[\|X-\hat{X}''\|^2]+\mathbb{E}[\|Y-\hat{Y}''\|]+\mathbb{E}[\|\hat{X}''-\hat{Y}''\|^2]\\
&\leq \mathbb{E}[\|X-\hat{X}'\|^2]+\mathbb{E}[\|Y-\hat{Y}\|]+\mathbb{E}[\|\hat{X}'-\hat{Y}\|^2]\\
&\leq \mathbb{E}[\|X-\hat{X}\|^2]+\mathbb{E}[\|Y-\hat{Y}\|]+\mathbb{E}[\|\hat{X}-\hat{Y}\|^2]+\frac{\epsilon}{2} \qquad \text{(A.1.19)}\\
&\leq D_{\mathrm{ncr}}(p_X,p_Y,R)+\epsilon, \qquad \text{(A.1.20)}
\end{aligned}$$

where (A.1.19) and (A.1.20) are due to (A.1.17) and (A.1.18), respectively. This completes the proof of Lemma 1. □

Lemma 2 (Deterministic on Finite Support)**.** *Let $X \leftrightarrow \hat{X} \leftrightarrow \hat{Y} \leftrightarrow Y$ be a Markov chain with $\mathbb{E}[X|\hat{X}] = \hat{X}$, $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$, and assume that $\hat{X}$ (or $\hat{Y}$) is a finite-support random variable. There exist deterministically related random variables $\hat{X}'$ and $\hat{Y}'$, with the support size no greater than that of $\hat{X}$ and $H(\hat{X}') = H(\hat{Y}') \leq H(\hat{X})$ (or $H(\hat{X}') = H(\hat{Y}') \leq H(\hat{Y})$), such that $X \leftrightarrow \hat{X}' \leftrightarrow \hat{Y}' \leftrightarrow Y$ form a Markov chain, $\mathbb{E}[X|\hat{X}'] = \hat{X}'$, $\mathbb{E}[Y|\hat{Y}'] = \hat{Y}'$, and*

$$\begin{aligned}
&\mathbb{E}[\|X-\hat{X}'\|^2]+\mathbb{E}[\|Y-\hat{Y}'\|]+\mathbb{E}[\|\hat{X}'-\hat{Y}'\|^2]\\
&=\mathbb{E}[\|X-\hat{X}\|^2]+\mathbb{E}[\|Y-\hat{Y}\|]+\mathbb{E}[\|\hat{X}-\hat{Y}\|^2].
\end{aligned}$$

Proof. Let $\tilde{Y} \triangleq \mathbb{E}[Y|\hat{X}]$. Since $\tilde{Y} \leftrightarrow \hat{Y} \leftrightarrow Y$ form a Markov chain and $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$, we have $\tilde{Y} = \mathbb{E}[\hat{Y}|\hat{X}]$. Construct a new joint distribution $p_{X,\hat{X}',\hat{Y}',Y}$ such that $p_{X,\hat{X}'} = p_{X,\hat{X}}$, $p_{\hat{X}',\hat{Y}'} = p_{\hat{X},\tilde{Y}}$, $p_{Y,\hat{Y}'} = p_{Y,\tilde{Y}}$, and $X \leftrightarrow \hat{X}' \leftrightarrow \hat{Y}' \leftrightarrow Y$ form a Markov chain. It is clear that $\mathbb{E}[X|\hat{X}'] = \hat{X}'$, $\mathbb{E}[Y|\hat{Y}'] = \hat{Y}'$, and

$$\begin{aligned}
&\mathbb{E}[\|X - \hat{X}'\|^2] + \mathbb{E}[\|Y - \hat{Y}'\|^2] + \mathbb{E}[\|\hat{X}' - \hat{Y}'\|^2] \\
&= \mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \tilde{Y}\|^2] + \mathbb{E}[\|\hat{X} - \tilde{Y}\|^2] \\
&= \mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + \mathbb{E}[\|\hat{Y} - \tilde{Y}\|^2] + \mathbb{E}[\|\hat{X} - \tilde{Y}\|^2] \\
&= \mathbb{E}[\|X - \hat{X}\|^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + \mathbb{E}[\|\hat{X} - \hat{Y}\|^2],
\end{aligned}$$

where the last equality is due to $\tilde{Y} = \mathbb{E}[\hat{Y}|\hat{X}]$. If the function that maps $\hat{X}'$ to $\hat{Y}'$ (or equivalently, maps $\hat{X}$ to $\tilde{Y}$) is invertible, then $\hat{X}$, $\hat{X}'$, $\hat{Y}'$ have the same support size and $H(\hat{X}) = H(X') = H(Y')$, which completes the proof. Otherwise, the support size of $\hat{Y}'$ must be strictly smaller than that of $\hat{X}'$ (which is the same as that of $\hat{X}$) and $H(\hat{Y}') < H(\hat{X}') = H(\hat{X})$. We can alternately reduce the support sizes of $\hat{X}'$ and $\hat{Y}'$ using this argument until they are deterministically related (and consequently have the same support size and the same entropy). This can be accomplished in a finite number of steps because the reduction in support size cannot continue forever. □

□

A.1.6 Uniform Distribution

Let $X \sim \text{Unif}[0, a]$ and $Y \sim \text{Unif}[0, b]$ be uniformly distributed random variables, where $a, b > 0$. Note that the density functions are given as: $p_X(x) = \frac{1}{a}, 0 \leq x \leq a$ and $p_Y(y) = \frac{1}{b}, 0 \leq y \leq b$ and $p_X(x)$ and $p_Y(y)$ are zero outside these intervals. The

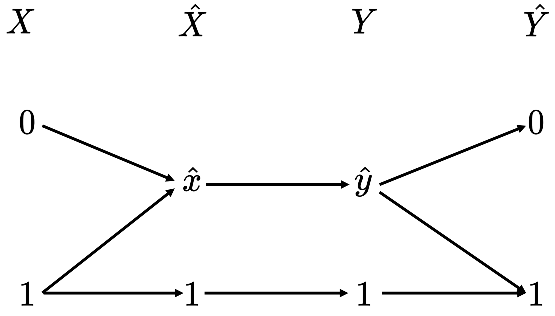


Figure A.1: Illustration of the entropy-constrained optimal transport plan for the binary case (assuming $q_X + q_Y \leq 1$), where $p_{\hat{X}}(\hat{x}) = p_{\hat{Y}}(\hat{y}) = 1 - H_b^{-1}(R)$ with $\hat{x} = \frac{q_X - H_b^{-1}(R)}{1 - H_b^{-1}(R)}$ and $\hat{y} = \frac{q_Y - H_b^{-1}(R)}{1 - H_b^{-1}(R)}$. It is interesting to note that the quantizer $p_{\hat{X}|X}$ does not depend on p_Y while the dequantizer $p_{Y|\hat{Y}}$ does not depend on p_X. So they are decoupled in a certain sense. Moreover, $p_{\hat{X}|X}$ and $p_{Y|\hat{Y}}$ coincide respectively with optimal quantizer $p_{\hat{X}^*|X}$ and dequantizer $p_{Y|\hat{Y}^*}$ in the conventional rate-distortion sense when $q_X, q_Y \leq 1/2$.

cumulative density functions of X and Y are given as follows:

$$C_X(x) = \begin{cases} 0, & x \leq 0, \\ \frac{x}{a}, & 0 \leq x \leq a, \\ 1, & x \geq a, \end{cases} \qquad C_Y(y) = \begin{cases} 0, & y \leq 0, \\ \frac{y}{b}, & 0 \leq y \leq b, \\ 1, & y \geq b. \end{cases}$$

Following Peyré and Cuturi [98, Remark 2.30] we have that the optimal transport (without rate constraint) is given by:

$$W_2^2(p_X, p_Y) = \int_0^1 (C_X^{-1}(r) - C_Y^{-1}(r))^2 dr = \frac{(b-a)^2}{3}, \tag{A.1.21}$$

where $C_X^{-1}(\cdot)$ and $C_Y^{-1}(\cdot)$ are the pseudo-inverse of the CDF functions for X and Y as defined in Peyré and Cuturi [98, Remark 2.30].

We will next develop an upper bound on $D_{\text{ncr}}(p_X, p_Y, R)$ using Theorem 1 when the rate is of the form $R = \log_2(N)$ for any $N \in \{1, 2, \ldots\}$, by considering the following choice for $\hat{X}$ and $\hat{Y}$:

$$\hat{X} \in \hat{\mathcal{X}} = \left\{ \frac{a}{2N}, \frac{3a}{2N}, \frac{5a}{2N}, \ldots, \frac{(2N-1)a}{2N} \right\}, \tag{A.1.22}$$

$$\hat{Y} \in \hat{\mathcal{Y}} = \left\{ \frac{b}{2N}, \frac{3b}{2N}, \frac{5b}{2N}, \ldots, \frac{(2N-1)b}{2N} \right\}. \tag{A.1.23}$$

To compute the upper bound we select $p_{\hat{X}|X}$ to correspond to scalar quantization of X i.e., given X we select $\hat{X}$ as a point in $\hat{\mathcal{X}}$ closest to X. The distribution $p_{\hat{Y}|Y}$ is defined in an analogous manner[1] . Our upper bound can be computed as:

$$D^{+}_{\mathrm{ncr}}(p_X, p_Y, R) = \mathbb{E}(X - \hat{X})^2 + \mathbb{E}(Y - \hat{Y})^2 + W_2^2(p_{\hat{X}}, p_{\hat{Y}}). \tag{A.1.24}$$

Note that with $\Delta = \frac{1}{N}$ we have that $\mathbb{E}(X - \hat{X})^2 = \frac{a^2\Delta^2}{12}$ and $\mathbb{E}(Y - \hat{Y})^2 = \frac{b^2\Delta^2}{12}$. Thus we only need to compute the third term. Following Peyré and Cuturi [98, Remark 2.28] we have that:

$$W_2^2(p_{\hat{X}}, p_{\hat{Y}}) = \frac{(b-a)^2\Delta^2}{4N} \sum_{i=1}^{N} (2i-1)^2 = \frac{(b-a)^2}{3} \left(1 - \frac{\Delta^2}{4} \right). \tag{A.1.25}$$

Thus using $\Delta^2 = 2^{-2R}$, we have that

$$D^{+}_{\mathrm{ncr}}(p_X, p_Y, R) = \frac{(b-a)^2}{3} + \frac{a \cdot b}{6} 2^{-2R}. \tag{A.1.26}$$

[1] Please note that we do not claim that the proposed choice is optimal with respect to D_{mse} in (5.3.4) although it is known to be optimal solution for a related problem - the entropy constrained scalar quantization ([47]). As a result we cannot claim to compute the upper bound $\bar{D}_{\mathrm{ncr}}$ stated Theorem 1 but provide another upper bound.

Note that the upper bound approaches the lower bound in (A.1.21) as $R \to \infty$, with exponential rate of convergence.

For the case of general R, we let $N = \lceil 2^R \rceil$ and following Gyorgy and Linder [47, Theorem 1], we select

$$\begin{aligned} \hat{X} \in \hat{\mathcal{X}} &= \left\{ \underbrace{\frac{ac}{2}}_{\hat{x}_1}, \underbrace{a\left(c + \frac{c'}{2}\right)}_{\hat{x_2}}, \underbrace{a\left(c + 3\frac{c'}{2}\right)}_{\hat{x_3}}, \ldots, \underbrace{a\left(c + (2N-3)\frac{c'}{2}\right)}_{\hat{x}_N} \right\}, \\ \hat{Y} \in \hat{\mathcal{Y}} &= \left\{ \underbrace{\frac{bc}{2}}_{\hat{y}_1}, \underbrace{b\left(c + \frac{c'}{2}\right)}_{\hat{y}_2}, \underbrace{b\left(c + 3\frac{c'}{2}\right)}_{\hat{y}_3}, \ldots, \underbrace{b\left(c + (2N-3)\frac{c'}{2}\right)}_{\hat{y}_N} \right\}, \end{aligned} \tag{A.1.27}$$

where c is the unique solution in the interval $(0, 1/N]$ to the equation:

$$-c \log c - (1-c) \log \frac{(1-c)}{N-1} = R,$$

and $c' = \frac{(1-c)}{N-1}$ holds. Note that the length of the first interval is c and the length of all other intervals is c'. In the special case where $R = \log_2 N$ we will have that $c = c' = \frac{1}{N}$ and our construction for $\hat{X}$ and $\hat{Y}$ is consistent with the previous case.

As before we use $p_{\hat{X}|X}$ and $p_{\hat{Y}|Y}$ to be the distributions associated with scalar quantization. Thus we have that:

$$\mathbb{E}(X - \hat{X})^2 = c\frac{a^2c^2}{12} + (1-c)\frac{a^2c'^2}{12}, \tag{A.1.28}$$

$$\mathbb{E}(Y - \hat{Y})^2 = c\frac{b^2c^2}{12} + (1-c)\frac{b^2c'^2}{12}. \tag{A.1.29}$$

Furthermore using the result stated in Peyré and Cuturi [98, Remark 2.30] we have

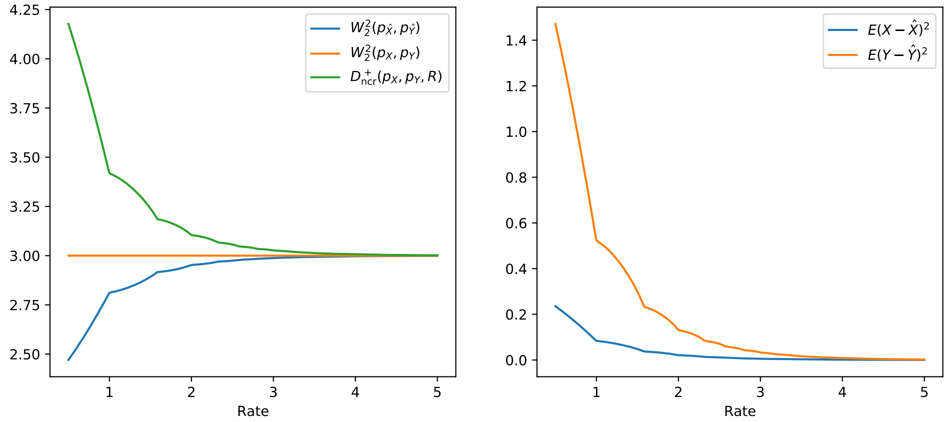


Figure A.2: Example of uniform sources with $a = 2$ and $b = 5$. The left plot shows the lower bound $W_2^2(p_X, p_Y)$ in (A.1.21) and the upper bound $D^+_{\text{ncr}}(p_X, p_Y, R)$ which is the sum of of the right hand side in (A.1.28), (A.1.29) and (A.1.32). For comparison we also show the value of $W_2^2(p_{\hat{X}}, p_{\hat{Y}})$. The right plot shows the distortions associated with the quantization and dequantization steps.

that

$$
\begin{aligned}
W_2^2(p_{\hat{X}}, p_{\hat{Y}}) &= c(\hat{x}_1 - \hat{y}_1)^2 + c' \sum_{j=2}^{N} (\hat{x}_j - \hat{y}_j)^2 && \text{(A.1.30)} \\
&= (b-a)^2 \frac{c^3}{4} + (b-a)^2 c' \sum_{j=1}^{N-1} \left(c + \frac{2j-1}{2} c' \right)^2 && \text{(A.1.31)} \\
&= (b-a)^2 \left(\frac{c^3}{4} + c^2 c'(N-1) + cc'^2 (N-1)^2 + \frac{1}{12} c'^3 (2N-1)(2N-3)(N-1) \right). && \text{(A.1.32)}
\end{aligned}
$$

Finally, the upper bound $D^+_{\text{ncr}}(p_X, p_Y, R)$ can be obtained by summing the right hand side of (A.1.28), (A.1.29) and (A.1.32). We provide a numerical evaluation of this upper bound in Fig. A.2.

A.1.7 Asymptotic Optimal Transport

Let $X_1, X_2, \cdots$ and $Y_1, Y_2, \cdots$ be i.i.d. processes with marginal distributions p_X and p_Y, respectively.

Definition 4 (Asymptotic Optimal Transport with Entropy Bottleneck — no common randomness)**.** *The asymptotic optimal transport from p_X to p_Y with an entropy bottleneck of R and without common randomness is defined as*

$$D^{(\infty)}_{\mathrm{ncr}}(p_X, p_Y, R) \triangleq \inf_{n \geq 1} D^{(n)}_{\mathrm{ncr}}(p_X, p_Y, R),$$

where

$$\begin{aligned} D^{(n)}_{\mathrm{ncr}}(p_X, p_Y, R) \triangleq & \inf_{p_{X^n,Z,Y^n} \in M_{\mathrm{ncr}}(\otimes_{i=1}^n p_X, \otimes_{i=1}^n p_Y)} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[d(X_i, Y_i)] \\ & s.t. \quad \frac{1}{n} H(Z) \leq R. \end{aligned}$$

Remark: It is clear that $D^{(1)}_{\mathrm{ncr}}(p_X, p_Y, R) = D_{\mathrm{ncr}}(p_X, p_Y, R)$. Moreover, one can readily show that $\{nD^{(n)}_{\mathrm{ncr}}(p_X, p_Y, R)\}_{n=1}^{\infty}$ is a subadditive sequence and consequently $D^{(\infty)}_{\mathrm{ncr}}(p_X, p_Y, R) = \lim_{n\to\infty} D^{(n)}_{\mathrm{ncr}}(p_X, p_Y, R)$.

Theorem 5. *We have*

$$\begin{aligned} D^{(\infty)}_{\mathrm{ncr}}(p_X, p_Y, R) = & \inf_{p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)} \mathbb{E}[d(X, Y)] \\ & s.t. \quad \max\{I(X; Z), I(Y; Z)\} \leq R. \end{aligned}$$

Proof. This result can be specialized from Theorem 1 in [113]. □

The following result is the counterpart of Theorem 1 in the asymptotic setting.

Theorem 6. *Let*

$$D^{(\infty)}_{\mathrm{mse}}(p_X,p_Y,R) \triangleq \inf_{p_{\hat{X}|X},p_{\hat{Y}|Y}} \mathbb{E}[\|X-\hat{X}\|^2]+\mathbb{E}[\|Y-\hat{Y}\|^2]+W_2^2(p_{\hat{X}},p_{\hat{Y}})$$
$$s.t.\quad \mathbb{E}[X|\hat{X}]=\hat{X},\quad \mathbb{E}[Y|\hat{Y}]=\hat{Y},\quad I(X;\hat{X})\le R,\quad I(Y;\hat{Y})\le R, \tag{A.1.33}$$

and

$$D_{\mathrm{mse}}(p_X,R) \triangleq \inf_{p_{\hat{X}|X}} \mathbb{E}[\|X-\hat{X}\|^2] \quad s.t.\quad I(X;\hat{X})\le R. \tag{A.1.34}$$

Moreover, let

$$\overline{D}^{(\infty)}_{\mathrm{ncr}}(p_X,p_Y,R) \triangleq D^{(\infty)}_{\mathrm{mse}}(p_X,R)+D^{(\infty)}_{\mathrm{mse}}(p_Y,R)+W_2^2(p_{\hat{X}^*},p_{\hat{Y}^*}), \tag{A.1.35}$$

$$\underline{D}^{(\infty)}_{\mathrm{ncr}}(p_X,p_Y,R) \triangleq D^{(\infty)}_{\mathrm{mse}}(p_X,R)+D^{(\infty)}_{\mathrm{mse}}(p_Y,R), \tag{A.1.36}$$

where $p_{\hat{X}^*}$ and $p_{\hat{Y}^*}$ are the marginal distributions induced by the minimizers $p_{\hat{X}^*|X}$ and $p_{\hat{Y}^*|Y}$ that attain $D^{(\infty)}_{\mathrm{mse}}(p_X,R)$ and $D^{(\infty)}_{\mathrm{mse}}(p_Y,R)$, respectively (assuming the existence and uniqueness of such minimizers). Then under the squared Eucledian distortion measure,

$$D^{(\infty)}_{\mathrm{ncr}}(p_X,p_Y,R) = D^{(\infty)}_{\mathrm{mse}}(p_X,p_Y,R). \tag{A.1.37}$$

In addition, we have

$$\overline{D}^{(\infty)}_{\mathrm{ncr}}(p_X,p_Y,R) \ge D^{(\infty)}_{\mathrm{ncr}}(p_X,p_Y,R) \ge \underline{D}^{(\infty)}_{\mathrm{ncr}}(p_X,p_Y,R), \tag{A.1.38}$$

and both inequalities are tight when $p_X=p_Y$.

Proof. For any $p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)$ with $\max\{I(X;Z), I(Y;Z)\} \leq R$,

$$\begin{aligned}\mathbb{E}[\|X-Y\|^2] &= \mathbb{E}[\|X-\mathbb{E}[X|Z]\|^2] + \mathbb{E}[\|Y-\mathbb{E}[Y|Z]\|^2] + \mathbb{E}[\|\mathbb{E}[X|Z]-\mathbb{E}[Y|Z]\|^2] \\ &\geq D^{(\infty)}_{\mathrm{mse}}(p_X, p_Y, R),\end{aligned}$$

where the last inequality follows from the definition of $D^{(\infty)}_{\mathrm{mse}}(p_X, p_Y, R)$ and the fact that

$$\max\{I(X;\mathbb{E}[X|Z]), I(Y;\mathbb{E}[Y|Z]\} \leq \max\{I(X;Z), I(Y;Z)\} \leq R.$$

In view of Theorem 5, we must have $D^{(\infty)}_{\mathrm{ncr}}(p_X, p_Y, R) \geq D^{(\infty)}_{\mathrm{mse}}(p_X, p_Y, R)$. On the other hand, for any $p_{\hat{X}|X}$, $p_{\hat{Y}|Y}$ with $\mathbb{E}[X|\hat{X}] = \hat{X}$, $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$, $I(X;\hat{X}) \leq R$, and $I(Y;\hat{Y}) \leq R$, we can construct a joint distribution $p_{X,\hat{X},\hat{Y},Y}$ such that $X \leftrightarrow \hat{X} \leftrightarrow \hat{Y} \leftrightarrow Y$ form a Markov chain, $p_{X,\hat{X}} = p_X p_{\hat{X}|X}$, $p_{Y,\hat{Y}} = p_Y p_{\hat{Y}|Y}$, and $p_{\hat{X},\hat{Y}}$ satisfying $\mathbb{E}[\|\hat{X}-\hat{Y}\|^2] = W_2^2(p_{\hat{X}}, p_{\hat{Y}})$. Note that

$$\begin{aligned}\mathbb{E}[\|X-Y\|^2] &= \mathbb{E}[\|X-\hat{X}\|^2] + \mathbb{E}[\|Y-\hat{Y}\|^2] + \mathbb{E}[\|\hat{X}-\hat{Y}\|^2] \\ &= \mathbb{E}[\|X-\hat{X}\|^2] + \mathbb{E}[\|Y-\hat{Y}\|^2] + W_2^2(p_{\hat{X}}, p_{\hat{Y}}). \qquad \text{(A.1.39)}\end{aligned}$$

Let $Z \triangleq \hat{X}$. It can be verified that $p_{X,Z,Y} \in M_{\mathrm{ncr}}(p_X, p_Y)$ and $\max\{I(X;Z), I(Y;Z)\} = \max\{I(X;\hat{X}), I(Y;\hat{X})\} \leq \max\{I(X;\hat{X}), I(Y;\hat{Y})\} \leq R$, which, together with (A.1.39), implies $D^{(\infty)}_{\mathrm{ncr}}(p_X, p_Y, R) \leq D^{(\infty)}_{\mathrm{mse}}(p_X, p_Y, R)$. This completes the proof of (A.1.37).

Dropping the term $W_2^2(p_{\hat{X}}, p_{\hat{Y}})$ in (A.1.33) yields

$$D^{(\infty)}_{\mathrm{ncr}}(p_X, p_Y, R) \geq \tilde{D}^{(\infty)}_{\mathrm{mse}}(p_X, R) + \tilde{D}^{(\infty)}_{\mathrm{mse}}(p_Y, R),$$

where

$$\tilde{D}_{\mathrm{mse}}^{(\infty)}(p_X, R) \triangleq \inf_{p_{\hat{X}|X}} \mathbb{E}[\|X - \hat{X}\|^2]$$
$$\text{s.t.} \quad \mathbb{E}[X|\hat{X}] = \hat{X}, \quad I(X; \hat{X}) \leq R.$$

and $\tilde{D}_{\mathrm{mse}}(p_Y, R)$ is definely analogously. On the other hand, choosing $p_{\hat{X}|X} = p_{\hat{X}'|X}$ and $p_{\hat{Y}|Y} = p_{\hat{Y}'|Y}$ in (A.1.33) gives

$$D_{\mathrm{ncr}}^{(\infty)}(p_X, p_Y, R) \leq \tilde{D}_{\mathrm{mse}}^{(\infty)}(p_X, R) + \tilde{D}_{\mathrm{mse}}^{(\infty)}(p_Y, R) + W_2^2(p_{\hat{X}'}, p_{\hat{Y}'}),$$

where $p_{\hat{X}'|X}$ and $p_{\hat{Y}'|Y}$ are the minimizers that attain $\tilde{D}_{\mathrm{mse}}^{(\infty)}(p_X, R)$ and $\tilde{D}_{\mathrm{mse}}^{(\infty)}(p_Y, R)$ respectively while $p_{\hat{X}'}$ and $p_{\hat{Y}'}$ are their induced marginal distributions. It is clear that $p_{\hat{X}'|X}$ and $p_{\hat{Y}'|Y}$ coincide with $p_{\hat{X}^*|X}$ and $p_{\hat{Y}^*|Y}$ respectively as the constraints $\mathbb{E}[X|\hat{X}] = \hat{X}$ and $\mathbb{E}[Y|\hat{Y}] = \hat{Y}$ are automatically satisfied by $p_{\hat{X}^*|X}$ and $p_{\hat{Y}^*|Y}$. This proves (A.1.38). For the special case $p_X = p_Y$, we have $p_{\hat{X}^*|X} = p_{\hat{Y}^*|Y}$ and consequently the upper bound and the lower bound in (A.1.38) coincide. □

Definition 5 (Asymptotic Optimal Transport with Entropy Bottleneck — with Common Randomness)**.** *The asymptotic optimal transport from p_X to p_Y with entropy bottleneck R and common randomness is defined as*

$$D_{\mathrm{cr}}^{(\infty)}(p_X, p_Y, R) \triangleq \inf_{n \geq 1} D_{\mathrm{cr}}^{(n)}(p_X, p_Y, R),$$

where

$$D_{\mathrm{cr}}^{(n)}(p_X, p_Y, R) \triangleq \inf_{p_{U,X^n,Z,Y^n} \in M_{\mathrm{cr}}(\otimes_{i=1}^n p_X, \otimes_{i=1}^n p_Y)} \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[d(X_i, Y_i)]$$
$$s.t. \quad \frac{1}{n} H(Z|U) \leq R.$$

Remark: It is clear that $D_{\mathrm{cr}}^{(1)}(p_X, p_Y, R) = D_{\mathrm{cr}}(p_X, p_Y, R)$. Moreover, one can readily show that $\{nD_{\mathrm{cr}}^{(n)}(p_X, p_Y, R)\}_{n=1}^{\infty}$ is a subadditive sequence and consequently $D_{\mathrm{cr}}^{(\infty)}(p_X, p_Y, R) = \lim_{n\to\infty} D_{\mathrm{ncr}}^{(n)}(p_X, p_Y, R)$.

Theorem 7. *We have*

$$D_{\mathrm{cr}}^{(\infty)}(p_X, p_Y, R) = \inf_{p_{X,Y} \in \Gamma(p_X, p_Y)} \mathbb{E}[d(X, Y)]$$
$$s.t. \quad I(X; Y) \leq R.$$

Proof. This result is known (see Theorem 7 in [114]). It is possible to give a simpler proof of the achievability part by leveraging Theorem 3 and the strong data processing inequality Li and El Gamal [70]. The converse is based on standard information-theoretic arguments. □

A.1.8 Gaussian Case

Let $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$ and $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$ be two Gaussian random variables, and let $d(\cdot, \cdot)$ be the squared distortion measure (i.e., $d(x, y) = (x - y)^2$). Let $D_{\min}^{(G)} \triangleq (\mu_X - \mu_Y)^2 + (\sigma_X - \sigma_Y)^2$ and $D_{\max}^{(G)} \triangleq (\mu_X - \mu_Y)^2 + \sigma_X^2 + \sigma_Y^2$. Note that $D_{\min}^{(G)}$ is the squared Wasserstein-2 distance between $\mathcal{N}(\mu_X, \sigma_X^2)$ and $\mathcal{N}(\mu_Y, \sigma_Y)$, which is the minimum $\mathbb{E}[(X - Y)^2]$ achievable by coupling X and Y. On the other hand, we have $\mathbb{E}[(X-Y)^2] = D^{(G)}_{\max}$ for X, Y independent. It is clear that $D^{(G)}_{\min}$ and $D^{(G)}_{\max}$ are the infimum and supremum of $D^{(\infty)}_{\mathrm{ncr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R)$ (as well as $D^{(\infty)}_{\mathrm{cr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R)$), respectively.

Theorem 8. *Assume squared distortion measure. Under no common randomness, we have*

$$D^{(\infty)}_{\mathrm{ncr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R) = D^{(G)}_{\min} + 2\sigma_X\sigma_Y 2^{-2R}, \quad R\in[0,\infty). \tag{A.1.40}$$

Moreover,

$$\overline{D}^{(\infty)}_{\mathrm{ncr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R) = D^{(\infty)}_{\mathrm{ncr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R), \quad R\in[0,\infty), \tag{A.1.41}$$

$$\underline{D}^{(\infty)}_{\mathrm{ncr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R) = (\sigma^2_X+\sigma^2_Y)2^{-2R}, \quad R\in[0,\infty). \tag{A.1.42}$$

With common randomness,

$$D^{(\infty)}_{\mathrm{cr}}(\mathcal{N}(\mu_X,\sigma^2_X),\mathcal{N}(\mu_Y,\sigma^2_Y),R) = D^{(G)}_{\max} - 2\sigma_X\sigma_Y\sqrt{1-2^{-2R}}, \quad R\in[0,\infty). \tag{A.1.43}$$

Proof. Consider $p_{\hat{X}|X}$ and $p_{\hat{Y}|Y}$ such that $\mathbb{E}[X|\hat{X}]$, $\mathbb{E}[Y|\hat{Y}]$, $I(X;\hat{X}) \leq R$, and $I(Y;\hat{Y}) \leq R$. Denote the mean and the variance of $\hat{X}$ by $\mu_{\hat{X}}$ and $\sigma^2_{\hat{X}}$, respectively. Similarly, denote the mean and the variance of $\hat{Y}$ by $\mu_{\hat{Y}}$ and $\sigma^2_{\hat{Y}}$, respectively. Clearly, $\mu_{\hat{X}} = \mu_X$, $\mu_{\hat{Y}} = \mu_Y$, $\sigma^2_{\hat{X}} = \sigma^2_X - \mathbb{E}[(X-\hat{X})^2]$, and $\sigma^2_{\hat{Y}} = \sigma^2_Y - \mathbb{E}[(Y-\hat{Y})^2]$.

Moreover,

$$\begin{aligned} W_2^2(p_{\hat{X}}, p_{\hat{Y}}) &\geq W_2^2(\mathcal{N}(\mu_{\hat{X}}, \sigma_{\hat{X}}^2), \mathcal{N}(\mu_{\hat{Y}}, \sigma_{\hat{Y}}^2)) \\ &= (\mu_{\hat{X}} - \mu_{\hat{Y}})^2 + (\sigma_{\hat{X}} - \sigma_{\hat{Y}})^2 \\ &= (\mu_X - \mu_Y)^2 + (\sigma_{\hat{X}} - \sigma_{\hat{Y}})^2. \end{aligned}$$

So we have

$$\begin{aligned} &\mathbb{E}[(X - \hat{X})^2] + \mathbb{E}[\|Y - \hat{Y}\|^2] + W_2^2(p_{\hat{X}}, p_{\hat{Y}}) \\ &\geq \sigma_X^2 - \sigma_{\hat{X}}^2 + \sigma_Y^2 - \sigma_{\hat{Y}}^2 + (\mu_X - \mu_Y)^2 + (\sigma_{\hat{X}} - \sigma_{\hat{Y}})^2 \\ &= D_{\max}^{(G)} - 2\sigma_{\hat{X}}\sigma_{\hat{Y}}. \end{aligned} \tag{A.1.44}$$

It can be verified that

$$\begin{aligned} R &\geq I(X; \hat{X}) \\ &= \frac{1}{2}\log(2\pi e \sigma_X^2) - h(X|\hat{X}) \\ &\geq \frac{1}{2}\log(2\pi e \sigma_X^2) - h(X - \hat{X}) \\ &\geq \frac{1}{2}\log(2\pi e \sigma_X^2) - \frac{1}{2}\log(2\pi e \mathbb{E}[(X - \hat{X})^2]) \\ &= \frac{1}{2}\log\frac{\sigma_X^2}{\sigma_X^2 - \sigma_{\hat{X}}^2}, \end{aligned}$$

which implies

$$\sigma_{\hat{X}} \leq \sigma_X\sqrt{1 - 2^{-2R}}. \tag{A.1.45}$$

Similarly,

$$\sigma_{\hat{Y}} \leq \sigma_Y \sqrt{1-2^{-2R}}. \tag{A.1.46}$$

Substituting (A.1.45) and (A.1.46) into (A.1.44) and invoking (A.1.37) in Theorem 6 shows

$$D^{(\infty)}_{\mathrm{ncr}}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R) \geq D^{(G)}_{\min} + 2\sigma_X\sigma_Y 2^{-2R}.$$

To see that this lower bound is tight, we can let

$$X = \hat{X} + N, \tag{A.1.47}$$

$$Y = \hat{Y} + \hat{N}, \tag{A.1.48}$$

where $\hat{X} \sim \mathcal{N}(\mu_X, \sigma_X^2(1-2^{-2R}))$ is independent of $N \sim \mathcal{N}(0, \sigma_X^2 2^{-2R})$ while $\hat{Y} \sim \mathcal{N}(\mu_Y, \sigma_Y^2(1-2^{-2R}))$ is independent of $\hat{N} \sim \mathcal{N}(0, \sigma_Y^2 2^{-2R})$. This completes the proof of (A.1.40).

To prove (A.1.41) and (A.1.42), it suffices to note the well-known fact that $p_{\hat{X}|X}$ and $p_{\hat{Y}|Y}$ associated with (A.1.47) and (A.1.48) attain $D^{(\infty)}_{\mathrm{mse}}(\mathcal{N}(\mu_X,\sigma_X^2),R)$ and $D^{(\infty)}_{\mathrm{mse}}(\mathcal{N}(\mu_Y,\sigma_Y^2),R)$, respectively.

Now we proceed to prove (A.1.43). Consider $p_{X,Y} \in \Gamma(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2))$ with $I(X;Y) \leq R$. Let $\xi \triangleq \mathbb{E}[(X-\mu_X)(Y-\mu_Y)]$. We have

$$\mathbb{E}[(X-Y)^2] = D^{(G)}_{\max} - 2\xi. \tag{A.1.49}$$

Moreover,

$$\begin{aligned}R &\geq I(X;Y)\\ &= \frac{1}{2}\log(2\pi e\sigma_X^2) + \frac{1}{2}\log(2\pi e\sigma_Y^2) - h(X,Y)\\ &\geq \frac{1}{2}\log(2\pi e\sigma_X^2) + \frac{1}{2}\log(2\pi e\sigma_Y^2) - \frac{1}{2}\log((2\pi e)^2(\sigma_X^2\sigma_Y^2 - \xi))\\ &= \frac{1}{2}\log\frac{\sigma_X^2\sigma_Y^2}{\sigma_X^2\sigma_Y^2 - \xi^2},\end{aligned}$$

which implies

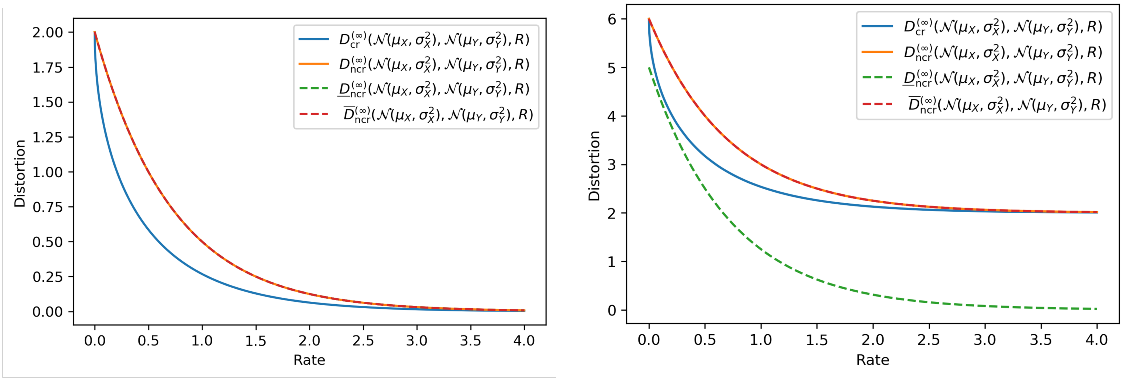
$$\xi \leq \sigma_X\sigma_Y\sqrt{1-2^{-2R}}. \tag{A.1.50}$$

Substituting (A.1.50) into (A.1.49) and invoking Theorem 7 shows

$$D_{\mathrm{cr}}^{(\infty)}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R) \geq D_{\max}^{(G)} - 2\sigma_X\sigma_Y\sqrt{1-2^{-2R}}.$$

To see that this lower bound is tight, we can let X and Y be jointly Gaussian with $\xi = \sigma_X\sigma_Y\sqrt{1-2^{-2R}}$. This completes the proof of Theorem 8. We acknowledge that the expression of $D_{\mathrm{cr}}^{(\infty)}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$ was established in [113, Section IV-B] for the special case when the Gaussian distributions have zero mean. However, to the best of our understanding, they only provided an upper bound for the case of no common randomness . □

In Figure A.3, We plot $D_{\mathrm{ncr}}^{(\infty)}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$, $\overline{D}_{\mathrm{ncr}}^{(\infty)}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$, $\underline{D}_{\mathrm{ncr}}^{(\infty)}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$, and $D_{\mathrm{cr}}^{(\infty)}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$ for two illustrative examples.



(a)

(b)

Figure A.3: Gaussian case distortion-rate tradeoffs. (a) $\mu_X = \mu_Y = 0$, $\sigma_X = \sigma_Y = 1$, where $\overline{D}^{(\infty)}_{\text{ncr}}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$ and $\underline{D}^{(\infty)}_{\text{ncr}}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$ coincide with $D^{(\infty)}_{\text{ncr}}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$; (b) $\mu_X = 0, \sigma_X = 1$, $\mu_Y = 1$, $\sigma_Y = 2$, where $\overline{D}^{(\infty)}_{\text{ncr}}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$ is tight but $\underline{D}^{(\infty)}_{\text{ncr}}(\mathcal{N}(\mu_X,\sigma_X^2),\mathcal{N}(\mu_Y,\sigma_Y^2),R)$ is loose. Moreover, it can be seen from both examples that common randomness can indeed help improve the distortion-rate tradeoff.

A.2 Experimental Results

A.2.1 Dataset

Image super-resolution is conducted on MNIST [63]. For synthesizing low-resolution images, we perform bilinear downsampling on the original image from 28 × 28 to 7 × 7. The samples in Figure 5.5(b) show that the low resolution digits are blurry and some of them are hard to recognize. Image denoising is conducted on SVHN [92]. In our experiments, we synthesize the noisy image with additive Gaussian noise. The standard deviation is set to 20.

A.2.2 Universal Quantization

Let $\mathcal{C}$ be our codebook for quantization. Recall that the encoder uses a tanh activation so its output lies in $(-1, 1)^d$. Given dimension d and L quantization levels per dimension as parameters, $\mathcal{C}$ will consist of L uniformly spaced intervals across all d dimensions. The upper bound of model rate is given by $d \log(L)$. With this codebook, universal quantization [43, 126, 162] is implemented as follows. We assume the sender and receiver have access to the same $u \sim U[-1/(L-1), +1/(L-1)]^d$. The sender computes

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

and sends z to the receiver. The receiver decodes the image by passing $z - u$ through the decoder. This is also known as a subtractive dither in literature [43]. For super-resolution and image denoising, the interval L is respectively fixed at 4 and 8.

A.2.3 Training Details

To induce distributional shift, we use the Wasserstein GAN for our experiments. We alternate between training the encoder/decoder f, g, and the critic h. By Kantorovich-Rubinstein duality [130], the critic is used to approximate

$$W_1(p_Y, p_{\tilde{Y}}) = \sup_{\|\nabla h\| \leq 1} \mathbb{E}h(Y) - \mathbb{E}h(\tilde{Y}), \tag{A.2.1}$$

where $\tilde{Y} = g(Q(f(X) + U) - U)$ for U as in Appendix A.2.2. The Lipschitz constraint is implemented with a gradient penalty [45] in practice.

Super-resolution. The training for end-to-end network lasts for 50 epochs. λ in (5.4.2) is fixed at $1e-3$ across all rates. The learning rate initialized to be 0.0001 and is decayed by a factor of 5 after 30 epochs. The Adam [58] optimizer is used. Table A.1 illustrates the detailed training setting. For the helper two-branch network at a specific rate constraint, we load the pre-trained encoder weight of the corresponding end-to-end network, as well as two randomly initialized decoders g_1, g_2. Note that only theese two decoders are trainable. During training, we use the Adam optimizer with the learning rate initialized at 0.0001. There are a total of 100 epochs until the convergence of the two decoders. The learning rate is decayed once at 50 epochs by a factor of 5. Detailed training settings are shown in Table A.2.

Image Denoising. The experiments for image denoising share many settings with image super-resolution. Tables A.1 and A.2 can be reused to reproduce the experiments on image denoising. Here, we list the difference between them. For denoising, the end-to-end model is trained for 100 epochs with λ fixed at $3e-3$ across all rates. The learning rate is decayed by a factor of 5 after 40 epochs. The two-branch model is trained for total 200 epochs and we decay the learning rate at 100 epochs by a factor of 5.

Table A.1: Hyperparameters used for training end-to-end model in Fig. 5.4(a). α is the learning rate, (β_1, β_2) are the parameters for Adam, and λ_{GP} is the gradient penalty coefficient.

	α	β_1	β_2	λ_{GP}
Encoder	10^{-4}	0.5	0.999	-
Decoder	10^{-4}	0.5	0.999	-
Critic	10^{-4}	0.5	0.999	10

A.2.4 Detailed Results in Figure. 5.5

In Fig. 5.5(a)(c), we have provided a comparison between the case with or without common randomness in the form of a scatter chart. Here, we present detailed

Table A.2: Hyperparameters used for training two-branch model in Fig. 5.4(b). α is the learning rate, (β_1, β_2) are the parameters for Adam, and λ_{GP} is the gradient penalty coefficient.

	α	β_1	β_2	λ_{GP}
Encoder	0	-	-	-
Decoder-1	10^{-4}	0.5	0.999	-
Decoder-2	10^{-4}	0.5	0.999	-
Critic	10^{-4}	0.5	0.999	10

quantities of each point in Fig. 5.5(a)(c). Table A.3 shows the number of each dot for super-resolution experiments, and Table A.4 present the value of each dot for image denoising. From the two tables, we can further see the utility of common randomness quantitatively.

Table A.3: The detailed number of MSE distortion loss in Fig. 5.5(a).

Super-resolution **with** Common Randomness

Rate	4	6	8	10	12	14	16	18
MSE	0.0515	0.0457	0.0420	0.0394	0.0372	0.0353	0.0339	0.0324
Rate	20	22	24	26	28	30	32	-
MSE	0.0313	0.0300	0.0297	0.0285	0.0280	0.0277	0.0269	-

Super-resolution **without** Common Randomness

Rate	4	6	8	10	12	14	16	18
MSE	0.0558	0.0506	0.0463	0.0435	0.0415	0.0396	0.0383	0.0367
Rate	20	22	24	26	28	30	32	-
MSE	0.0351	0.0337	0.0331	0.0328	0.0315	0.0308	0.0300	-

A.2.5 Comparison with Baseline

To illustrate the effectiveness of our system, we compare with a baseline method that separately deal with the tasks of image restoration and compression. For the

Table A.4: The detailed number of MSE distortion loss in Fig. 5.5(c).

Image Denoising **with** Common Randomness

Rate	12	18	24	30	36	42	48	54	60
MSE	0.0219	0.0195	0.0175	0.01634	0.0154	0.0147	0.0138	0.0135	0.0129
Rate	66	72	78	84	90	96	102	108	114
MSE	0.0126	0.0123	0.0118	0.0117	0.0115	0.0112	0.0109	0.0107	0.0106

Image Denoising **without** Common Randomness

Rate	12	18	24	30	36	42	48	54	60
MSE	0.0230	0.0208	0.0189	0.0175	0.0165	0.0157	0.0151	0.0145	0.014
Rate	66	72	78	84	90	96	102	108	114
MSE	0.0137	0.0133	0.0130	0.0127	0.0123	0.0120	0.0118	0.0116	0.0114

restoration (image super-resolution and denoising), we respectively build two U-Nets with skip connections and train them in the unsupervised manner by adopting the Eq. 5.4.2 as objective i.e.,

$$L_1 = \mathbb{E}[\|X - \tilde{Y}\|^2] + \lambda_1 W_1(p_Y, p_{\tilde{Y}}). \tag{A.2.2}$$

After the restoration networks are trained to converge, we fix their weights and use them to produce restored images $\tilde{Y}$ given degraded one X. Afterwards, we adopt our end-to-end network as compression network by minimizing the following loss at different rates:

$$L_1 = \mathbb{E}[\|\tilde{Y} - Y^+\|^2] + \lambda W_1(p_Y, p_{Y^+}), \tag{A.2.3}$$

where Y^+ is the outputs of compression network. Note that, to guarantee the distribution of reconstructed Y^+ is close to that of target images, we implement a penalty on the Wasserstein-1 distance in (A.2.2) and (A.2.3). For the image super-resolution, we experimentally selected $\lambda_1 = 0.05$ and $\lambda_2 = 0.01$. For the image denoising, we experimentally selected $\lambda_1 = 0.03$ and $\lambda_2 = 0.005$. Once the compression network is converged, we report the final MSE distortion between Y^+ and X using $\mathbb{E}[\|X - Y^+\|^2]$.

The detailed results for entropy-constrained image super-resolution and denoising are respectively shown in Table A.5 and Table A.6. It can easily check throughout the tables that our end-to-end systems outperform the baselines.

Table A.5: Comparison between our end-to-end system with the baseline method for image super-resolution. Numbers are the MSE distortion loss for a particular rate. Best results are in **bold**.

Super-resolution **with** Common Randomness

Rate	4	6	8	10	12	14	16	18
Baseline	0.0603	0.0568	0.0544	0.0530	0.0523	0.0511	0.0503	0.0498
Ours	**0.0515**	**0.0457**	**0.0420**	**0.0394**	**0.0372**	**0.0353**	**0.0339**	**0.0342**
Rate	20	22	24	26	28	30	32	-
Baseline	0.0489	0.0485	0.0484	0.0482	0.0478	0.0476	0.0471	-
Ours	**0.0313**	**0.0300**	**0.0297**	**0.0285**	**0.0280**	**0.0277**	**0.0269**	-

Super-resolution **without** Common Randomness

Rate	4	6	8	10	12	14	16	18
Baseline	0.0620	0.0585	0.0573	0.0555	0.0543	0.0535	0.0532	0.0520
Ours	**0.0558**	**0.0506**	**0.0463**	**0.0435**	**0.0415**	**0.0396**	**0.0383**	**0.0367**
Rate	20	22	24	26	28	30	32	-
Baseline	0.0517	0.0512	0.0506	0.0505	0.0497	0.0495	0.0490	-
Ours	**0.0351**	**0.0337**	**0.0331**	**0.0328**	**0.0315**	**0.0308**	**0.0300**	-

Table A.6: Comparison between our end-to-end system with the baseline method for image denoising. Numbers are the MSE distortion loss for a particular rate. Best results are in **bold**.

Image Denoising **with** Common Randomness

Rate	12	18	24	30	36	42	48	54	60
Baseline	0.0242	0.0213	0.0189	0.0173	0.0162	0.0154	0.0148	0.0142	0.0136
Ours	0.0219	0.0195	0.0175	0.0163	0.0154	0.0147	0.0138	0.0135	0.0129
Rate	66	72	78	84	90	96	102	108	114
Baseline	0.0134	0.0130	0.0127	0.0124	0.0120	0.0118	0.0116	0.0111	0.0110
Ours	0.0126	0.0123	0.0118	0.0117	0.0115	0.0112	0.0109	0.0107	0.0106

Image Denoising **without** Common Randomness

Rate	12	18	24	30	36	42	48	54	60
Baseline	0.0252	0.0216	0.0202	0.0185	0.0178	0.0164	0.0158	0.0154	0.0149
Ours	0.0230	0.0208	0.0189	0.0175	0.0165	0.0157	0.0151	0.0145	0.014
Rate	66	72	78	84	90	96	102	108	114
Baseline	0.0147	0.0144	0.0140	0.0135	0.0133	0.0129	0.0126	0.0120	0.0117
Ours	0.0137	0.0133	0.0130	0.0127	0.0123	0.0120	0.0118	0.0116	0.0114

A.2.6 Comparison with Ground Truth

In order to illustrate the rate-distortion trade-offs, we report the MSE distortion that is measured between *degraded input images* and decoder outputs in Figure 5.5. Since the input and output distributions are different, we do not expect MSE → 0 as the rate increases. The MSE distortion between degraded input and restored output is still able to reveal how much content information of the input is preserved in output (lower is better).

We now additionally show the MSE distortion between *ground truth* and decoder outputs in Tables A.7 and A.8. Concretely, we measure MSE distortion $\mathbb{E}[\|Y - \tilde{Y}\|^2]$, where Y is ground truth and $\tilde{Y}$ is the network output. Note that *for training, the ground truth is only used in an unsupervised fashion with unpaired noisy images*, and here the ground truth-noisy image pairs are only used for test time evaluation.

Note also that the MSE distortion is correspondingly lower if common randomness is adopted.

Table A.7: Illustration of MSE distortion between network outputs and ground truth for super-resolution.

Super-resolution **with** Common Randomness

Rate	4	8	12	16	20	24	28	32
MSE	0.0582	0.0464	0.0408	0.0380	0.0360	0.0353	0.0348	0.0343

Super-resolution **without** Common Randomness

Rate	4	8	12	16	20	24	28	32
MSE	0.0646	0.0492	0.0426	0.0390	0.0375	0.0362	0.0353	0.0345

Table A.8: Illustration of MSE distortion between network outputs and ground truth for image denoising.

Image Denoising **with** Common Randomness

Rate	12	24	36	48	60	72	84	96	108
MSE	0.0157	0.0114	0.0092	0.0077	0.0069	0.0062	0.0056	0.0052	0.0047

Image Denoising **without** Common Randomness

Rate	12	24	36	48	60	72	84	96	108
MSE	0.0169	0.0128	0.0104	0.0090	0.0080	0.0072	0.0066	0.0060	0.0057

A.2.7 Breakdown of the Table 5.1

It is worth reminding that the each number in Table 5.1 is the total distortion 5.4.5. Here, we provide a breakdown of total distortion in each of the three term for joint training, i.e, $\mathbb{E}[\|X - \tilde{Y}_1\|^2]$, $\mathbb{E}[\|\tilde{Y} - \tilde{Y}_2\|^2]$ and $\mathbb{E}[\|\tilde{Y}_1 - \tilde{Y}_2\|^2]$.

A.2.8 Network Architecture

Super-resolution. The detailed network structure for end-to-end model and two-branch model are respectively presented in Table A.10 and Table A.11. The last linear layer of the encoder controls the number of output symbols.

Denoising. The detailed network structure for end-to-end model and two-branch model are respectively presented in Table A.12 and Table A.13. The last linear layer of the encoder controls the number of output symbols.

Table A.9: Breakdown of the Table 5.1. At any rate, it can be observed that the total losses of our approximation system (L_2) are very close to that of end-to-end learning system under the setting without common randomness ($\mathbb{E}[\|X - \tilde{Y}\|^2]$).

Image Super-resolution Using Joint Training

Rate	$\mathbb{E}[\|X - \tilde{Y}_1\|^2]$	$\mathbb{E}[\|\tilde{Y} - \tilde{Y}_2\|^2]$	$\mathbb{E}[\|\tilde{Y}_1 - \tilde{Y}_2\|^2]$	L_2	$\mathbb{E}[\|X - \tilde{Y}\|^2]$
4	0.0355	0.0227	0.0004	**0.0586**	**0.0558**
10	0.0223	0.0222	0.0008	**0.0453**	**0.0435**
20	0.0136	0.0191	0.00013	**0.0349**	**0.0351**
30	0.00122	0.0172	0.0015	**0.0309**	**0.0308**

Image Denoising Using Joint Training

Rate	$\mathbb{E}[\|X - \tilde{Y}_1\|^2]$	$\mathbb{E}[\|\tilde{Y} - \tilde{Y}_2\|^2]$	$\mathbb{E}[\|\tilde{Y}_1 - \tilde{Y}_2\|^2]$	L_2	$\mathbb{E}[\|X - \tilde{Y}\|^2]$
12	0.01911	0.00497	0.00020	**0.02428**	**0.02302**
30	0.01458	0.00435	0.00026	**0.01919**	**0.01746**
60	0.01168	0.00378	0.00032	**0.01578**	**0.01401**
90	0.01035	0.00323	0.00028	**0.01386**	**0.01229**

Table A.10: Model architectures of end-to-end network used in super-resolution.

Encoder
Input
Conv2D, l-ReLU
Conv2D, l-ReLU
Flatten
Linear, l-ReLU
Linear, l-ReLU
Linear, Tanh
Quantizer

Decoder
Input
Linear, BatchNorm1D, l-ReLU
Linear, BatchNorm1D, l-ReLU
Unflatten
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, BatchNorm2D, Sigmoid

Critic
Input
Conv2D, l-ReLU
Conv2D, l-ReLU
Conv2D, l-ReLU
Linear

Table A.11: Model architectures of two-branch network used in super-resolution.

Encoder
Input
Conv2D, l-ReLU
Conv2D, l-ReLU
Flatten
Linear, l-ReLU
Linear, l-ReLU
Linear, Tanh
Quantizer

Decoder1 and 2
Input
Linear, BatchNorm1D, l-ReLU
Linear, BatchNorm1D, l-ReLU
Unflatten
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, Sigmoid

Table A.12: Model architectures of end-to-end network used in image denoising.

Encoder
Input
Conv2D, l-ReLU
Conv2D, l-ReLU
Conv2D, l-ReLU
Flatten
Linear, Tanh
Quantizer

Decoder
Input
Linear, BatchNorm1D, l-ReLU
Linear, BatchNorm1D, l-ReLU
Unflatten
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, BatchNorm2D, l-ReLU
ConvT2D, BatchNorm2D, Sigmoid

Critic
Input
Conv2D, l-ReLU
Conv2D, l-ReLU
Conv2D, l-ReLU
Linear

Table A.13: Model architectures of two-branch network used in image denoising. ResBlock is formed using two Conv2D and skip connection.

Encoder
Input
Conv2D, l-ReLU
Conv2D, l-ReLU
Conv2D, l-ReLU
Flatten
Linear, Tanh
Quantizer

Decoder1 and 2
Input
Linear, BatchNorm1D, l-ReLU
Linear, BatchNorm1D, l-ReLU
Unflatten
ConvT2D, l-ReLU
ResBlock, ConvT2D, l-ReLU
ResBlock, ConvT2D, l-ReLU
ResBlock, ConvT2D, Sigmoid

Bibliography

[1] Agustsson, E., Mentzer, F., Tschannen, M., Cavigelli, L., Timofte, R., Benini, L., and Van Gool, L. (2017). Soft-to-hard vector quantization for end-to-end learning compressible representations. In *Advances in Neural Information Processing Systems*, pages 1142–1152.

[2] Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., and Gool, L. V. (2019). Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 221–231.

[3] Aleotti, F., Tosi, F., Poggi, M., and Mattoccia, S. (2018). Generative adversarial networks for unsupervised monocular depth prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0.

[4] Alhashim, I. and Wonka, P. (2018). High quality monocular depth estimation via transfer learning. *arXiv preprint arXiv:1812.11941*.

[5] Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Hasan, M., Van Essen, B. C., Awwal, A. A., and Asari, V. K. (2019). A state-of-the-art survey on deep learning theory and architectures. *Electronics*, **8**(3), 292.

[6] Ancuti, C., Ancuti, C. O., and Timofte, R. (2018a). Ntire 2018 challenge on image dehazing: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 891–901.

[7] Ancuti, C. O., Ancuti, C., Timofte, R., and De Vleeschouwer, C. (2018b). O-haze: a dehazing benchmark with real hazy and haze-free outdoor images. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 754–762.

[8] Ancuti, C. O., Ancuti, C., Sbert, M., and Timofte, R. (2019a). Dense-haze: A benchmark for image dehazing with dense-haze and haze-free images. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1014–1018. IEEE.

[9] Ancuti, C. O., Ancuti, C., Timofte, R., Van Gool, L., Zhang, L., and Yang, M.-H. (2019b). Ntire 2019 image dehazing challenge report. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0.

[10] Ancuti, C. O., Ancuti, C., and Timofte, R. (2020). NH-HAZE: an image dehazing benchmark with non-homogeneous hazy and haze-free images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE CVPR 2020.

[11] Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223.

[12] Atapour-Abarghouei, A. and Breckon, T. P. (2018). Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2800–2810.

[13] Bai, Y., Wu, X., and Özgür, A. (2020). Information constrained optimal transport: From talagrand, to marton, to cover. In *2020 IEEE International Symposium on Information Theory (ISIT)*, pages 2210–2215. IEEE.

[14] Berman, D., Avidan, S., *et al.* (2016). Non-local image dehazing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1674–1682.

[15] Blau, Y. and Michaeli, T. (2018). The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237.

[16] Blau, Y. and Michaeli, T. (2019). Rethinking lossy compression: The rate-distortion-perception tradeoff. In *International Conference on Machine Learning*, pages 675–685.

[17] Cai, B., Xu, X., Jia, K., Qing, C., and Tao, D. (2016). Dehazenet: An end-to-end system for single image haze removal. *IEEE Transactions on Image Processing*, **25**(11), 5187–5198.

[18] Chen, D., He, M., Fan, Q., Liao, J., Zhang, L., Hou, D., Yuan, L., and Hua, G. (2019). Gated context aggregation network for image dehazing and deraining. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1375–1383.

[19] Chen, P.-Y., Liu, A. H., Liu, Y.-C., and Wang, Y.-C. F. (2019a). Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware

representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2624–2632.

[20] Chen, W., Fu, Z., Yang, D., and Deng, J. (2016). Single-image depth perception in the wild. In *Advances in neural information processing systems*, pages 730–738.

[21] Chen, W., Qian, S., and Deng, J. (2019b). Learning single-image depth from videos using quality assessment networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5604–5613.

[22] Chen, Z., Wang, Y., Yang, Y., and Liu, D. (2021). Psd: Principled synthetic-to-real dehazing guided by physical priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7180–7189.

[23] Chi, Z., Wang, Y., Yu, Y., and Tang, J. (2021). Test-time fast adaptation for dynamic scene deblurring via meta-auxiliary learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9137–9146.

[24] Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B. (2016). The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223.

[25] Cuturi, M. (2013). Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, **26**, 2292–2300.

[26] Deng, Q., Huang, Z., Tsai, C.-C., and Lin, C.-W. (2020). Hardgan: A haze-aware

representation distillation gan for single image dehazing. In *European Conference on Computer Vision*, pages 722–738. Springer.

[27] Deng, Z., Zhu, L., Hu, X., Fu, C.-W., Xu, X., Zhang, Q., Qin, J., and Heng, P.-A. (2019). Deep multi-model fusion for single-image dehazing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2453–2462.

[28] Dong, H., Pan, J., Xiang, L., Hu, Z., Zhang, X., Wang, F., and Yang, M.-H. (2020). Multi-scale boosted dehazing network with dense feature fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2157–2167.

[29] Dong, J. and Pan, J. (2020). Physics-based feature dehazing networks. In *European Conference on Computer Vision*, pages 188–204. Springer.

[30] Eigen, D. and Fergus, R. (2015). Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, pages 2650–2658.

[31] Eigen, D., Puhrsch, C., and Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374.

[32] El Gamal, A. and Kim, Y.-H. (2011). *Network information theory*. Cambridge university press.

[33] Fattal, R. (2014). Dehazing using color-lines. *ACM transactions on graphics (TOG)*, **34**(1), 1–14.

[34] Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pages 1126–1135. PMLR.

[35] Fourure, D., Emonet, R., Fromont, E., Muselet, D., Tremeau, A., and Wolf, C. (2017). Residual conv-deconv grid network for semantic segmentation. *arXiv preprint arXiv:1707.07958*.

[36] Fu, H., Gong, M., Wang, C., Batmanghelich, K., and Tao, D. (2018). Deep ordinal regression network for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2002–2011.

[37] Fu, M., Liu, H., Yu, Y., Chen, J., and Wang, K. (2021). Dw-gan: A discrete wavelet transform gan for nonhomogeneous dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 203–212.

[38] Garg, R., BG, V. K., Carneiro, G., and Reid, I. (2016). Unsupervised cnn for single view depth estimation: Geometry to the rescue. In *European Conference on Computer Vision*, pages 740–756. Springer.

[39] Geiger, A., Lenz, P., and Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE.

[40] Girshick, R. (2015). Fast R-CNN. In *IEEE International Conference on Computer Vision*, pages 1440–1448. IEEE.

[41] Godard, C., Mac Aodha, O., and Brostow, G. J. (2017). Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279.

[42] Godard, C., Mac Aodha, O., Firman, M., and Brostow, G. J. (2019). Digging into self-supervised monocular depth estimation. In *Proceedings of the IEEE international conference on computer vision*, pages 3828–3838.

[43] Gray, R. M. and Stockham, T. G. (1993). Dithered quantizers. *IEEE Transactions on Information Theory*, **39**(3), 805–812.

[44] Gu, S., Zhang, L., Zuo, W., and Feng, X. (2014). Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869.

[45] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777.

[46] Guo, X., Li, H., Yi, S., Ren, J., and Wang, X. (2018). Learning monocular depth by distilling cross-domain stereo networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 484–500.

[47] Gyorgy, A. and Linder, T. (2000). Optimal entropy-constrained scalar quantization of a uniform source. *IEEE Transactions on Information Theory*, **46**(7), 2704–2711.

[48] He, K., Sun, J., and Tang, X. (2010). Single image haze removal using dark channel prior. *IEEE transactions on pattern analysis and machine intelligence*, **33**(12), 2341–2353.

[49] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[50] He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

[51] Henaff, O. (2020). Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR.

[52] Hinton, G., Vinyals, O., and Dean, J. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

[53] Hong, M., Xie, Y., Li, C., and Qu, Y. (2020). Distilling image dehazing with heterogeneous task imitation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3462–3471.

[54] Huang, G., Liu, Z., Van Der Maaten, L., and Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708.

[55] Jiao, J., Cao, Y., Song, Y., and Lau, R. (2018). Look deeper into depth: Monocular depth estimation with semantic booster and attention-driven loss. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 53–69.

[56] Johnson, J., Alahi, A., and Fei-Fei, L. (2016). Perceptual losses for real-time style transfer and super-resolution. In *ECCV*.

[57] Joshi, M., Dredze, M., Cohen, W., and Rose, C. (2012). Multi-domain learning: when do domains matter? In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1302–1312.

[58] Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *3rd International Conference for Learning Representations (ICLR)*.

[59] Kligvasser, I., Shaham, T. R., and Michaeli, T. (2018). xUnit - Learning a Spatial Activation Function for Efficient Image Restoration. *CVPR*, pages 2433–2442.

[60] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. In *Conference on Neural Information Processing Systems*.

[61] Ladicky, L., Shi, J., and Pollefeys, M. (2014). Pulling things out of perspective. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 89–96.

[62] Laina, I., Rupprecht, C., Belagiannis, V., Tombari, F., and Navab, N. (2016). Deeper depth prediction with fully convolutional residual networks. In *2016 Fourth international conference on 3D vision (3DV)*, pages 239–248. IEEE.

[63] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, **86**(11), 2278–2324.

[64] Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A. P., Tejani, A., Totz, J., Wang, Z., and Shi, W. (2017). Photo-Realistic

Single Image Super-Resolution Using a Generative Adversarial Network. *CVPR*, pages 105–114.

[65] Lehtinen, J., Munkberg, J., Hasselgren, J., Laine, S., Karras, T., Aittala, M., and Aila, T. (2018). Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*.

[66] Li, B., Shen, C., Dai, Y., Van Den Hengel, A., and He, M. (2015a). Depth and surface normal estimation from monocular images using regression on deep features and hierarchical crfs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1119–1127.

[67] Li, B., Peng, X., Wang, Z., Xu, J.-Z., and Feng, D. (2017). Aod-net: All-in-one dehazing network. In *Proceedings of the IEEE International Conference on Computer Vision*.

[68] Li, B., Ren, W., Fu, D., Tao, D., Feng, D., Zeng, W., and Wang, Z. (2019). Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing*, **28**(1), 492–505.

[69] Li, B., Gou, Y., Liu, J. Z., Zhu, H., Zhou, J. T., and Peng, X. (2020). Zero-shot image dehazing. *IEEE Transactions on Image Processing*, **29**, 8457–8466.

[70] Li, C. T. and El Gamal, A. (2018). Strong functional representation lemma and applications to coding theorems. *IEEE Transactions on Information Theory*, **64**(11), 6967–6978.

[71] Li, R., Pan, J., Li, Z., and Tang, J. (2018). Single image dehazing via conditional

generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8202–8211.

[72] Li, Y., Tan, R. T., and Brown, M. S. (2015b). Nighttime haze removal with glow and multiple light colors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 226–234.

[73] Lin, T.-Y., Goyal, P., Girshick, R. B., He, K., and Dollár, P. (2017). Focal Loss for Dense Object Detection. *ICCV*, pages 2999–3007.

[74] Liu, F., Shen, C., Lin, G., and Reid, I. (2015). Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence*, **38**(10), 2024–2039.

[75] Liu, J., Wu, H., Xie, Y., Qu, Y., and Ma, L. (2020). Trident dehazing network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

[76] Liu, X., Ma, Y., Shi, Z., and Chen, J. (2019a). Griddehazenet: Attention-based multi-scale network for image dehazing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7314–7323.

[77] Liu, Y., Pan, J., Ren, J., and Su, Z. (2019b). Learning deep priors for image dehazing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2492–2500.

[78] Liu, Z., Xiao, B., Alrabeiah, M., Wang, K., and Chen, J. (2018). Generic model-agnostic convolutional neural network for single image dehazing. *arXiv preprint arXiv:1810.02862*.

[79] Lu, Y., Sarkis, M., and Lu, G. (2020). Multi-task learning for single image depth estimation and segmentation based on unsupervised network. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10788–10794.

[80] Matsumoto, R. (2018). Introducing the perception-distortion tradeoff into the rate-distortion theory of general information sources. *IEICE Communications Express*, **7**(11), 427–431.

[81] Matsumoto, R. (2019). Rate-distortion-perception tradeoff of variable-length source coding for general information sources. *IEICE Communications Express*, **8**(2), 38–42.

[82] Menon, S., Damian, A., Hu, S., Ravi, N., and Rudin, C. (2020a). Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 2437–2445.

[83] Menon, S., Damian, A., Hu, S., Ravi, N., and Rudin, C. (2020b). Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 2437–2445.

[84] Mentzer, F., Agustsson, E., Tschannen, M., Timofte, R., and Van Gool, L. (2018). Conditional probability models for deep image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4394–4402.

[85] Mentzer, F., Toderici, G. D., Tschannen, M., and Agustsson, E. (2020). High-fidelity generative image compression. In *Advances in Neural Information Processing Systems*, volume 33.

[86] Mirza, M. and Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

[87] Nah, S., Hyun Kim, T., and Lee, K. M. (2017). Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 257–265. IEEE.

[88] Nam, H. and Han, B. (2016). Learning multi-domain convolutional neural networks for visual tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4293–4302.

[89] Narasimhan, S. G. and Nayar, S. K. (2002). Vision and the atmosphere. *International Journal of Computer Vision*, **48**(3), 233–254.

[90] Nath Kundu, J., Krishna Uppala, P., Pahuja, A., and Venkatesh Babu, R. (2018). Adadepth: Unsupervised content congruent adaptation for depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2656–2665.

[91] Nathan Silberman, Derek Hoiem, P. K. and Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In *ECCV*.

[92] Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning.

[93] Pan, X., Zhan, X., Dai, B., Lin, D., Loy, C. C., and Luo, P. (2021). Exploiting deep generative prior for versatile image restoration and manipulation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[94] Pang, Y., Nie, J., Xie, J., Han, J., and Li, X. (2020). Bidnet: Binocular image dehazing without explicit disparity estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5930–5939, Los Alamitos, CA, USA. IEEE Computer Society.

[95] Park, S., Yoo, J., Cho, D., Kim, J., and Kim, T. H. (2020a). Fast adaptation to super-resolution networks via meta-learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pages 754–769. Springer.

[96] Park, T., Efros, A. A., Zhang, R., and Zhu, J.-Y. (2020b). Contrastive learning for unpaired image-to-image translation. In *European Conference on Computer Vision*, pages 319–345. Springer.

[97] Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. (2017). Automatic differentiation in pytorch. In *NIPS Workshop*.

[98] Peyré, G. and Cuturi, M. (2019). Computational optimal transport. *Foundations and Trends in Machine Learning*, **11**(5-6).

[99] Pilzer, A., Xu, D., Puscas, M., Ricci, E., and Sebe, N. (2018). Unsupervised adversarial depth estimation using cycled generative networks. In *2018 International Conference on 3D Vision (3DV)*, pages 587–595. IEEE.

[100] Pilzer, A., Lathuiliere, S., Sebe, N., and Ricci, E. (2019). Refine and distill: Exploiting cycle-inconsistency and knowledge distillation for unsupervised monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9768–9777.

[101] Poggi, M., Tosi, F., and Mattoccia, S. (2018). Learning monocular depth estimation with unsupervised trinocular assumptions. In *2018 International Conference on 3D Vision (3DV)*, pages 324–333. IEEE.

[102] Qin, X., Wang, Z., Bai, Y., Xie, X., and Jia, H. (2020). Ffa-net: Feature fusion attention network for single image dehazing. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**(07), 11908–11915.

[103] Qu, Y., Chen, Y., Huang, J., and Xie, Y. (2019). Enhanced pix2pix dehazing network. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8152–8160.

[104] Ramamonjisoa, M. and Lepetit, V. (2019). Sharpnet: Fast and accurate recovery of occluding contours in monocular depth estimation. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 0–0.

[105] Rebuffi, S.-A., Bilen, H., and Vedaldi, A. (2017). Learning multiple visual domains with residual adapters. *arXiv preprint iclrarXiv:1705.08045*.

[106] Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **39**(6), 1137–1149.

[107] Ren, W., Liu, S., Zhang, H., Pan, J., Cao, X., and Yang, M.-H. (2016). Single image dehazing via multi-scale convolutional neural networks. In *European conference on computer vision*, pages 154–169. Springer.

[108] Ren, W., Ma, L., Zhang, J., Pan, J., Cao, X., Liu, W., and Yang, M.-H. (2018). Gated fusion network for single image dehazing. In *IEEE Conference on Computer Vision and Pattern Recognition*.

[109] Rippel, O. and Bourdev, L. (2017). Real-time adaptive image compression. In *International Conference on Machine Learning*, pages 2922–2930.

[110] Roy, A. and Todorovic, S. (2016). Monocular depth estimation using neural regression forest. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5506–5514.

[111] Sahu, G. and Seal, A. (2019). Image dehazing based on luminance stretching. In *2019 International Conference on Information Technology (ICIT)*, pages 388–393. IEEE.

[112] Sahu, G., Seal, A., Krejcar, O., and Yazidi, A. (2021). Single image dehazing using a new color channel. *Journal of Visual Communication and Image Representation*, **74**, 103008.

[113] Saldi, N., Linder, T., and Yüksel, S. (2015a). Output constrained lossy source coding with limited common randomness. *IEEE Transactions on Information Theory*, **61**(9), 4984–4998.

[114] Saldi, N., Linder, T., and Yüksel, S. (2015b). Randomized quantization and

source coding with constrained output distribution. *IEEE Transactions on Information Theory*, **61**(1), 91–106.

[115] Scharstein, D. and Szeliski, R. (2003). High-accuracy stereo depth maps using structured light. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 1, pages I–I. IEEE.

[116] Schoenauer-Sebag, A., Heinrich, L., Schoenauer, M., Sebag, M., Wu, L. F., and Altschuler, S. J. (2019). Multi-domain adversarial learning. *arXiv preprint arXiv:1903.09239*.

[117] Shao, Y., Li, L., Ren, W., Gao, C., and Sang, N. (2020). Domain adaptation for image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2808–2817.

[118] Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A. P., Bishop, R., Rueckert, D., and Wang, Z. (2016). Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883.

[119] Sicilia, A., Zhao, X., Minhas, D. S., O'Connor, E. E., Aizenstein, H. J., Klunk, W. E., Tudorascu, D. L., and Hwang, S. J. (2021). Multi-domain learning by meta-learning: Taking optimal steps in multi-domain loss landscapes by inner-loop learning. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 650–654. IEEE.

[120] Soh, J. W., Cho, S., and Cho, N. I. (2020). Meta-transfer learning for zero-shot

super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3516–3525.

[121] Song, T., Kim, Y., Oh, C., and Sohn, K. (2018). Deep network for simultaneous stereo matching and dehazing. In *BMVC*, page 5.

[122] Sun, Y., Wang, X., Liu, Z., Miller, J., Efros, A., and Hardt, M. (2020). Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pages 9229–9248. PMLR.

[123] Tai, Y., Yang, J., Liu, X., and Xu, C. (2017). MemNet: A Persistent Memory Network for Image Restoration. In *The Conference on Computer Vision and Pattern Recognition*, pages 4539–4547.

[124] Tai, Y.-W., Liu, S., Brown, M. S., and Lin, S. (2010). Super resolution using edge prior and single image detail synthesis. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 2400–2407. IEEE.

[125] Tateno, K., Tombari, F., Laina, I., and Navab, N. (2017). Cnn-slam: Real-time dense monocular slam with learned depth prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6243–6252.

[126] Theis, L. and Agustsson, E. (2021). On the advantages of stochastic encoders. *arXiv preprint arXiv:2102.09270*.

[127] Theis, L. and Wagner, A. B. (2021). A coding theorem for the rate-distortion-perception function. *arXiv preprint arXiv:2104.13662*.

[128] Tschannen, M., Agustsson, E., and Lucic, M. (2018). Deep generative models

for distribution-preserving lossy compression. In *Advances in Neural Information Processing Systems*, pages 5929–5940.

[129] Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). Deep image prior. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9446–9454.

[130] Villani, C. (2009). *Optimal transport: old and new*, volume 338. Springer.

[131] Wang, P., Shen, X., Lin, Z., Cohen, S., Price, B., and Yuille, A. L. (2015). Towards unified depth and semantic prediction from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2800–2809.

[132] Wang, S., Stavrou, P. A., and Skoglund, M. (2021a). Generalized talagrand inequality for sinkhorn distance using entropy power inequality. *arXiv preprint arXiv:2109.08430*.

[133] Wang, W., Yuan, X., Wu, X., and Liu, Y. (2017). Fast image dehazing method based on linear transformation. *IEEE Transactions on Multimedia*, **19**(6), 1142–1155.

[134] Wang, W., Wen, F., Yan, Z., Ying, R., and Liu, P. (2021b). Optimal transport for unsupervised restoration learning. *arXiv preprint arXiv:2108.02574*.

[135] Wang, X., Cai, Z., Gao, D., and Vasconcelos, N. (2019). Towards universal object detection by domain attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7289–7298.

[136] Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli, E. P., *et al.* (2004). Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, **13**(4), 600–612.

[137] Wong, A. and Soatto, S. (2019). Bilateral cyclic constraint and adaptive regularization for unsupervised monocular depth prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5644–5653.

[138] Wu, H., Liu, J., Xie, Y., Qu, Y., and Ma, L. (2020). Knowledge transfer dehazing network for nonhomogeneous dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*.

[139] Wu, H., Qu, Y., Lin, S., Zhou, J., Qiao, R., Zhang, Z., Xie, Y., and Ma, L. (2021). Contrastive learning for compact single image dehazing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10551–10560.

[140] Wu, Y., Ying, S., and Zheng, L. (2018). Size-to-depth: A new perspective for single image depth estimation. *arXiv preprint arXiv:1801.04461*.

[141] Xu, D., Ricci, E., Ouyang, W., Wang, X., and Sebe, N. (2017). Multi-scale continuous crfs as sequential deep networks for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5354–5362.

[142] Xu, D., Wang, W., Tang, H., Liu, H., Sebe, N., and Ricci, E. (2018). Structured attention guided convolutional neural fields for monocular depth estimation. In

Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3917–3925.

[143] Yair, N. and Michaeli, T. (2018). Multi-Scale Weighted Nuclear Norm Image Restoration. *CVPR*.

[144] Yan, Z., Wen, F., Ying, R., Ma, C., and Liu, P. (2021). On perceptual lossy compression: The cost of perceptual reconstruction and an optimal training framework. In *International Conference on Machine Learning*.

[145] Yang, D. and Sun, J. (2018). Proximal dehaze-net: A prior learning-based deep network for single image dehazing. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 702–717.

[146] Yang, Y. and Hospedales, T. M. (2014). A unified perspective on multi-domain and multi-task learning. *arXiv preprint arXiv:1412.7489*.

[147] Yu, Y., Liu, H., Fu, M., Chen, J., Wang, X., and Wang, K. (2021). A two-branch neural network for non-homogeneous dehazing via ensemble learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 193–202.

[148] Zhang, G., Qian, J., Chen, J., and Khisti, A. (2021). Universal rate-distortion-perception representations for lossy compression. *arXiv preprint arXiv:2106.10311*.

[149] Zhang, H. and Patel, V. M. (2018). Densely connected pyramid dehazing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3194–3203.

[150] Zhang, J. and Tao, D. (2020). Famed-net: A fast and accurate multi-scale end-to-end dehazing network. *IEEE Transactions on Image Processing*, **29**, 72–84.

[151] Zhang, K., Zuo, W., Gu, S., and Zhang, L. (2017). Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938.

[152] Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., and Fu, Y. (2018a). Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301.

[153] Zhang, Y., Tian, Y., Kong, Y., Zhong, B., and Fu, Y. (2018b). Residual dense network for image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[154] Zhang, Z., Cui, Z., Xu, C., Jie, Z., Li, X., and Yang, J. (2018c). Joint task-recursive learning for semantic segmentation and depth estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 235–251.

[155] Zhao, H., Gallo, O., Frosio, I., and Kautz, J. (2017). Loss functions for image restoration with neural networks. *IEEE Transactions on Computational Imaging*, **3**(1), 47–57.

[156] Zhao, S., Fu, H., Gong, M., and Tao, D. (2019). Geometry-aware symmetric domain adaptation for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9788–9798.

[157] Zhou, C., Zhang, H., Shen, X., and Jia, J. (2017a). Unsupervised learning of

stereo matching. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1567–1575.

[158] Zhou, L. and Kaess, M. (2020). Windowed bundle adjustment framework for unsupervised learning of monocular depth estimation with u-net extension and clip loss. *IEEE Robotics and Automation Letters*, **5**(2), 3283–3290.

[159] Zhou, T., Brown, M., Snavely, N., and Lowe, D. G. (2017b). Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1858.

[160] Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232.

[161] Zhu, Q., Mai, J., and Shao, L. (2015). A fast single image haze removal algorithm using color attenuation prior. *IEEE transactions on image processing*, **24**(11), 3522–3533.

[162] Ziv, J. (1985). On universal quantization. *IEEE Transactions on Information Theory*, **31**(3), 344–347.

[163] Zoran, D., Isola, P., Krishnan, D., and Freeman, W. T. (2015). Learning ordinal relationships for mid-level vision. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 388–396.

[164] Zou, Y., Luo, Z., and Huang, J.-B. (2018). Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 36–53.