AN ANALYSIS ON THE EFFECT OF DYNAMIC RANGE ON OBJECT DETECTION WITH DEEP NEURAL NETWORKS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

İSMAİL HAKKI KOÇDEMİR

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

SEPTEMBER 2021

Approval of the thesis:

AN ANALYSIS ON THE EFFECT OF DYNAMIC RANGE ON OBJECT DETECTION WITH DEEP NEURAL NETWORKS

submitted by **İSMAİL HAKKI KOÇDEMİR** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Mehmet Halit S. Oğuztüzün
Head of Department, **Computer Engineering** _____

Assoc. Prof. Dr. Sinan Kalkan
Supervisor, **Computer Engineering, METU** _____

Prof. Dr. A. Aydın Alatan
Co-supervisor, **Electrical-Electronics Engineering, METU** _____

Examining Committee Members:

Prof. Dr. Ahmet Oğuz Akyüz
Computer Engineering, METU _____

Assoc. Prof. Dr. Sinan Kalkan
Computer Engineering, METU _____

Assoc. Prof. Dr. Nazlı İkizler-Cinbiş
Computer Engineering, Hacettepe University _____

Assoc. Prof. Dr. Hacer Yalım Keleş
Computer Engineering, Hacettepe University _____

Assist. Prof. Dr. Emre Akbaş
Computer Engineering, METU _____

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: İsmail Hakkı Koçdemir

Signature :

ABSTRACT

AN ANALYSIS ON THE EFFECT OF DYNAMIC RANGE ON OBJECT DETECTION WITH DEEP NEURAL NETWORKS

Koçdemir, İsmail Hakkı
M.S., Department of Computer Engineering
Supervisor: Assoc. Prof. Dr. Sinan Kalkan
Co-Supervisor: Prof. Dr. A. Aydın Alatan

September 2021, 60 pages

An important problem in computer vision, particularly in object detection, is being able to perceive objects even under challenging illumination conditions. Being robust to such conditions is especially important in applications, such as autonomous driving. Despite the significance of the problem, existing autonomous driving systems use deep object detection networks with low-dynamic range (LDR) images during both the training phase and the testing phase. In this thesis, we investigate whether high-dynamic range (HDR) images can provide better performance for object detection in autonomous driving systems. For this purpose, we provide a comprehensive analysis of the effect of dynamic range on object detection performance. We compare LDR and HDR images on different illumination conditions and show that HDR performs on par with to LDR counterparts when used without pre-processing including normalization and gamma correction. We also show that after applying this certain pre-processing operations, HDR is able achieve on par detection performance with tone-mapped LDR. Moreover, we propose a novel framework to jointly optimize deep-learning-based tone-mapping operators and object detection networks by using

a Generative Adversarial approach. Our architecture achieves the best tone-mapping quality score while maintaining a competitive performance to the best classical tone-mapping operator in terms of detection performance.

Keywords: Object Detection, High Dynamic Range, Low Dynamic Range, Tone-Mapping, Generative Adversarial Networks

ÖZ

DİNAMİK ARALIĞIN DERİN SİNİR AĞLARI İLE NESNE TESPİTİNDEKİ ETKİSİNİN ANALİZİ

Koçdemir, İsmail Hakkı
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. Sinan Kalkan
Ortak Tez Yöneticisi: Prof. Dr. A. Aydın Alatan

Eylül 2021 , 60 sayfa

Bilgisayarlı görü için, özellikle nesne tespiti için, zorlu ışık koşullarında etraftaki nesnelerin algılanabilmesi önemli bir problem teşkil etmektedir. Özellikle sürücüsüz araçlar gibi nesne tespiti algoritmalarının uygulama alanlarının çoğu, bu koşullara karşı dayanıklı olmayı gerektirmektedir. Fakat, bu probleme karşın var olan otonom sürüş sistemleri nesne tespiti için düşük dinamik aralıklı imgelerde eğitilmekte ve test edilmektedirler. Bu tezde, yüksek dinamik aralıklı (YDA) imgelerin daha iyi nesne tespiti performansı sağlayıp sağlamadığı analiz ediyoruz. Bu doğrultuda, dinamik aralığın nesne tespiti performansına etkisinin kapsamlı analizini vermekteyiz. YDA imgeler ile Düşük Dinamik Aralıklı (DDA) imgeleri farklı aydınlanma koşullarını dikkate alan yeni performans kategorilerinde değerlendirmekteyiz. Bu değerlendirmlere sonucunda, normalizasyon ve gama düzeltme gibi ön işleme yöntemleri kullanılmadığı takdirde, YDA imgeler DDA imgelere göre daha kötü nesne tespiti performansı gösterdiği görülmektedir. Ayrıca, bahsedilen ön işleme yöntemleri kullanıldığında YDA imgeler en iyi performans veren ton*dönAn Ek olarak, derin öğrenme tabanlı ton-

dönüşümü operatörlerinin ve nesne tespiti yapan derin ağların ortak optimizasyonu için bir mimari önermekteyiz. Önerdiğimiz mimari, en iyi ton dönüşümü kalite skorlarını elde ederken, nesne tespiti performansında da en başarılı ton-dönüşümü ile yarışır skor elde edebilmektedir.

Anahtar Kelimeler: Nesne Tespiti, Yüksek Dinamik Aralık, Düşük Dinamik Aralık, Ton-dönüşümü, Üretici Çekişmeli Ağlar

To my family

ACKNOWLEDGMENTS

I would like to thank everyone who supported me throughout my thesis journey helping me keep going in good times and hard times:

To my family; my mom and dad, my sister and brother. There are no words to describe how supportive, loving and caring family they are. None of it could have been possible without them always standing with me and encouraging me in every step of the way.

To my supervisor, Sinan Kalkan. He was such a good supervisor in that he always pointed me to the right direction and cared about helping me with the issues I encountered along the way even if It was not related to my studies.

To my co-supervisor, Aydın Alatan. He was always very helpful and undersandıng with my progress. He gave countless invaluable advice that made this thesis possible.

To my project team members, Prof. Ahmet Oguz Akyuz, Dr. Alper Koz, and Prof. Alan Chalmers for their very helpful and valuable comments and discussions.

To my friends; Samet, Artun, Hazan and Erdi. We went trough this process together and it was them who made it more enjoyable, easier and fun.

Finally, We'd like to thank United Kingdom Royal Academy of Engineering for funding the project "H3DR Camera: Smart Embedded Camera with Multi-exposure & Multi-view Capability for Autonomous Vehicles". Additionally, the numerical calculations reported in this thesis were partially performed at TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources).

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

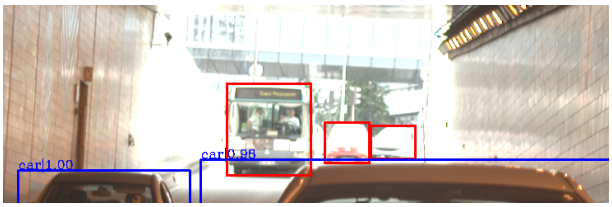
LIST OF ABBREVIATIONS

HDR	High Dynamic Range
LDR	Low Dynamic Range
GAN	Generative Adversarial Network
TMO	Tone-mapping Operator
CNN	Convolutional Neural Network
R-CNN	Region-based Convolutional Neural Network
ANN	Artificial Neural Network
TMQI	Tone-Mapping Quality Index
HDRVDP	High Dynamic Range Visual Difference Predictor
CIVDM	Contrast Invariant Visual Difference Metric

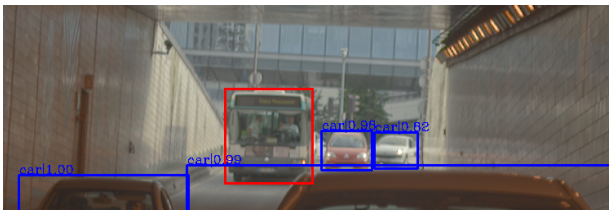
CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition



(a) LDR image.



(b) Tone-mapped HDR image.

Figure 1.1: (a) LDR images make it difficult for detection of objects (shown in blue boxes) in adverse illumination conditions e.g. in tunnels. (b) This can be alleviated using HDR images or tone-mapped LDR images. Blue: detected objects. Red: Missed objects. (Input image: from the "tunnel video clip" of the EU NEVEx Project. Tone-mapping: Durand et al. [4])

One of most-studied problems in computer vision is object detection. It occupies a central spot in the community's interest and has seen significant amount of progress in the recent years. This progress has been mostly due to the realization that the artificial neural networks (ANNs), and Convolutional Neural Networks (CNNs) in particular, can solve object detection and other relevant tasks better than hand-crafted methods

given enough computing power and data [9].

One of the key applications of object detection algorithms is autonomous driving. In autonomous driving, a vehicle needs to be able to locate and classify every object around it, through the images coming from its sensors/cameras, then plan its movement in real time in a robust way [10]. This heavily depends on how good and robust the performance of perception (incl. object detection) is.

A robust object detection algorithm should be able to preserve its accuracy in challenging environments, such as different weather conditions (rainy, foggy) or adverse lighting. Especially, lighting conditions such as direct sunlight hitting the sensor, or just before the exit of a dark tunnel makes the detection task much more challenging, as can be seen in Figure 1.1. In such scenarios, a camera's ability to capture the lightest and brightest areas in a scene without losing any detail, i.e. the the dynamic range of the camera, makes a significant difference [11]. The problem usually arises from the fact that cameras with low dynamic range (LDR) struggle with such conditions, whereas a human can quickly adapt to different brightness and perceive objects in both dark and bright areas in the same scene thanks to the adaptation and wider dynamic range capabilities of the human eye [12].

Modern cameras with high dynamic range (HDR) capabilities can also capture the details in a scene with extremely bright and quite dark regions. Although scenario is a crucial issue, utilization of HDR cameras for challenging lighting conditions in object detection, and more importantly in autonomous driving, has not been studied extensively.

To this end, in this thesis, the following research problems are studied:

- Do deep object detectors perform better with LDR or HDR images?
- Which tone-mapping operators are more suitable for object detectors?
- Can we perform tone mapping in such a way that the performance of an object detector is maximized, while simultaneously maintaining a perceptual quality for the tone-mapped images?

1.2 Contributions and Novelties

In this thesis, we propose the following main contributions:

- A comparative analysis of HDR, standard LDR and tone-mapped LDR images in object detection. Our analysis reveals that, while raw HDR performs slightly worse on the average compared to standard LDR and tone-mapped LDR. We also observe that HDR needs pre-processing operations such as normalization and gamma correction before using with modern object detection networks to achieve similar performance to the best performing tone-mapped LDR images. Finally, we observe that when the detection network is pre-trained with LDR images before HDR training, the gap in detection performance between HDR and LDR widens.

- A modified version of the object detection performance metric (mean average precision) for different illumination conditions (luminance, dynamic range and entropy) inside object bounding boxes, (similar to the calculation of the performance for different object area intervals).

- A novel methodology for training a deep learning-based tone-mapping operator and an object detection network jointly. We analyse different ways to combine the feedback from tone-mapping quality and object detection quality during training (e.g. different weights for each objective). We provide a comparative analysis of the proposed methodology against both the hand-crafted tone-mapping operators and also deep-learning based tone-mapping operators, in terms of object detection quality and image/HDR quality. By fine-tuning the contributions from tone-mapping and object detection objective, we are able to achieve a tone-mapping operator which achieves significantly higher tone-mapping quality scores and also slightly improves the detection performance.

1.3 The Outline of the Thesis

This thesis is comprised of five chapters.

In the first chapter (Introduction), the problem to be addressed and contributions are declared in brief and simple terms.

In Chapter 2 (Background), required theoretical background is given for the problem domain, which are HDR imaging and Object detection; and also the existing algorithms for solving these problems, which are generative adversarial networks, object detection and tone-mapping algorithms.

In the next chapter (Does Wider Dynamic Range Improve Object Detection?), an analysis of the effect of the dynamic range on object detection performance is provided.

In Chapter 4 (Joint Optimization of Tone-mapping and Object Detection Algorithms), a training methodology is introduced for joint optimization of tone-mapping and object detection objectives. Additionally, The proposed method is compared against different methods and baselines.

In the final chapter (Conclusion), the objective of the thesis and the contributions are summarised.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Generative Adversarial Networks

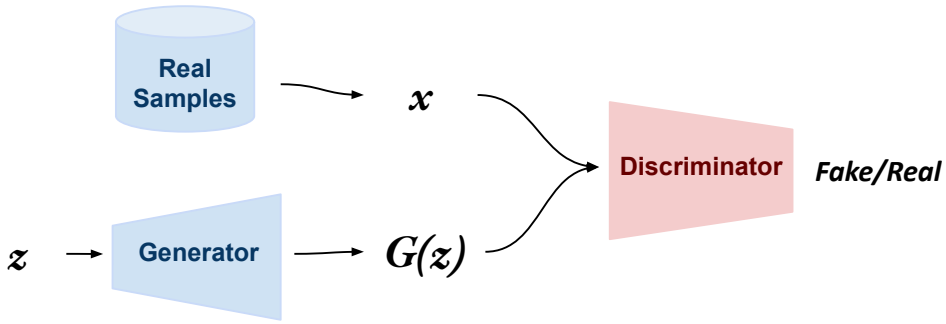


Figure 2.1: Overall architecture of Generative Adversarial Networks.

Generative Adversarial Networks (GANs), originally proposed by [13], are a framework for training deep neural networks for drawing samples from highly complex distributions (e.g. dog images). They are based on an adversarial min-max game between two components, namely the discriminator D and the generator G, which can be seen in Figure 2.1. On one hand, the generator is trained to map a sample z from an easy-to-sample distribution p_z (e.g. Multivariate Gaussian), which is also called the latent distribution, to the target distribution p_{data}. On the other hand, the discriminator is trained to discriminate between the real samples x from p_{data} and the samples obtained by the generator.

Although there have been many variations proposed since the invention of the GANs [14, 15], the original objective function for the each of the components, the generator

and the discriminator, is as follows:

$$\mathcal{L}_{GAN}(G, D) = \mathbb{E}_{x \sim p_{data}(x)}[\log\ D(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z))], \tag{2.1}$$

and the optimal generator is obtained by:

$$G^* \leftarrow \min_G \max_D \mathcal{L}_{GAN}, \tag{2.2}$$

where the discriminator's output $D(\cdot)$ defines the probability of the sample being real. Apart from the novel variations [16, 17], training is traditionally performed using Stochastic Gradient Descent (SGD), where each of the components is trained for a single SGD step in an alternating fashion.

There exist well-known problems in the training of GANs, such as vanishing gradients, mode collapse and general convergence problems due to the instability of the min-max objective. Vanishing gradient occurs when the discriminator becomes quite strong, to the point that it cannot provide useful feedback for the generator, which in turn causes the sizes of updates to the generator tend to zero. On the other hand, mode collapse occurs when generator learns to output only a single mode of the desired distribution to fool the discriminator more easily. Although over time the discriminator learns to discriminate in this single mode, the generator then usually jumps to another mode. Consequently, the generator ends up looping over the modes of the distribution.

There are several works addressing the vanishing gradient [14, 13] and the mode collapse [14, 16] problems by various strategies including novel objectives and regularization techniques. Additionally, convergence properties and possible remedies also have been studied extensively since the invention of GANs [18, 19, 20].

2.1.1 Conditional Generative Adversarial Networks

Conditional GAN [21] modifies the original GAN objective in Equation 2.1 by introducing a new variable on which G and D conditions their mapping. This is achieved with the following objective:

$$\mathcal{L}_{CGAN}(G, D) = \mathbb{E}_{x \sim p_{data}(x)}[\log D(x, y)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z, y), y)], \tag{2.3}$$

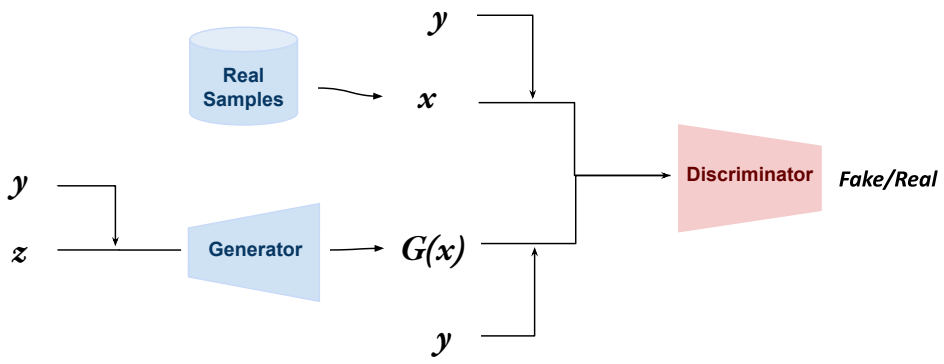


Figure 2.2: Overall architecture of Conditional Generative Adversarial Networks.

where y is the additional variable on which generation and discrimination are conditioned. In practice, one way to integrate the additional variable y is to simply concatenate it to the inputs of D and G, namely z and x, respectively [22].

2.1.2 Least Squares Generative Adversarial Networks

Least Squares GAN [15] is one of the solutions proposed as a remedy for the vanishing gradient problem. It is claimed to improve the quality of the generated samples, while also stabilizing the training process. The objective of Least Squares GAN is given as follows:

$$\begin{aligned} \mathcal{L}_{LSGAN}(D) &= \frac{1}{2}\mathbb{E}_{x\sim p_{data}(x)}[(D(x)-b)^2] + \frac{1}{2}\mathbb{E}_{x\sim p_z(z)}[(D(G(z))-a)^2] \\ \mathcal{L}_{LSGAN}(G) &= \frac{1}{2}\mathbb{E}_{x\sim p_z(z)}[(D(G(z))-c)^2], \end{aligned} \tag{2.4}$$

where x is the input image. Additionally; a, b, and c need to satisfy the conditions of $b-c=1$ and $b-a=2$. This is required so that the minimization of Equation 2.4 corresponds to the minimization of Pearson χ^2 divergence between $p_{data}+p_g$ and $2p_g$, where p_g denotes the distribution of generated samples output by G.

2.2 Object Detection

Computer vision problems have seen significant improvements over the hand-crafted methods thanks to deep learning approaches. Object detectors mainly solve two tasks:

localization and classification [9]. While the former is solved by regressing bounding box coordinates around the object in the image, the latter is addressed by classifying (identifying) the category of the localized object. A high-level objective for an object detection network can be generally defined as follows:

$$\mathcal{L}_{Det} = \mathcal{L}_{cls}(p, p^*) + \lambda \mathcal{L}_{loc}(t, t^*), \tag{2.5}$$

where x is the input image, p^* and t^* are the ground-truth category and location of the target object, F_θ is the detection network parametrized by θ, p and t are category and the location of the detected object output by F_θ and λ is the weighting parameter for the localization objective. For the classification objective, the cross entropy loss is commonly used:

$$CE(p, p^*) = -\sum_i p_i^* \log(p_i), \tag{2.6}$$

where p_i's, which is output by F_θ, are the estimated probabilities for each category, and p_i^*'s are the ground-truth probabilities. For the localization objective, generally, smooth L_1 is used [23] between the coordinates output by the network t and ground truth coordinates t^*:

$$\mathcal{L}_{loc}(t, t^*) = \sum_{i \in \{x,y,w,h\}} smooth_{L_1}(t_i - t_i^*), \tag{2.7}$$

where x, y, w and h are the coordinates defining row index, column index, width and height of the estimated bounding box t and ground truth bounding box $t*$, respectively.

Approaches on how to design F_θ using deep networks can be considered under two categories: namely one-stage and two-stage detection.

2.2.1 Two-stage Object Detection Networks

Two-stage detectors have two distinct parts; one for region (object) proposal generation and another stage for classification of region proposals into objects as shown in Figure 2.3. The region proposal stage is generally a convolutional neural network that first extracts features and using these features, produces coordinates for regions

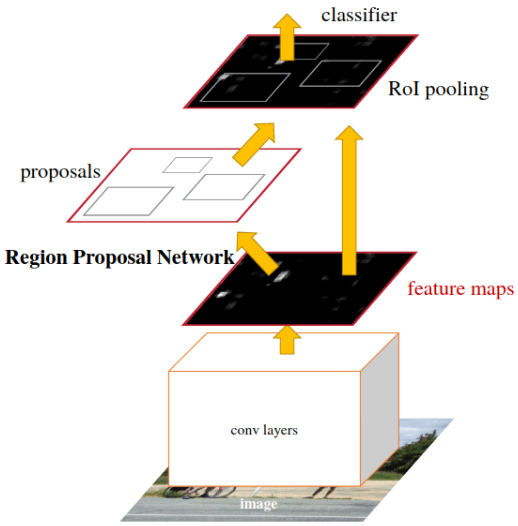


Figure 2.3: Overall architecture of Faster R-CNN taken from [5]. Faster R-CNN consists of a Region Proposal Layer, followed by classification and regression heads. The Region of Interest Pooling layer extracts the features for a region proposal.

possibly containing objects together with how confident it is that the coordinates contain an object. The second stage aims to classify the proposed regions and find tighter bounding boxes around the objects.

R-CNN [24], which stands for Region-based Convolutional Neural Networks, is the seminal work for two-stage detectors. Later, it was extended by Fast R-CNN [25], Faster R-CNN [5] and with many features ranging from better feature extractors (ResNet [26], MobileNets [27], HRnet [28]) to design changes (Cascade R-CNN [29], Feature Pyramids [7] etc.).

2.2.1.1 One-stage Detection Networks

One-stage detectors unify the two stages in a single architecture. Unlike two-stage detectors that first extract regions then operate on the a narrower set of extracted regions; one stage-detection directly produces the detection output in densely sampled locations in the image with a single pass. YOLO [30] and SSD [31] were the first well-known examples of one-stage detectors, with the difference being how the prediction is made on multi-scale features. However, making predictions over densely

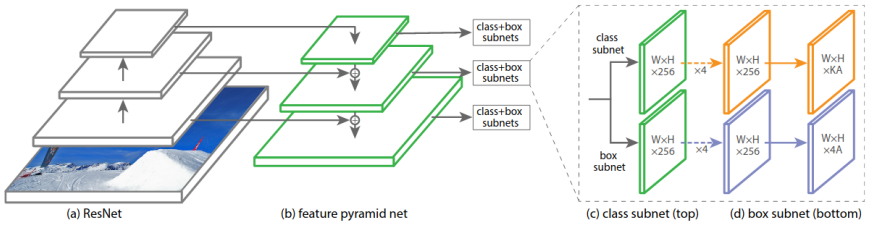


Figure 2.4: Overall architecture of RetinaNet taken from [6]. Backbone (a) is extended with convolutional feature pyramid [7] (b). It is followed by two subnetworks, one for classifying anchor boxes (c) and one for regressing from anchor boxes to object boxes (d).

sampled locations turned out to cause imbalance problems due to the fact that many locations did not contain objects, which overwhelmed the loss with background samples. As a solution to this problem, RetinaNet [6], whose architecture is shown in Figure 2.4, introduced so-called focal loss for the single-stage detectors which is given by the following formula:

$$FocalLoss(p, p^*) = -(1 - p_c)^{\gamma} \log(p_c), \tag{2.8}$$

where p_c is the probability for the correct category. This is shown to help reducing the effect of imbalance problems both in foreground-background and foreground-foreground classes when $\gamma > 0$, which that puts more emphasise on the mis-classified examples [6].

2.3 High Dynamic Range (HDR) Imaging

Dynamic range, although its definition may vary for different domains, is generally defined to be the ratio of smallest and largest value that a certain quantity can take [32]. Particularly, in the imaging domain, luminance is the quantity for which we calculate the dynamic range.

In order to form an HDR image that captures the full range of luminance values that exist in a real-life scene, there exist two main approaches [32]. The first approach is to design sensors that have extended abilities to capture wider dynamic range. In the second approach, one achieves an HDR image by merging multiple LDR images that

are captured with different exposure levels [33, 34, 35].

The dynamic range of a captured scene can be calculated as follows:

$$L = \log_2(\gamma_{peak}) - \log_2(\gamma_{noise}), \tag{2.9}$$

where L is called the *log exposure range* which captures the ratio between the darkest and brightest regions in a scene.

2.3.1 Tone Mapping Operators

Most monitors do not have the ability to display the range of luminance values in an HDR image [12]. The purpose of tone-mapping operators (TMOs) is to map the luminance values of an HDR image into the range of a standard monitor that can only represent LDR scenes, while also keeping the perceptual information (such as color and contrast) intact as much as possible [36]. In the rest of this section, two different approaches for TMOs are explained.

2.3.1.1 Classical Tone Mapping Operators

Classical TMOs rely on hand-crafted algorithms for compressing the dynamic range in an HDR image. These algorithms can be divided into two categories: *global* and *local* [37]. Global TMOs [38, 39, 40, 41] apply the same mapping to every region of the image. Local TMOs, however, use adaptive mapping depending on the statistics of the local neighborhood of a pixel [42, 4, 43, 41, 44]. On top of these broad categories, other methods [45, 46, 47, 48] have been proposed, mimicking the related aspects of the human visual system so that the final mapping would be perceptually more natural [37].

2.3.1.2 Deep Learning-based Tone Mapping Operators

Tone mapping operators have been no exemption from the areas that have benefited from the success of the deep learning algorithms. In the recent years, several works have proposed using generative models (most commonly GANs) for learning tone

mapping from the data [1, 49, 50, 51]. Approaches introduced in these works mostly rely on the image-to-image translation algorithms that are based on conditional generative adversarial networks [22]. A baseline formulation of these approaches can be defined within the conditional GAN framework as follows:

$$\mathcal{L}_{CGAN}(G, D) = \mathbb{E}_{x\sim p_{data}(x)}[\log D(x, T(x))] + \mathbb{E}_{x\sim p_x(data)}[\log(1 - D(x, G(x))], \tag{2.10}$$

where x is the HDR image, and T is the classical tone mapping operator that produces the ground-truth tone-mapped image $T(x)$. $G(x)$ is the image produced by the generator which tries to mimic the behaviour of the classical tone-mapping operator. Finally, D tries to discriminate between the images tone-mapped by T and G.

Rana et al. [1] improve the baseline objective given in Equation 2.10 by adding feature matching loss which penalizes the generator with the discrepancy between representations of the generated image and the ground-truth image in the intermediate layers of D:

$$\mathcal{L}_{FM}(G, D) = \sum_i \frac{1}{N_i} ||D^{(i)}(x, T(x)) - D^{(i)}(x, G(x))||_i, \tag{2.11}$$

where index i represents the activations in the i^{th} layer of the D and N_i represents the number of elements in the i^{th} layer. Furthermore, they add a so-called perceptual loss which similarly penalizes the discrepancy between the representations of generated and ground-truth images in the intermediate layers of a pre-trained feature extractor of an object recognition network as follows:

$$\mathcal{L}_{PRP}(G) = \sum_i \frac{1}{N_i} ||F_\Phi^{(i)}(T(x)) - F_\Phi^{(i)}(G(x))||_i, \tag{2.12}$$

where index i represents the activations in the i^{th} layer of the pretrained object recognition backbone F_Φ (e.g. a VGGNet [52] or ResNet [26]).

In addition to feature matching and perceptual losses, [50] claims further improvements by using the gradient profile loss which measures the similarity between gradient maps of the generated and the ground-truth image:

$$\mathcal{L}_{\mathcal{GPL}} = \sum_c \left(\frac{1}{H} trace(\nabla G(x) \nabla T(x)^\intercal) + \frac{1}{W} trace(\nabla G(x)^\intercal \nabla T(x)) \right), \tag{2.13}$$

where H and W represents height and width of the sample, $(\cdot)^\intercal$ represents transpose operation, and c represents the channel index.

Another approach that diverges from the aforementioned approaches is proposed by Su et al. [51], introducing a more structured design for the generator architecture where a global compression curve γ and edge preserving kernels are estimated explicitly with separate deep networks, instead of learning the parameters of a single generator that directly outputs the tone-mapped image in a more black-box manner.

2.4 Related Work

In this section, we present studies directly relevant for our work.

2.4.1 Object Detection/Recognition with HDR content

Scientific efforts on object detection/recognition that input HDR images are limited in the literature. One possible reason is the lack of a general purpose HDR detection dataset, e.g. at a scale similar to COCO [53] or Pascal VOC [54] datasets, which provided a significant boost for the deep object detection literature. For this reason, the few studies that perform object detection/recognition from HDR images use either a limited number of tone-mapped images [55] or synthetic data [56, 57]. Mukherjee et al. [55], for example, collect their own dataset and test well-known object detection networks on tone-mapped images. This study is limited in that the authors do not use HDR or tone-mapped LDR images for training the network. Instead they use a network pre-trained on LDR images and then test this network only on tone-mapped LDR images.

In a more related study, Mukherjee et al. [57] generate an HDR dataset from LDR images and train their network on the generated HDR dataset. Then they test their network on real-world HDR images and measure its performance on the subset of the images where the dynamic range is larger. This study is also limited in that it does not use real-world images for training the network but only for testing. Furthermore, the subset they use has limited size and does not consider the analysis of the different ranges of dynamic range spectrum such as lower or medium dynamic range scenes. Weiher [56], on the other hand, applies domain adaptation methods to a synthetic HDR dataset and train a semantic segmentation network on this adapted dataset. They

test the network with real-world LDR and HDR images to see whether the domain adaptation improves the performance. Due to the limited size of the synthetic dataset, they also pretrain the network on the COCO dataset.

2.4.2 Augmenting Downstream Vision Tasks with Generative (Adversarial) Networks

In this section we present studies which combine generative adversarial networks with downstream vision tasks, such as image classification and object detection. Usually, the approaches either primarily aim to improve the performance in the downstream task by the help of generative objective, or aim to generate better examples with the help of the downstream task.

Odena et al. [58] although do not train a separate network for the downstream task, their approach augments the GAN objective by adding a classification output to the discriminator. They also condition the generator on the class to be generated. Talebi et al. [8] propose to learn a image resizer model while also jointly trying to classify the given image, whose architecture diagram is given in Figure 2.5 leading to a resizing operation optimized for classification task. They claim to achieve better classification performance compared to classical resizing operations such as resizing with bilinear interpolation, while also introducing noise to the resized image which can be seen in Figure 2.6. Liu et al. [59] uses GANs to generate samples as a data augmentation for small object detection. The generator in the architecture tries to fool discriminator while also trying to maximize the detection loss of the detector. Simultaneously, discriminator tries to discriminate the fake samples and detector tries to detect object in the fake sample. Similarly, Rashid et al. [60] performs data augmentation with GANs for skin lesion classification. On a more related study, Vandenhende et al.[61] designs a three-player GAN for hard example generation, where the additional third player is a image classifier.

On the other hand, Liang et al. [59] integrates GANs to object detection architecture in order to boost the performance on small objects by trying to generate representations of small objects which are similar to that of large objects. Finally, Prakash et al. [62] propose a GAN-based detection framework where the discriminator tries

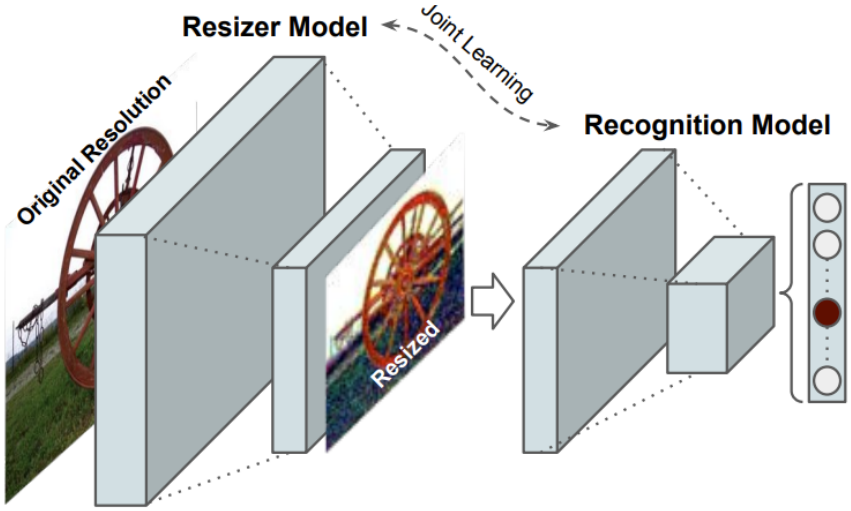


Figure 2.5: Overall architecture of the image resizer model. Figure taken from Talebi et al. [8].

to discriminate between the detection results coming from original and distorted versions of the same image. A summary of the methods combining generative tasks with downstream vision tasks is given in Table 2.1.

Table 2.1: Studies which propose to use generative networks for the augmentation of the downstream vision tasks.

Method	Generation Task	Downstream Task	Joint Training	Adversary
[58]	Data Generation	Classification	✓	Discriminator
[8]	Resizing	Classification	✓	-
[60]	Data Augmentation	Detection	X	Discriminator
[59]	Data Augmentation	Detection	✓	Discriminator, Detector
[61]	Data Augmentation	Classification	✓	Discriminator, Classifier
[63]	Feature Enhancement	Detection	✓	Discriminator
[62]	Detection	Detection	✓	Discriminator

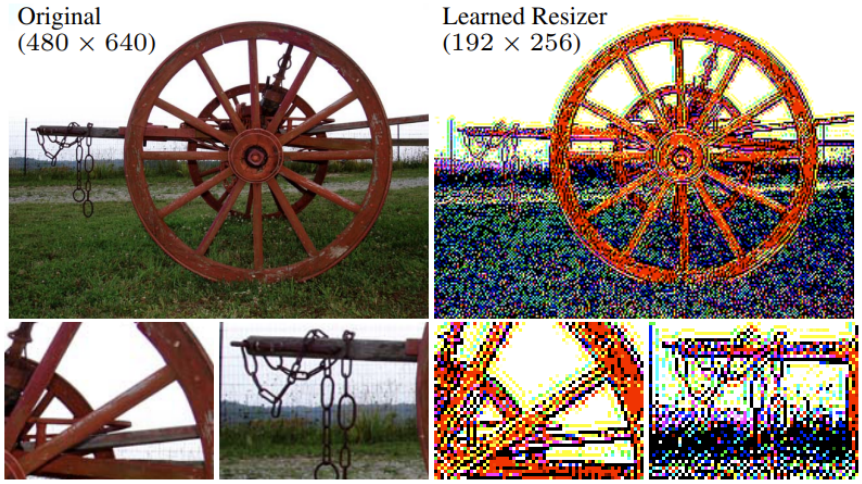


Figure 2.6: Sample images produced by the original resizer and the resizer. Figure taken from Talebi et al. [8].

CHAPTER 3

DOES HIGHER DYNAMIC RANGE IMPROVE OBJECT DETECTION?

In this chapter, we propose and investigate using high-dynamic range (HDR) images for better object detection in autonomous driving systems. To be specific, we present and analyze the following approaches for integrating HDR information into object detection:

- Directly using the LDR image obtained by the camera as input to the existing object detectors.
- Using the HDR image (either reconstructed from multiple LDR images with different exposures or captured by an HDR camera) as input to existing object detectors.
- Using the tone-mapped LDR image (tone-mapped by commonly used tone-mapping operators in the literature) from HDR images as input to existing object detectors.

Our Contributions and Main Results. Existing studies analyzing the different ways HDR images can be used for object detection are limited. Moreover, to the best of our knowledge, there is no study training object detection networks with real-world HDR images from scratch and comparing it with LDR images in a fair way. Our contribution is to provide a systematic and fair analysis on using LDR, HDR and tone-mapped LDR images for object detection in autonomous driving systems. Additionally, we introduce novel performance measures for analyzing the performance of detectors for different illumination conditions.

Our results argue that, tone-mapped HDR images on the average produce better de-

tection results. Surprisingly, using raw HDR images directly without tone-mapping, however, shows inferior performance on average, compared tone-mapped LDR images. Furthermore, not every tone-mapping operator (TMO) is able to bring this performance improvement over LDR or raw HDR images. To find out its reason, we analyse and compare each of these approaches by the proposed evaluation categories based on the dynamic range of the object bounding boxes and number of instances in each class.

To test the hypothesis that HDR image content can help to improve object detection accuracy in autonomous driving settings, we train our object detection network with standard LDR images, tone-mapped LDR images and real-world HDR images, separately and from scratch, as illustrated in Figure 3.1.

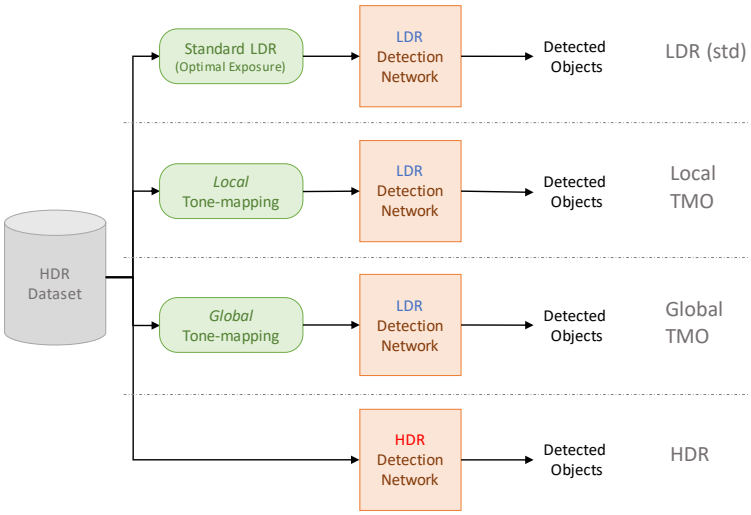


Figure 3.1: The different methods we have analyzed for object detection with LDR and HDR images.

3.1 Methodology

In this section, we describe the methods that we have analyzed and the dataset that we used for evaluating the methods.

3.1.1 Tone-mapping Operators and the Compared Approaches

Tone-mapping operators (TMOs) try to project high-dimensional HDR content into LDR images while preserving the details and color appearance of the original content. For our paper, we have chosen widely used local and global TMOs and compared them with standard LDR and HDR images, namely (see also Figure 3.1):

(i) Standard LDR: As the baseline, we take the "standard" LDR images as input to the object detection network. However, since this was not available in the Cityscapes dataset, we used the method of optimal exposure compression proposed by [64] to achieve the best exposed LDR image from the HDR one.

(ii) HDR:

- HDR: We directly provided the 16-bit HDR images of the CityScapes dataset as input to the detection network.
- HDR with Gamma: Gamma correction with $\gamma = 2.2$ is performed on the HDR images.

(iii) LDR with global tone-mapping:

- Reinhard: The widely-used photographic tone mapping method by Reinhard et al. [41] in global mode.
- Logarithmic compression: Logarithmic compression on the HDR images – this is provided by the CityScapes dataset.

(iv) LDR with local tone-mapping:

- Reinhard: The tone mapping method by Reinhard et al. [41] in local mode.
- Durand: The tone mapping method by Durand et al. [4]. Target contrast is set to 4. For the rest of the parameters, default values in the PFSTools [65] are used.

- Mantiuk: The tone mapping method by Mantiuk et al. [43]. Scaling factor is set to 0.7 and saturation correction is set to 1.0, as used in the OpenCV implementation [66].

- Fattal: The tone mapping method by Fattal et al. [42]. All parameters are default parameters provided by PFSTools [65].

3.1.2 Object Detection Network

In all experiments, we use Faster R-CNN [23] as our detector, as it is the seminal work for two stage detectors, providing a good baseline performance without bells and whistles. We follow the general architecture with backbone Resnet-50 and feature pyramid networks [7] as commonly performed in the literature [7, 6, 67].

3.1.3 Dataset

Being the only readily available dataset with HDR images for autonomous driving, we use the CityScapes dataset [68] in this paper. CityScapes provides 16-bit HDR images and the corresponding LDR images obtained by logarithmic compression. The dataset contains 30 object categories, 8 with instance segmentation labels (namely; car, person, bicycle, rider, motorcycle, truck, bus, and train) and 2975 training, 500 validation and 1525 testing images. The dataset does not include bounding boxes for the objects. We used an existing tool for converting instance segmentation masks to object bounding boxes available in the mmdetection toolbox [69]. Since we need the ground-truth labels for extracting the bounding box information, we are unable to use the actual test set for CityScapes since the ground-truth is not publicly available. Instead, we use the validation set as the test set (500 images), and split the training set into training (2625 images) and validation sets (350 images).

3.2 Experiments and Results

3.2.1 Implementation and training details

To eliminate any possibility of bias, we preferred to train Faster-RCNN from scratch on CityScapes in all experiments, instead of using a detector pretrained on LDR images. We used a similar training configuration for to the ones used in [69, 70] but slightly adjusted for training from scratch. We also tuned the learning rate for each input type. As the optimizer, we used Stochastic Gradient Descent (SGD), which is decreased by a factor of 10 at epoch 88. We also employed linear warm-up with ratio 0.1 at the beginning for 500 iterations. The networks were trained for 104 epochs on a single GPU with a batch size of 4. We also reduced the size of the images by half while keeping the ratio intact.

3.2.2 Evaluation measures

Average Precision (AP). We use AP as our evaluation measure, as it is commonly used in the object detection benchmarks [53, 54]. AP is effectively a measure of the area under the precision-recall curve and AP@0.5 is calculated with 0.50 intersection-over-union (IoU) threshold. We also use the COCO-style mAP [53], which averages AP over 10 IoU thresholds and classes.

AP for different illumination categories. To investigate scenarios where HDR or LDR might be advantageous, we calculate AP for objects (i.e. their bounding boxes) separately for different illumination categories. For this, we use

- *luminance*, which is average of the pixels in the box after converting to gray-scale, $\text{mAP}_{\text{L-LUM}}$ for low luminance (0-5 th percentile), $\text{mAP}_{\text{L-M-LUM}}$ for low-to-medium luminance (5-50th percentile), $\text{mAP}_{\text{M-H-LUM}}$ for medium-to-high luminance (50-95th percentile), and $\text{mAP}_{\text{H-LUM}}$ for high luminance (95-100th percentile).

- *dynamic range* (DR) [71], which is the logarithm of the ratio of maximum luminance to the minimum luminance for the pixels in the box, $\text{mAP}_{\text{L-DR}}$ for low

DR (0-5th percentile), mAP$_{\text{L-M-DR}}$ for low-to-medium DR (5-50th percentile), mAP$_{\text{M-H-DR}}$ for medium-to-high DR (50-95th percentile), and mAP$_{\text{H-DR}}$ for high DR (95-100th percentile).

- *entropy*, which is the entropy of the distribution derived from the luminance values of the pixels in the box, mAP$_{\text{L-ENT}}$ for low entropy (0-5th percentile), mAP$_{\text{L-M-ENT}}$ for low-to-medium entropy (5-50th percentile), mAP$_{\text{M-H-ENT}}$ for medium-to-high entropy (50-95th percentile), and mAP$_{\text{H-ENT}}$ for high entropy (95-100th percentile).

3.2.3 Experiments

On the CityScapes dataset (see Section 3.1.3), we conduct the following experiments.

Table 3.1: Overall performance (mAP scores) for the methods described in Section 3.1.1, where the learning rate is tuned for each method separately.

Method		Learning-rate	AP@0.5	mAP
Std. LDR		4e-3	55.1	33.1
Local TMO	Reinhard [41]	2e-3	55.7	33.2
	Durand [4]	2e-3	55.7	32.9
	Mantiuk [43]	4e-3	55.4	32.7
	Fattal [42]	2e-3	53.9	32.1
Global TMO	Log comp.	2e-3	53.7	31.9
	Reinhard [41]	2e-3	55.0	32.7
HDR with Gamma		2e-3	**56.1**	**33.3**
HDR		8e-3	55.2	32.9

Experiment 1: Overall Performance. Table 3.1 shows the scores calculated over all classes. Overall, we observe that HDR with gamma correction performs slightly better than its closest counterparts, yet we do not observe a strong benefit of using HDR content either in the form of HDR or tone-mapped LDR when compared against the Std. LDR. Additionally, applying gamma correction seems to improve the performance of the HDR images.

Experiment 2: Performance under different illumination categories. This experiment is designed to reveal possible advantages of HDR content, which is absent in Table 3.1: However, we do not observe such advantage in favor of HDR in our experiments. We analyse the performance of different methods on different illumination catefories as defined in Section 3.2.2

- *dynamic range*. Figure 3.2 compares methods under four intervals of dynamic range. However, we do not observe an interval where HDR is advantageous. Nonetheless, some methods including HDR with Gamma and Standard LDR seem to perform slightly better than the rest in high dynamic range areas.

- *luminance*. Figure 3.3 compares methods under four intervals of average luminance. Similarly, we do not observe an interval where HDR is advantageous. Additionally, Standard LDR significantly stands out in low luminance areas.

- *entropy*. Figure 3.4 compares methods under four intervals of average luminance. Similarly, we do not observe an interval where HDR is advantageous, and all methods perform similary wihtout any significant visible gap that is observed in low luminance areas.

Experiment 3: Performance with respect to object size. Object size can be an important factor on the performance of detectors under different illumination conditions because it might be that, e.g., objects closer to the camera appear bigger and they may have higher dynamic range. For this analysis, we extend the original object size intervals in COCO-style mAP calculation [53] from three categories to nine categories. The mAP scores in Figure 3.5 suggest an increase in performance with larger object sizes as expected. However, we do not observe a notable difference between tone-mapped LDR images and HDR images. This observation is supported by Figure 3.6 which plots the distribution of objects with respect to their sizes and dynamic ranges.

Experiment 4: Does HDR need more data? 16-bit HDR naturally spans a larger space of intensity values and therefore, we hypothesize that the HDR-trained detectors might require more data to obtain the same level of performance as LDR images. To test this hypothesis, we consider training LDR and HDR networks with different

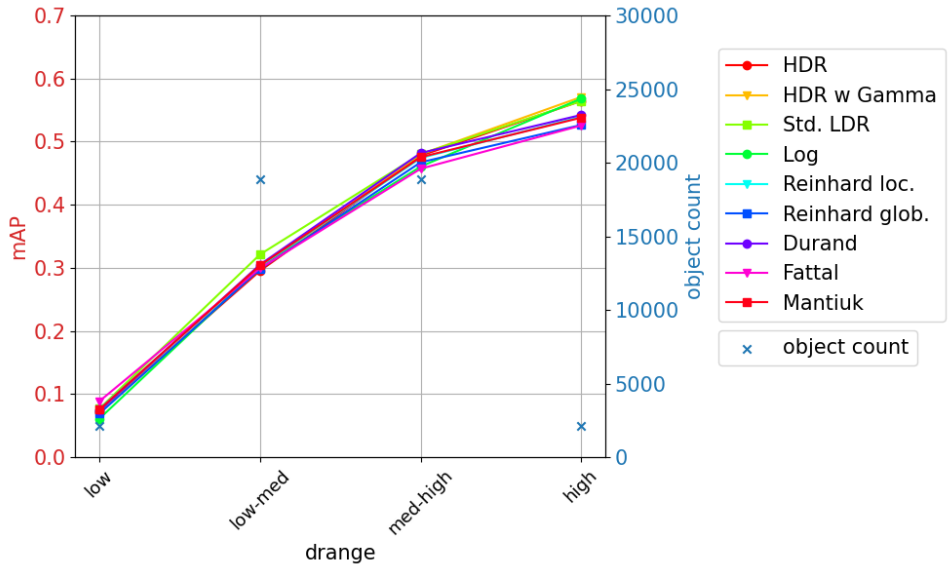


Figure 3.2: Overall performance (mAP scores) under different dynamic range intervals for the methods described in Section 3.1.1. All models are evaluated on the validation set.

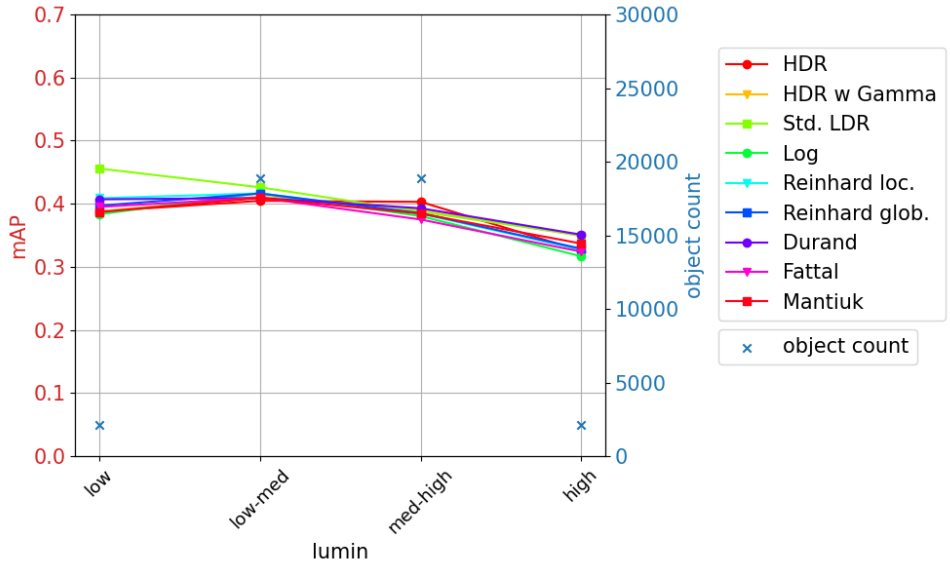


Figure 3.3: Overall performance (mAP scores) under different luminance intervals for the methods described in Section 3.1.1. All models are evaluated on the validation set.

amounts of data. In Figure 3.7, we see that, indeed, an HDR network provides a comparable level of performance with an LDR network with approx. 10% more data.

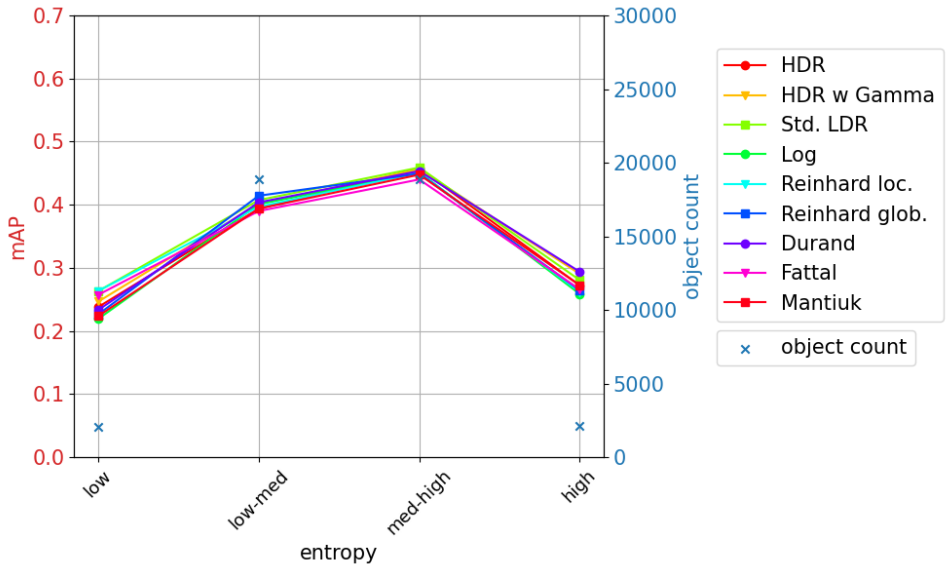


Figure 3.4: Overall performance (mAP scores) under different entropy intervals for the methods described in Section 3.1.1. All models are evaluated on the validation set.

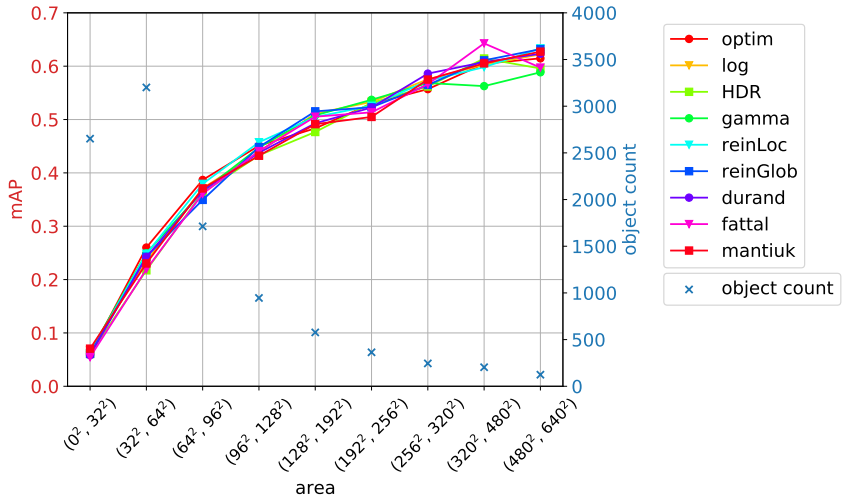


Figure 3.5: Overall performance (mAP scores) for different object size intervals for the methods described in Section 3.1.1.

Qualitative Results. We provide visual detection results in Figure 3.8 from a challenging scene from the CityScapes dataset. In this example scene, we see that HDR content helps detecting the cars further away in the road in an over-exposed region, otherwise undetectable in the LDR content. However, apart from the ones in Figure

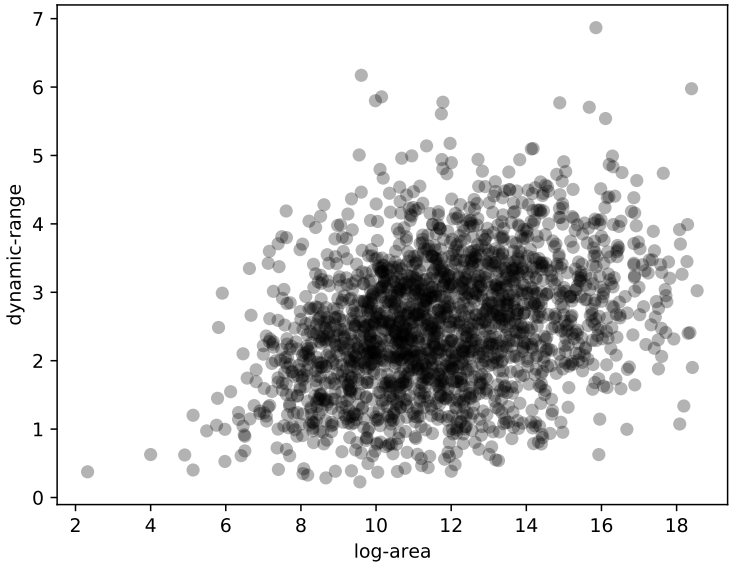


Figure 3.6: The distribution of object bounding box size (log. area) and dynamic range for all object classes.

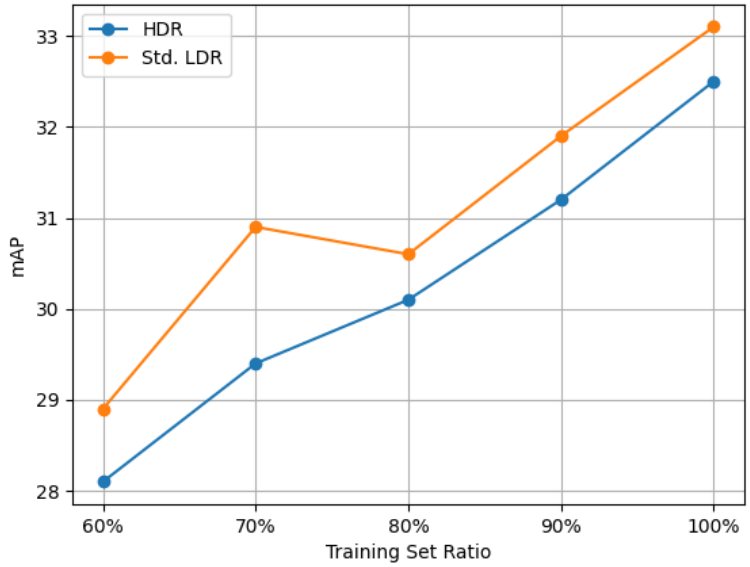
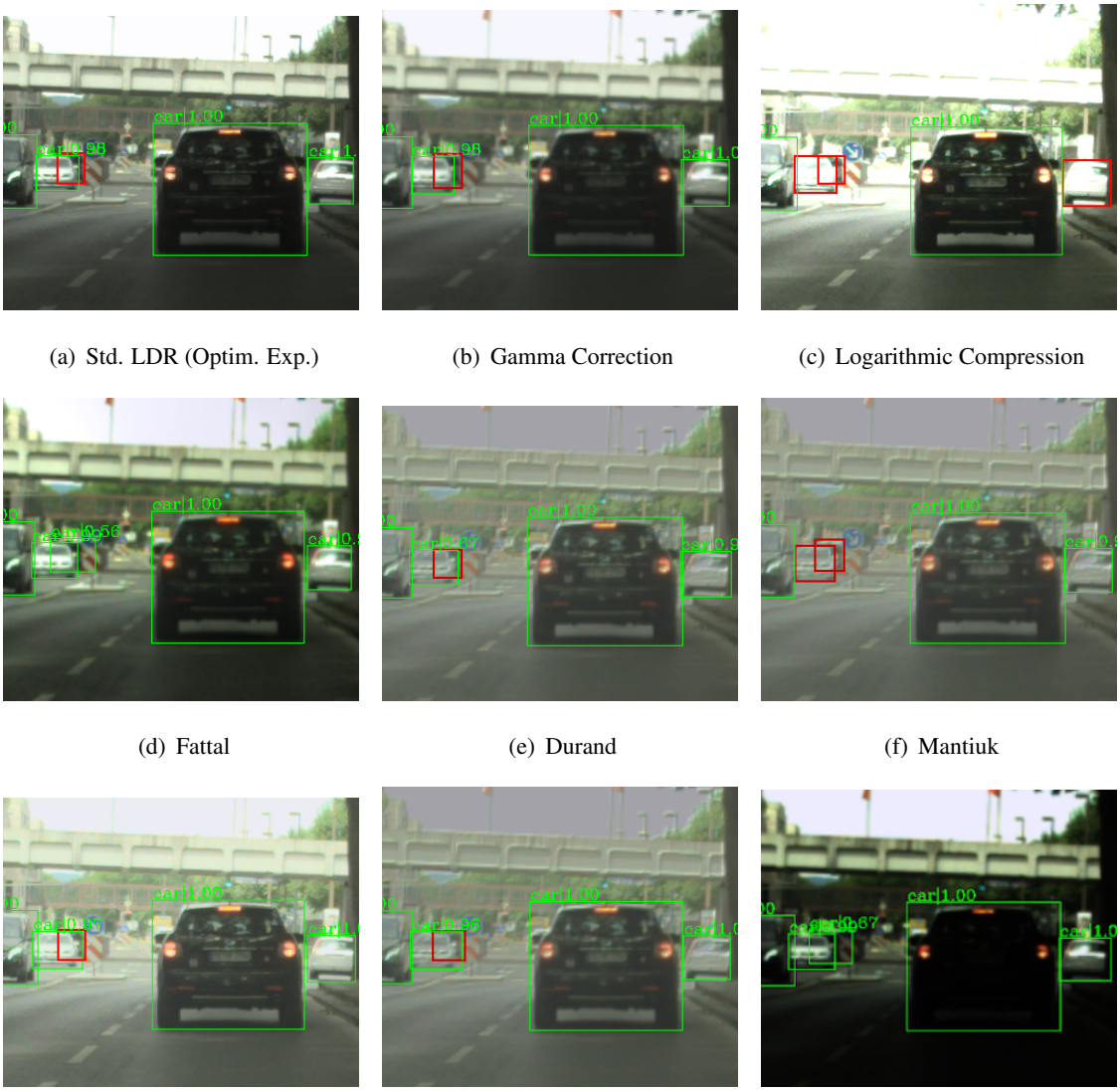


Figure 3.7: Providing less data for Std. LDR and HDR. X-axis Percentages ($X\%$) indicate the ratio of the training set used in the experiment. Y-axis indicate the detections score (mAP).

3.8, we were not able find similar scenes where HDR images became advantageous to use. Thus, given the dataset, we can not confidently conclude that HDR should be used instead of LDR for certain scenarios.



(a) Std. LDR (Optim. Exp.) (b) Gamma Correction (c) Logarithmic Compression

(d) Fattal (e) Durand (f) Mantiuk

(g) Reinhard (Global) (h) Reinhard (Local) (i) 16-bit HDR

Figure 3.8: Detection results. Missed objects are shown in red.

3.3 Discussion

Based on our experiments, we make the following observations:

1. *Overall, HDR images provide **worse** performance than LDR images.*

   The overall analysis in Experiment 1 (Table 3.1) suggests that the improvement obtained by tone-mapped LDR or HDR images is rather minimal.

2. *Illumination conditions of the objects in a scene does not seem to make a difference between HDR images and LDR images*

   Contrary to our expectation that HDR content can help significantly with challenging high dynamic range regions inside a given image, we observe that HDR image performs similarly to the Std. LDR or tone-mapped LDR images when analyzed within different dynamic range intervals for regions containing objects. We attribute this to the lack of challening cases in the dataset. Additionally, applying gamma correction brings improvement over the using HDR images directly.

3. *A detector using HDR images generally requires more data.*

   The reason is that HDR images span a wider range of intensity values and therefore, as the input space is wider, a detector trained on HDR images requires more samples. We provide an experimental evidence for this hypothesis in Figure 3.7 where we compare a HDR-trained detector with an LDR-trained detector using different amount of training data.

CHAPTER 4

JOINT OPTIMIZATION OF TONE MAPPING AND OBJECT DETECTION ALGORITHMS

Due to the advances in deep generative modeling, multiple studies have used GANs to design TMOs, as discussed in Chapter 2. These methods use a discriminator to differentiate between LDR images obtained by a classical TMO and the LDR images obtained by a generator. Although such approaches have produced remarkable tone-mapping results, they rely on the classical tone-mapped LDR images while training the generator, leading to a TMO that combines the aspects of the different classical TMOs depending on the image.

Classical TMOs, designed by hand, do not consider the performance in downstream vision problems, such as object detection or image classification. One natural question arising from this situation is whether one can train this framework with additional objectives that can improve the performance in different downstream vision tasks, instead of only focusing on the making the images look as similar to the classically tone-mapped images as possible. Hence, in this chapter, we propose to jointly optimize GAN-based TMOs and Deep Object Detection Networks.

4.1 Methodology

4.1.1 Overview

An overview of our approach is depicted in Figure 4.1. As illustrated in the figure, we essentially have a GAN-based tone-mapping method that is augmented by the supervision signal of a detector. This is achieved by making the generator to generate

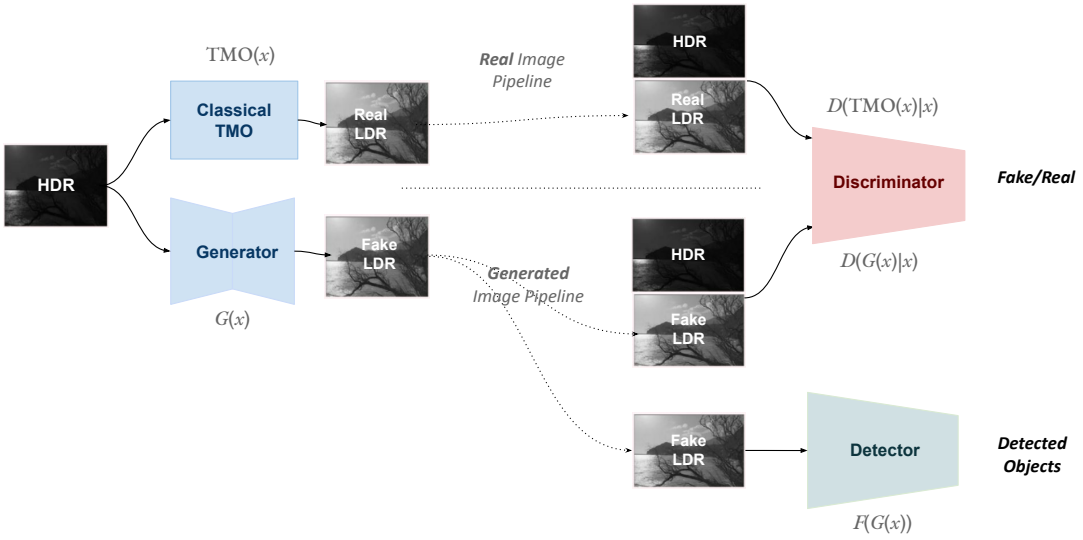


Figure 4.1: Overall architecture diagram for the proposed method that combines object detection and tone-mapping objectives.

images such that not only a visual similarity to a classical TMO result is enforced, but the object detection performance on the generated images are also taken into account.

4.1.2 Details of the Proposed Method

In order to jointly optimize the object detection network and deep learning-based TMOs, we modify the objective in the conditional least squares GAN-based TMO framework as follows:

$$G^* \leftarrow \arg\min_{G,F}\min_{D} \mathcal{L}_D + \alpha_{\text{det}}\mathcal{L}_{Det} + \alpha_{\text{non-det}}(\mathcal{L}_G + \beta\mathcal{L}_{GPL} + \gamma\mathcal{L}_{FM}), \tag{4.1}$$

where D, G and F represent discriminator, generator and object detection networks, respectively; and α_{det}, β and γ are the weights for the detection loss (classification and localization combined), gradient profile loss and feature matching loss, respectively. $\alpha_{\text{non-det}}$ controls the weight for all the losses that are not related to detection directly. The individual loss terms are defined as follows:

$$\mathcal{L}_{Det} = \mathbb{E}_{x\sim p_{data}}[L_{cls}(F(G(x)) + \lambda L_{loc}(F(G(x)))], \tag{4.2}$$

$$\mathcal{L}_G = \frac{1}{2}\,\mathbb{E}_{x\sim p_{data}}[(1 - D(G(x)|x))^2], \tag{4.3}$$

$$\mathcal{L}_D = \frac{1}{2}\,\mathbb{E}_{x\sim p_{data}}[(D(\text{TMO}(x)|x) - 1)^2] + \frac{1}{2}\,\mathbb{E}_{x\sim p_{data}}[D(G(x)|x)^2], \tag{4.4}$$

where x and TMO(x) represent the HDR and ground-truth tone-mapped LDR images; the overall block diagram can be examined in Figure 4.1.

Note on α_{det} and $\alpha_{\text{non-det}}$. In order to decouple the effect of detection loss on the generator and the detector, we apply the α_{det} in the backward pass in between the detector and the generator, by scaling the gradients flowing from the detector to the generator. In this way, we manage not to cause the detector get unnecessarily large (or small) updates while also providing stronger (or weaker, in case it is needed) influence on the generator from detection objective. This becomes necessary for α_{det} as we update the generator and the detector on the same input simultaneously, whereas it is not necessary for $\alpha_{\text{non-det}}$ due to alternating updates between the generator and the discriminator.

4.1.2.1 Architecture Details

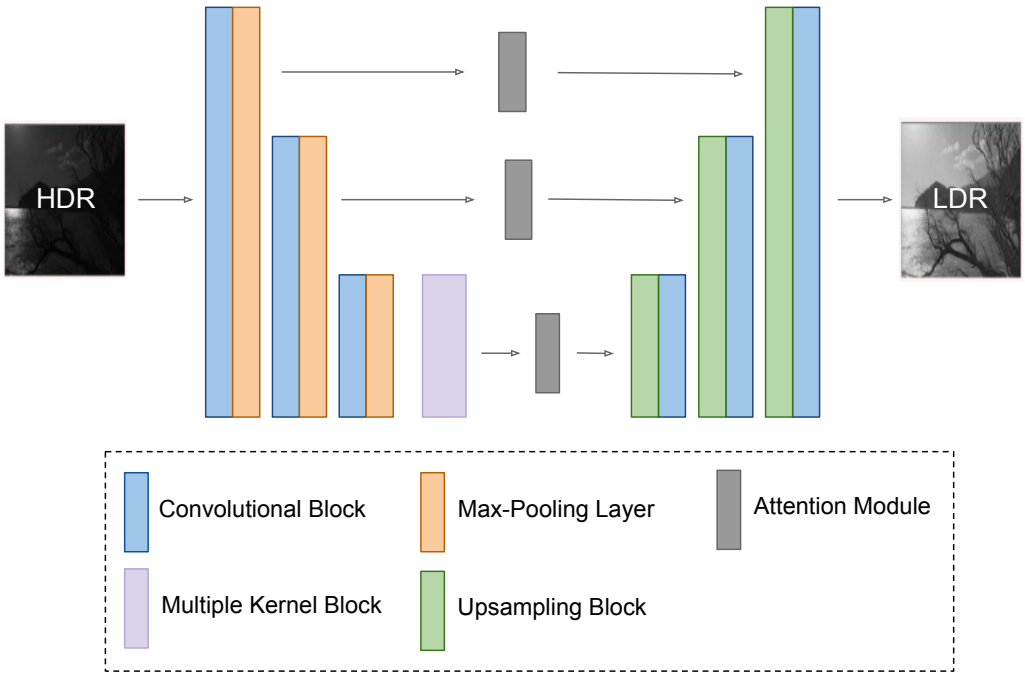


Figure 4.2: Overall diagram for the proposed generator architecture.

Generator (G). The proposed generator is a UNet-based [72] generator where upsampling convolutions are replaced by bilinear upsampling as suggested by Panetta et al. [50]. We use leaky ReLU as the activation function and Instance Normalization [73] for normalizing the activation values. Additionally, we augment the skip connections in the UNet architecture with attention [74]. Each feature map in a single level of the network is carried across by a separate attention module, where attention queries $Q^{(i)}$ for layer i are calculated from original image while keys $K^{(i)}$ and values $V^{(i)}$ are calculated from intermediate feature maps as follows (following the attention notation from [74]):

$$\begin{aligned} Q^{(i)} &= A_Q^i(I^{(i)}), \\ K^{(i)} &= A_K^i(G^{(i)}(I)), \\ V^{(i)} &= A_V^i(G^{(i)}(I)), \end{aligned} \tag{4.5}$$

where i represents layer index for G. A_Q^i, A_K^i, A_V^i are single layer convolutional networks, and $I^{(i)}$ is the original image downsampled to the spatial size of the feature map in $G^{(i)}$. The resulting feature maps are calculated as follows [74]:

$$\text{Attention}(Q^{(i)}, K^{(i)}, V^{(i)}) = \text{softmax}\left(\frac{Q^{(i)}K^{(i)^\top}}{\sqrt{d_k}}\right)V^{(i)}, \tag{4.6}$$

where d_k is the normalization constant which corresponds to the dimensionality of the $Q^{(i)}$ and $K^{(i)}$ [74].

Jiang et al. [75] also use a similar approach, however they use the constant brightness map of the image as attention weights at all levels, whereas our module learns the attention weights. At the innermost layer of the generator, we also convolve the feature maps with multiple kernels with different sizes (Multiple Kernel Block); namely 3, 5, 7 and 9, and concatenate the resulting feature maps. To speed up the attention mechanism we also make use of the efficient attention mechanism proposed by Shen et al. [76], which reduces the complexity of the calculation to linear-time and linear-space. The overall architecture can be seen in Figure 4.2.

Discriminator (D). The discriminator is a standard patch discriminator which has a 70×70 receptive field in the final layer [22]. Similar to the generator, we use leaky ReLU as the activation function and Instance Normalization for normalizing the activation values.

Detector (F). As the one-stage detector, we use RetinaNet [6], since it is a prominent model for one stage detectors providing a reasonable starting point among detectors. We follow the general architecture with backbone Resnet-50 [26] and feature pyramid networks [7]. As the two stage detector, we use Faster R-CNN [23], similarly using Resnet-50 [26] and feature pyramid networks [7] for feature extraction.

4.2 Experimental Settings and Details

4.2.1 Tone-mapping Operators and the Compared Approaches

In this chapter, we have selected the following approaches for comparison:

(i) Detection with LDR images:

- **LDR**: We extract the the middle exposure from each HDR image to obtain LDR images similar to Mukherjee et al. [57].
- **Std. LDR**: Additionally, we take the optimal exposure LDR images as input to the object detection network. We use the method of optimal exposure compression proposed by [64] to achieve the best exposed LDR image from the HDR one, as if the scene is captured by a virtual LDR camera with an optimal exposure set-up.

(ii) Detection with HDR images:

- **HDR**: We directly provide the HDR images.
- **HDR with Gamma**: Before feeding the images to the network; firstly, we apply min-max normalization to each image by subtracting its minimum value and dividing by its pixel range, secondly we apply gamma encoding (correction), and finally we scale it to the range of pixel values of an LDR image (which is $[0, 255]$).

(iii) Detection with LDR images obtained by the classical tone-mapping operators:

- **Ashikhmin**: The tone mapping method by Ashikhmin et al. [77].

- **Reinhard**: The tone mapping method by Reinhard et al. [41] in local mode.

- **Durand**: The tone mapping method by Durand et al. [4]. Target contrast is set to 4. For the rest of the parameters, default values in the PFSTools [65] are used.

- **Mantiuk**: The tone mapping method by Mantiuk et al. [43]. Scaling factor is set to 0.7 and saturation correction is set to 1.0, as used in the OpenCV implementation [66].

- **Fattal**: The tone mapping method by Fattal et al. [42]. All parameters are default parameters provided by PFSTools [65].

- **Best TMQI per picture**: For this method, we choose the best performing tone-mapping operator in TMQI metric for each picture in the dataset.

(iv) Detection with learning-based tone-mapping:

- **TMO-GAN**: Our proposed architecture, without jointly training with an object detector.

- **TMO-GAN + Detector**: Proposed architecture with the detector, which is jointly trained with TMO-GAN. This method has two versions in terms of initialization of the each component:
  (i) **COCO version.** Detector is pre-trained on MS COCO dataset and TMO-GAN is randomly initialized,
  (ii) **OOD version.** TMO-GAN is pre-trained on OOD (out-of-distribution) dataset without the detector, and the detector is pre-trained disjointly on the outputs of the trained and frozen TMO-GAN on top of the MS COCO pre-training.

  Additionally, in terms of the detector training, we train the detector either (i) only on the generated images or (ii) on real ground-truth images together with generated ones, to ensure the stability of the detector.

4.2.2 Dataset

For all the experiments in this section, we use the OOD dataset [55] which consists of HDR images with annotated labels for 20 classes from the Pascal VOC [54] dataset. We filter and divide the dataset such that we have 1491 training and 380 test images. Additionally we downsize the dataset into 1024×576 resolution before performing the experiments. We form our ground truth LDR images by selecting the best classical TMO among the given ones in Section 4.2.1 for each picture based on the TMQI (Tone-mapping Quality Index) metric [2].

4.2.3 Implementation and training details

Initializations. We initialize all RetinaNet architectures at least from COCO pre-trained versions. For joint training with TMO-GAN, we employ different initialization for the detector and TMO-GAN as mentioned in Section 4.2.1.

Data Augmentation. We perform the same data augmentation techniques for all experiments. We apply random cropping with minimum of 0.3 scaling factor, so that the crop contains at least 1 ground truth object bounding box with minimum of 0.3 Intersection-over-Union with the original box. Furthermore, we apply random horizontal flipping with a probability of 0.5.

4.2.3.1 Dataset Tone-mapped by Classical TMOs + Detector (disjoint training)

For the detectors we use Stochastic Gradient Descent (SGD) with a learning rate of 0.001, which is decreased by a factor of 10 at epoch 7. We also employ linear warm-up with ratio 0.1 at the beginning for 500 iterations. The networks are trained for 14 epochs on a single GPU with a batch size of 8. We also reduce the size of the images to 1024×576.

4.2.3.2 Dataset Tone-mapped by TMO-GAN + Detector (disjoint training)

We use Adam [78] with a learning rate of 0.0002 for generator and discriminator. The networks are trained for 20 epochs on a single GPU with a batch size of 8. After 20 epochs, the learning rate is decayed to 0 linearly until 50^{th} epoch. We also reduce the size of the images to 1024 width and 576 height. β is set to 0.8 and γ is set to 10. The detector is trained in an identical way to Section 4.2.3.1.

4.2.3.3 Dataset Tone-mapped by TMO-GAN + Detector (joint training)

- **COCO version.** We use Adam [78] with a learning rate of 0.0002 for the generator and the discriminator. They are trained for 20 epochs on a single GPU with a batch size of 8. After 20 epochs, the learning rate is decayed to 0 linearly until 50^{th} epoch. Simultaneously, using SGD, the learning rate for detector starts as 0 and linearly increases to 0.001 for 5 epochs. Then, it is trained for 25 epochs. Finally the learning rate is decayed by a factor of 10 and trained for another 5 epochs.

- **OOD version.** We finetune the pre-trained networks using Adam [78] with a learning rate of 0.00001 for the generator, the discriminator and the detector. They are trained for 20 epochs on a single GPU with a batch size of 8.

During experiment, β is set to 0.8 whereas γ is set to 10, for both versions. Similarly, we reduced the size of the images to 1024×576, for both versions. α_{det} and $\alpha_{non-det}$ are set to 1 for all of the experiments except for the one where we provide different weights for different objectives using α_{det} and $\alpha_{non-det}$, and analyse the results.

4.2.4 Evaluation measures

We use the same evaluation measures, namely mAP, as the previous chapter (Section 3.2.2). For evaluation of tone-mapping quality, we use the Tone-mapping Quality Index (TMQI) as our measure [2], which outputs three different scores: (i) TMQI-Q giving the overall quality score, (ii) TMQI-N giving the naturalness score of the tone-

mapped image, and (iii) TMQI-S giving the structural fidelity with respect to the original HDR image. For additional quality metrics, we use the probability of contrast loss and contrast amplification outputs of the contrast invariant visual difference predictor (CIVDM) [3]. However, as the output probabilities are given for each pixel in the image, we take the pixel with the maximum probability to reduce the score to a scalar value, which is also performed by Mantiuk et al. [79] for probability maps. We denote the maximum probability of visible contrast loss as CL_{max} and maximum probability of visible contrast amplification as CA_{max}.

4.3 Experiments and Results

4.3.1 Experiment 1: TMO-GAN - TMO Quality

In this experiment, we provide an ablation study and comparative analysis for the TMO-GAN. We train the TMO-GAN without the detector using the following optional additions:

- TMO-GAN: The proposed architecture without attention modules.
- TMO-GAN + *hard-tanh*: Hard-tanh function is applied to the output of the generator to confine its range to $[0, 1]$:

$$hard\text{-}tanh(x) = \begin{cases} x, & \text{if } -1 \leq x \leq 1, \\ 1, & \text{if } x > 1, \\ -1, & \text{if } x < -1, \end{cases}$$

- TMO-GAN + *attention*: Attention module described in Figure 4.2 is added to the generator.
- TMO-GAN + *skip-image*: A scaled version of the input image (0.8 is used in the experiments) is added to the output of the generator.

In Table 4.1, we observe that, when all additions (hard-tanh, attention, and skip-image) are integrated to the generator, TMO-GAN outperforms all other single TMOs in terms of overall quality (TMQI-Q), while it gets very close to the the dataset formed

Table 4.1: Tone-mapping quality results for multiple TMOs, comparing hand-crafted TMOs with ours and other deep-learning based approach proposed by Rana et al. [1]. TMQI-Q, TMQI-N and TMQI-S give the scores for overall quality, naturalness and structural fidelity, respectively [2]. CL_{max} and CA_{max} columns indicate the probabilities of contrast loss and contrast amplification, proposed by Aydin et al. [3]. hard-tanh column indicates whether hard-tanh is applied to the output of the generator. Attention column indicates whether attention mechanism defined in Figure 4.2 is included. skip-image column indicates whether original image is added to the output of the generated image. Best scores are given in bold font, second best scores are underlined.

Method	hard-tanh	attention	skip-image	TMQI ↑			CL_{max} ↓	CA_{max} ↓
				Q	N	S		
LDR	-	-	-	76.1	79.8	4.4	0.99	0.54
Std. LDR	-	-	-	89.7	90.9	53.1	0.88	**0.51**
Ashikhmin	-	-	-	88.4	88.8	47.9	0.69	0.73
Durand	-	-	-	89.0	**92.2**	45.5	0.66	0.63
Fattal	-	-	-	88.8	**92.2**	45.4	0.85	0.72
Mantiuk	-	-	-	86.5	91.6	34.2	**0.46**	0.60
Reinhard	-	-	-	89.6	85.9	71.5	0.76	0.61
Best TMQI per pict.	-	-	-	**94.9**	<u>91.9</u>	**80.3**	0.78	0.69
DeepTMO [1]	-	-	-	93.4	89.8	74.0	0.80	0.83
TMO-GAN	X	X	X	91.2	72.2	78.1	0.78	0.91
TMO-GAN	✓	X	X	94.2	90.5	78.3	0.85	0.80
TMO-GAN	✓	X	✓	94.2	90.7	78.1	0.82	0.79
TMO-GAN	✓	✓	X	94.4	90.8	78.8	0.81	0.84
TMO-GAN	✓	✓	✓	<u>94.6</u>	90.9	<u>79.7</u>	0.81	0.83

by best TMQI per-picture strategy. It should be noted that it is not possible to try all TMO algorithms and determine the best algorithm for each image in any practical scenario; hence this case is an oracle. Additionally, we see that TMO-GAN falls behind for other TMO evaluation metrics CL_{max} and CA_{max}. Expectedly, it performs similarly to Best TMQI per picture method as this method is the ground-truth for the training.

We also provide qualitative results in Figure 4.3. Firstly, we can observe that vanilla

TMO-GAN in (a) produces artefacts without the hard-tanh. Adding the hard-tanh, we observe that the output in (b) becomes quite similar to the ground-truth in (d). Moreover, when we use activation, attention and skip-image all together, we see that the generator improves over the ground truth making the very bright areas on the table more visible. However, other TMOs such as Durand and Mantiuk are also able to capture the details on the table, while providing darker images overall.

(a) TMO-GAN (b) TMO-GAN + hard-tanh (c) TMO-GAN + hard-tanh + attention + skip-image

(d) Best TMQI (Reinhard) (e) Durand (f) Mantiuk

(g) Std. LDR (optimal exposure) (h) Ashikhmin (i) Fattal

Figure 4.3: Qualitative tone-mapping results for different methods.

4.3.2 Experiment 2: TMO-GAN + Detector, Joint Training

In these experiments, we jointly train TMO-GAN with the detector. We perform the same experiments with RetinaNet and Faster-RCNN. Tables 4.2 and 4.3 compares the detection performance and TMO quality of the proposed architecture against classical methods and disjoint training. Based on the result in Tables 4.2 and 4.3, we observe

that,

- TMO-GAN + RetinaNet OOD achieves the best detection scores while being able to preserve the high quality tone-mapping in terms of TMQI compared to TMO-GAN (vanilla). However, the detection performance is not significantly higher than the best classical method (Mantiuk).
- Training on real images on top of generated images simultaneously improves the performance. We hypothesise that this make the training of the detector more stable in the beginning as the generated images have worse quality in the initial phase of the training. Additionally, using a pretrained TMO-GAN together with the pre-trained detector can also further improve the detection performance.
- Faster-RCNN performs worse compared to RetinaNet in the OOD dataset, for both classical methods and proposed joint training. Additionally, unlike RetinaNet case, joint training does not improve over the classical methods and disjoint training.

We also compare our joint training methodolgy to other methods under three categories of illumination similar to Chapter 3 (see Section 3.2.2 for the definitions).

- *dynamic range*. Figures 4.4 and 4.7 shows the detection performance under four intervals of dynamic range. We do not observe strong agreement between RetinaNet nad Faster-RCNN in low dynamic range areas. However they agree on high dynamic range areas. Our architectures (TMO-GAN + RetinaNet and TMO-GAN + Faster-RCNN) performs similarly to other TMOs. Finally, unlike other evaluation categories (entropy and luminance), dynamic-range brings out the most difference in edge cases: in areas with lowest dynamic-range, the methods seems to differ the most. Particularly, HDR without normalization and gamma correction performs worse in low dynamic range areas in both figures.
- *luminance*. Figures 4.5 and 4.7 compares methods under four intervals of average luminance. Contrary to our intuitions, HDR performs worse in underexposed low luminance areas. Nevertheless, applying normalization and gamma

correction makes HDR images the best performer for low luminance regions. Additionally, our architectures (TMO-GAN + RetinaNet and TMO-GAN + Faster-RCNN) performs similarly to other TMOs.

- *entropy*. Figures 4.6 and 4.9 compares methods under four intervals of entropy values. Similarly, we see an inferior performance with HDR images in low entropy areas when no normalization or gamma correction is done. Our architectures (TMO-GAN + RetinaNet and TMO-GAN + Faster-RCNN) performs similarly to other TMOs.

Table 4.2: Overall performance (mAP scores for detection, and TMQI and CIVDM for TMO quality) for the methods described in Section 4.2.1, where the detector is chosen as RetinaNet. The best are shown in bold. RT: RetinaNet.

TMO	Joint Train.	Train on Real	mAP ↑	TMQI-Q ↑	CL_{max} ↓	CA_{max} ↓
HDR	-	-	26.3	-	-	-
HDR with Gamma	-	-	29.8	-	-	-
LDR	-	-	28.2	76.1	0.99	0.54
Std. LDR	-	-	31.0	88.9	0.88	**0.51**
Durand	-	-	30.6	89.0	0.66	0.63
Mantiuk	-	-	31.3	86.5	**0.46**	0.60
Reinhard	-	-	29.6	89.6	0.76	0.61
Fattal	-	-	29.8	88.8	0.85	0.72
Ashikhmin	-	-	30.1	88.4	0.69	0.73
Best TMO per Pict.	-	-	30.0	**94.9**	0.78	0.69
TMO-GAN	X	-	30.0	94.6	0.81	0.83
TMO-GAN+RT COCO	✓	X	29.0	94.1	0.85	0.85
TMO-GAN+RT COCO	✓	✓	30.2	94.2	0.83	0.84
TMO-GAN+RT OOD	✓	✓	**31.6**	94.5	0.82	0.83

4.3.3 Experiment 3: The effect of α_{det} and $\alpha_{\text{non-det}}$

In these experiments, we try a range of values for the weights applied on the two different objectives. We chose RetinaNet for the detector component, and trained it on synthetic and real images together. We also equipped TMO-GAN with all additional features: hard-tanh, attention and skip-image, and jointly trained the overall

Table 4.3: Overall performance (mAP scores for detection, and TMQI and CIVDM for TMO quality) for the methods described in Section 4.2.1, where the detector is chosen as Faster R-CNN. The best are shown in bold. FCNN: Faster R-CNN.

TMO	Joint Train.	Train on Real	mAP ↑	TMQI-Q ↑	CL_{max} ↓	CA_{max} ↓
HDR	-	-	23.5	-	-	-
HDR with Gamma	-	-	27.7	-	-	-
LDR	-	-	24.7	76.1	0.99	0.54
Std. LDR	-	-	28.3	88.9	0.88	**0.51**
Durand	-	-	28.8	89.0	0.66	0.63
Mantiuk	-	-	29.1	86.5	**0.46**	0.60
Reinhard	-	-	28.1	89.6	0.76	0.61
Fattal	-	-	**29.5**	88.8	0.85	0.72
Ashikhmin	-	-	28.9	88.4	0.69	0.73
Best TMO per Pict.	-	-	29.3	**94.9**	0.78	0.69
TMO-GAN	X	-	28.6	94.6	0.81	0.83
TMO-GAN+FCNN COCO	✓	X	26.3	94.2	0.94	0.81
TMO-GAN+FCNN COCO	✓	✓	27.3	94.0	0.87	0.84
TMO-GAN+FCNN OOD	✓	✓	27.7	94.3	0.81	0.83

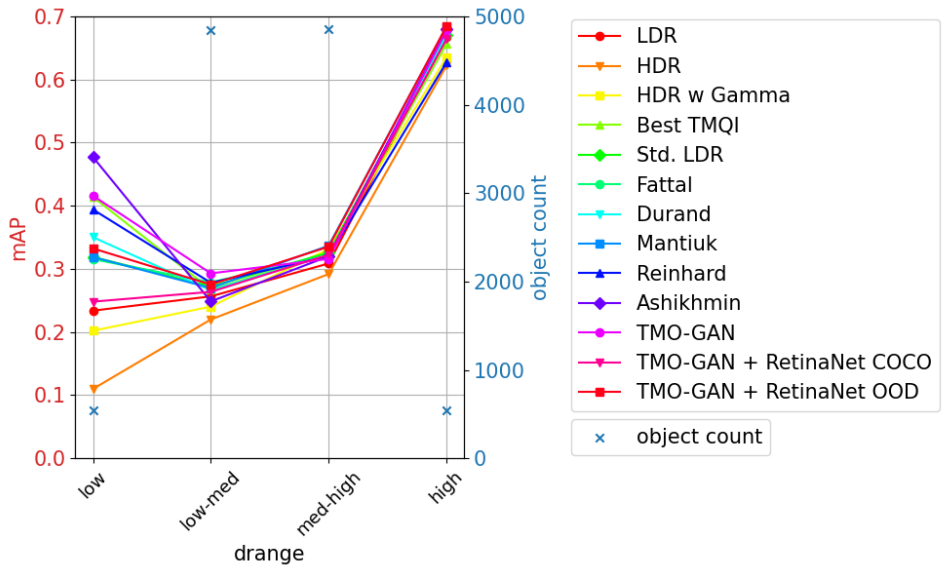


Figure 4.4: Overall performance (mAP scores) for RetinaNet under different dynamic range intervals for the methods described in Section 4.2.1.

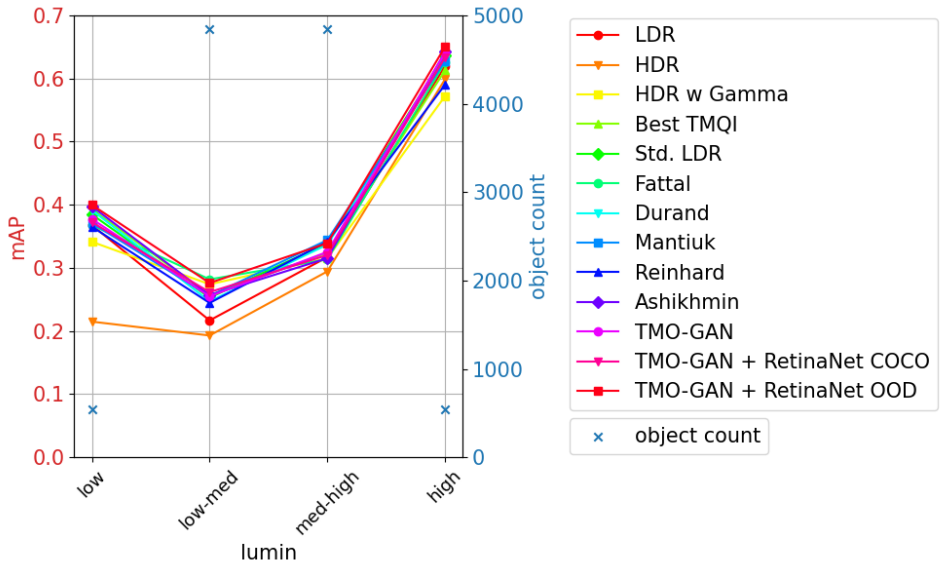


Figure 4.5: Overall performance (mAP scores) for RetinaNet under different luminance intervals for the methods described in Section 4.2.1.

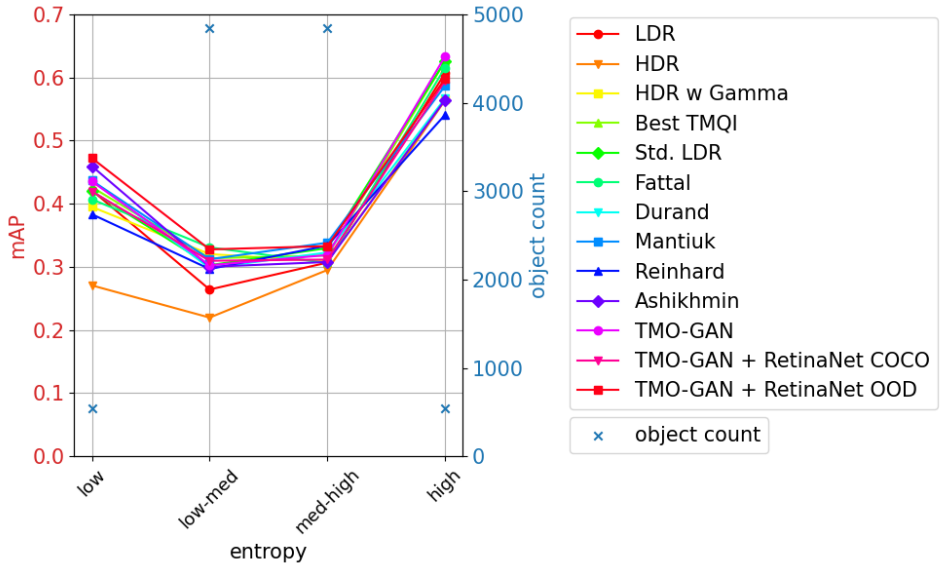


Figure 4.6: Overall performance (mAP scores) for RetinaNet under different dynamic range intervals for the methods described in Section 4.2.1.

architecture similar to Experiment 2.

As it can be examined from Table 4.4, larger influence of the detector on the generator does not improve the results, producing close score to the baseline. However, after

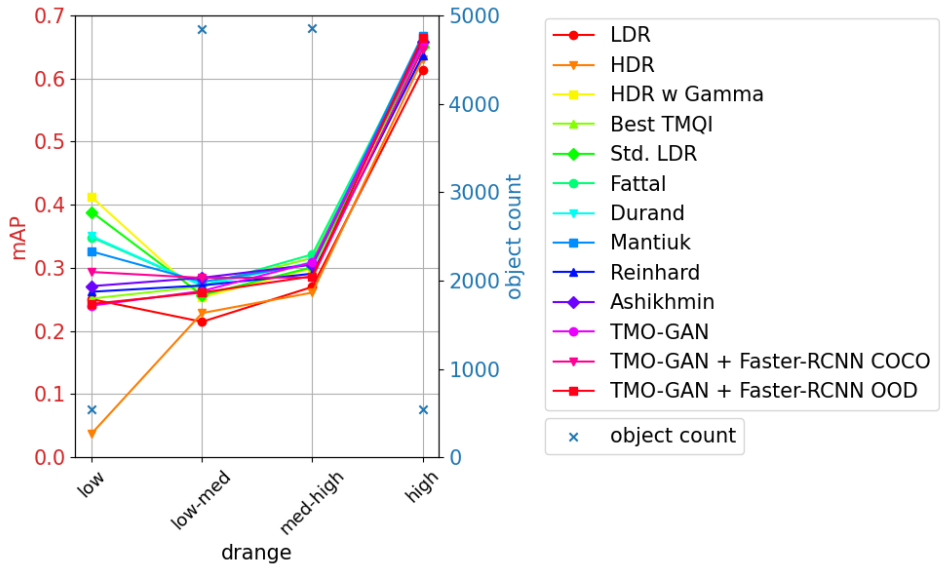


Figure 4.7: Overall performance (mAP scores) for Faster-RCNN under different dynamic range intervals for the methods described in Section 4.2.1.

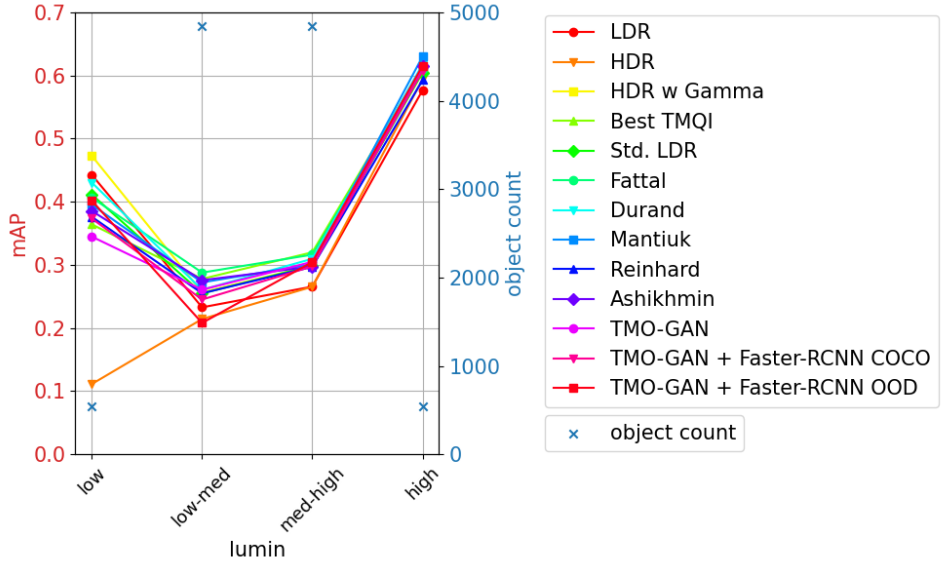


Figure 4.8: Overall performance (mAP scores) for Faster-RCNN under different luminance intervals for the methods described in Section 4.2.1.

fine-tuning the contribution from the detector, we achieve a peak detection performance when $\alpha_{\text{det}} \sim 1.2$. Furthermore, decreasing the influence of the discriminator related objectives (Equation 4.1) via $\alpha_{\text{non-det}}$ deteriorates the detection performance significantly.

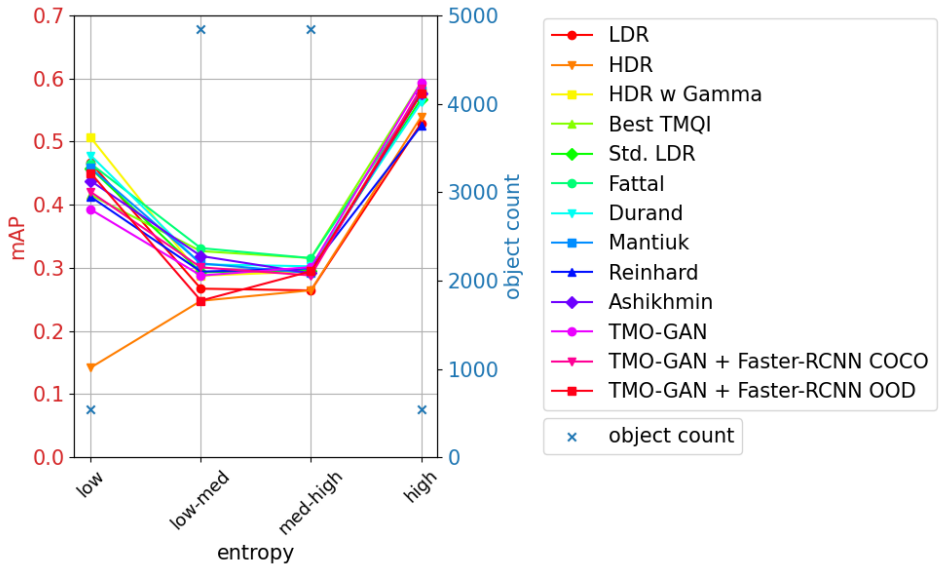


Figure 4.9: Overall performance (mAP scores) for Faster-RCNN under different entropy intervals for the methods described in Section 4.2.1.

4.3.4 Experiment 4: Object-aware Patch Discriminator

In this step, we propose applying different weights for the discriminator feedback for locations with objects and without objects. As our primary aim is to detect objects, using this approach, we aim to penalize the generator to produce better images on the locations containing objects. We produce binary masks which contain 1 if the pixel belongs to any object, or 0 otherwise. Then, we resize this mask to the output of the patch discriminator by using nearest neighbor interpolation. Finally we use the mask to apply increased weights to the locations that contain objects as follows:

$$\mathcal{L}_G = \frac{1}{w \times h} \sum_{i,j} \left[\lambda_{obj} M_{i,j} D(I)_{i,j} + (1 - M_{i,j}) D(I)_{i,j} \right], \tag{4.7}$$

where M is resized binary mask; and D is the discriminator; and I is the input image. i and j are the coordinates of the discriminator output with width w and height h. λ_{obj}, on the other hand, designates the weight applied to the locations that contain objects. For the experiments, we use the same settings as in Experiment 2 that combines TMO-GAN and RetinaNet.

As shown in Table 4.5, we find that setting $\lambda_{obj} = 6$ improves the detection scores significantly, while all other values degrade the overall performance.

Table 4.4: Overall performance (mAP scores for detection, and TMQI, CIVDM for TMO quality) for different values of α_{det} and $\alpha_{\text{non-det}}$, where the RetinaNet is chosen as the detector. The best are shown in bold.

TMO	α_{det}	$\alpha_{\text{non-det}}$	mAP ↑	TMQI-Q ↑	CL$_{max}$ ↓	CA$_{max}$ ↓
HDR with Gamma	-	-	29.8	-	-	-
Mantiuk	-	-	31.3	86.5	**0.46**	**0.60**
Best TMO per Pict.	-	-	30.0	**94.9**	0.78	0.69
TMO-GAN	-	-	30.0	94.6	0.81	0.83
TMO-GAN + RetinaNet COCO	0.8	1.0	29.6	94.2	0.84	0.84
TMO-GAN + RetinaNet COCO	1.0	1.0	30.2	94.2	0.83	0.84
TMO-GAN + RetinaNet COCO	1.2	1.0	**31.4**	94.2	0.81	0.85
TMO-GAN + RetinaNet COCO	1.5	1.0	30.9	94.2	0.85	0.83
TMO-GAN + RetinaNet COCO	2.0	1.0	29.8	94.2	0.90	0.84
TMO-GAN + RetinaNet COCO	4.0	1.0	28.9	94.1	0.81	0.82
TMO-GAN + RetinaNet COCO	6.0	1.0	30.1	94.4	0.86	0.85
TMO-GAN + RetinaNet COCO	8.0	1.0	29.4	94.3	0.83	0.84
TMO-GAN + RetinaNet COCO	1.0	0.8	28.0	93.9	0.91	0.81
TMO-GAN + RetinaNet COCO	1.0	0.6	28.4	93.3	0.91	0.85
TMO-GAN + RetinaNet COCO	1.0	0.4	27.9	93.3	0.94	0.83
TMO-GAN + RetinaNet COCO	1.0	0.2	28.3	92.1	0.91	0.82
TMO-GAN + RetinaNet COCO	1.0	0.1	27.3	91.2	0.90	0.81

Table 4.5: Overall performance (mAP scores for detection, and TMQI, CIVDM for TMO quality) for different values of λ_{obj}, where the RetinaNet is chosen as the detector. The best are shown in bold.

TMO	λ_{obj}	mAP ↑	TMQI-Q ↑	CL$_{max}$ ↓	CA$_{max}$ ↓
HDR with Gamma	-	29.8	-	-	-
Mantiuk	-	31.3	86.5	**0.46**	**0.60**
Best TMO per Pict.	-	30.0	**94.9**	0.78	0.69
TMO-GAN + RetinaNet COCO	1.0	30.2	94.2	0.83	0.84
TMO-GAN + RetinaNet COCO	2.0	29.2	94.1	0.79	0.85
TMO-GAN + RetinaNet COCO	4.0	28.8	94.1	0.89	0.84
TMO-GAN + RetinaNet COCO	6.0	**31.6**	94.2	0.83	0.84
TMO-GAN + RetinaNet COCO	8.0	28.8	93.9	0.88	0.83

Table 4.6: Performance of the TMO-GAN with and without discriminator, revealing the improvement achieved with the joint objective for object detection over the baseline of generator + detector.

TMO	mAP ↑	TMQI-Q ↑	CL_{max} ↓	CA_{max} ↓
HDR with Gamma	29.8	-	-	-
Mantiuk	31.3	86.5	**0.46**	**0.60**
Best TMO per Pict.	30.0	**94.9**	0.78	0.69
TMO-GAN	30.0	94.6	0.81	0.83
TMO-GAN + RetinaNet OOD	**31.6**	94.2	0.83	0.84
TMO-GAN - Discriminator + RetinaNet COCO	28.2	87.3	0.80	0.87

4.3.5 Experiment 5: HDR vs. TMO-GAN without the Discriminator

With joint training, we effectively use a larger network (Generator + Detector) to detect objects. Here, we design an experiment where we can see how much TMO-GAN + Detector (Joint Training) improves the detection performance on HDR images over just using a larger network . In order to achieve this goal, we remove the discriminator from the architecture and use only the detection loss (equivalent to setting $\alpha_{\text{non-det}}$ to zero). As we can see from Table 4.6, the baseline network without the discriminator can improve over the HDR. However, it falls behind our joint method which show the additional improvement provided by the joint training. This result is also in accordance with the results in Experiments 3 where decreasing $\alpha_{\text{non-det}}$ hurts the performance as well.

4.3.6 Experiment 6: Detection Performance vs. TMO Quality

In this experiment, we aim to examine the relation between the detection performance (mAP) and TMO quality metrics (TMQI-Q, CL_{max}, CA_{max}). To that end, we plot the mAP scores against the TMO quality metrics separately in the Figure 4.10, for different methods used in this chapter. As can be seen in the figure, TMO quality metrics show different trends for Classical TMOs and our architectures. Firstly, changes in detection performance affect our architectures (TMO-GAN and TMO-GAN + RetinaNet) much less compared to classical TMOs. This may be due the fact that our

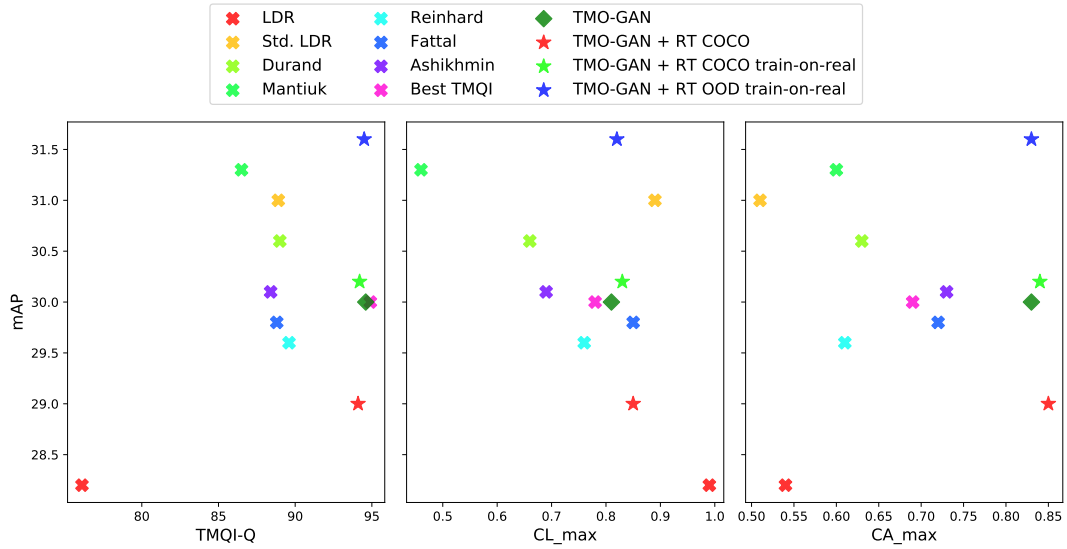


Figure 4.10: Relation between detection performance (mAP) and TMO quality metrics (TMQI-Q, CL_{max}, CA_{max}).

architectures try to remain close to the ground truth (Best TMQI per picture dataset) which keeps the variation between its different configurations less than that of between classical TMOs. Secondly, when we analyse the classical TMOs in isolation, we observe that characterics of the correlation between detection performance and each TMO quality measure differs from one another. This suggests that using a different metric instead of TMQI-Q as the ground-truth image selection when training the TMO-GAN might yield different and possibly better detection results.

CHAPTER 5

CONCLUSION

In this thesis, we investigated the use of high dynamic range content for object detection. For this end, we provided an extensive analysis of the different approaches and proposed a novel method that jointly trains a tone-mapping operator and a detector. Firstly, we consider training object detectors with tone-mapped HDR images. We systematically evaluated the performances of deep object detectors trained with (i) LDR images, (ii) HDR images and (iii) tone-mapped HDR images for scenes with different illuminations, different object categories, and objects of different sizes. Secondly, we try to improve existing GAN-based tone-mapping operators by introducing a detection network into the 2 player adversarial game and train generator and the object detection network jointly, which we call TMO-GAN. We compare the performance of this system against classical tone-mapping operators in terms of image quality and detection performance.

5.1 Concluding Remarks

To be more specific, the following research problems were studied (problem statements are copied from Chapter 1) and the following answers were obtained:

- **Do deep object detectors perform better with LDR or HDR images?**

  Given the results in Chapter 3 where we compare HDR, LDR and tone-mapped LDR images; we observe that LDR images perform slightly better than raw HDR images. However, applying pre-processing (normalization and gamma correction) before feeding HDR images to the network improves the detection performance. As demonstrated in Chapter 3, HDR images obtain a detection

performance that is on-par with best tone-mapped LDR images when they are pre-processed and the detection network is trained from scratch. Additionally, as demonstrated in Chapter 4, when the detection network is pre-trained with LDR images prior to training with HDR, preprocessing still improves the performance. However, HDR still falls behind the best performing tone-mapped LDR images. Additionally, we do not observe significant difference between LDR and HDR images under different illumination conditions such as dynamic-range and luminance of the object to be detected.

- **Which tone-mapping operators are more suitable for object detectors?**

  We do observe an agreement between Chapters 3 and 4 in terms of the ranking of the TMOs for detection performance. Although the fact that we use a different datasets and selected TMOs do not exactly match, gamma-corrected HDR performs best on the Cityscapes dataset [68] (however the second best is local version of Reinhard's [41], and the difference is not significant), whereas Mantiuk [43] performs better on the OOD dataset [55], which needs further investigation.

- **Can we perform tone mapping in such a way that the performance of an object detector is maximized while simultaneously maintaining a perceptual quality for the tone-mapped images?**

  By jointly training the detection and tone-mapping objectives we are able to improve detection performance (although the difference between the second best is not very significant) while maintaining a quite good tone-mapping quality in terms of the selected measure [2], which is significantly better than the classical TMOs.

5.2 Limitations and Future Work

In this thesis we do not provide an analysis of the proposed joint architecture on the Cityscapes dataset [68]. As an extension, all experiments conducted on the OOD dataset [55] can be performed on the Cityscapes dataset as well. Additionally, a possibly more suitable TMO quality metric such as CIVDM can be used for ground truth

LDR selection when training the TMO-GAN. Finally, the experiments in Chapter 3 and Chapter 4 can be performed with the same set of classical TMOs.

REFERENCES

[1] A. Rana, P. Singh, G. Valenzise, F. Dufaux, N. Komodakis, and A. Smolic, "Deep tone mapping operator for high dynamic range images," *IEEE Transactions on Image Processing*, vol. 29, pp. 1285–1298, 2019.

[2] H. Yeganeh and Z. Wang, "Objective quality assessment of tone-mapped images," *IEEE Transactions on Image Processing*, vol. 22, no. 2, pp. 657–667, 2012.

[3] T. O. Aydin, R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "Dynamic range independent image quality assessment," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, pp. 1–10, 2008.

[4] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pp. 257–266, 2002.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.

[6] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

[7] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2117–2125, 2017.

[8] H. Talebi and P. Milanfar, "Learning to resize images for computer vision tasks," *arXiv preprint arXiv:2103.09950*, 2021.

[9] L. Liu, W. Ouyang, X. Wang, P. Fieguth, J. Chen, X. Liu, and M. Pietikäinen,

"Deep learning for generic object detection: A survey," *International journal of computer vision*, vol. 128, no. 2, pp. 261–318, 2020.

[10] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, p. 100057, 2021.

[11] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010.

[12] R. Mantiuk, G. Krawczyk, R. Mantiuk, and H.-P. Seidel, "High-dynamic range imaging pipeline: perception-motivated representation of visual content," in *Human Vision and Electronic Imaging XII*, vol. 6492, p. 649212, International Society for Optics and Photonics, 2007.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[14] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International conference on machine learning*, pp. 214–223, PMLR, 2017.

[15] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. Paul Smolley, "Least squares generative adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 2794–2802, 2017.

[16] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," *arXiv preprint arXiv:1611.02163*, 2016.

[17] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," *Advances in neural information processing systems*, vol. 30, 2017.

[18] K. Roth, A. Lucchi, S. Nowozin, and T. Hofmann, "Stabilizing training of generative adversarial networks through regularization," *arXiv preprint arXiv:1705.09367*, 2017.

[19] L. Mescheder, A. Geiger, and S. Nowozin, "Which training methods for gans do actually converge?," in *International conference on machine learning*, pp. 3481–3490, PMLR, 2018.

[20] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," *arXiv preprint arXiv:1701.04862*, 2017.

[21] M. Mirza and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv:1411.1784*, 2014.

[22] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

[23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *IEEE PAMI*, vol. 39, no. 6, 2016.

[24] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2015.

[25] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

[27] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[28] J. Wang, K. Sun, T. Cheng, B. Jiang, C. Deng, Y. Zhao, D. Liu, Y. Mu, M. Tan, X. Wang, *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[29] Z. Cai and N. Vasconcelos, "Cascade r-cnn: Delving into high quality object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6154–6162, 2018.

[30] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[31] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp. 21–37, Springer, 2016.

[32] F. Dufaux, P. Le Callet, R. Mantiuk, and M. Mrak, *High dynamic range video: from acquisition, to display and applications*. Academic Press, 2016.

[33] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *ACM SIGGRAPH 2008 classes*, pp. 1–10, 2008.

[34] M. A. Robertson, S. Borman, and R. L. Stevenson, "Dynamic range improvement through multiple exposures," in *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)*, vol. 3, pp. 159–163, IEEE, 1999.

[35] T. Mertens, J. Kautz, and F. Van Reeth, "Exposure fusion," in *15th Pacific Conference on Computer Graphics and Applications (PG'07)*, pp. 382–390, IEEE, 2007.

[36] S. V. Lakshmi, J. Janet, T. Bellarmine, *et al.*, "Analysis of tone mapping operators on high dynamic range images," in *2012 Proceedings of IEEE Southeastcon*, pp. 1–6, IEEE, 2012.

[37] P. Ledda, A. Chalmers, T. Troscianko, and H. Seetzen, "Evaluation of tone mapping operators using a high dynamic range display," *ACM Transactions on Graphics (TOG)*, vol. 24, no. 3, pp. 640–648, 2005.

[38] G. Ward, "A contrast-based scalefactor for luminance display.," *Graphics Gems*, vol. 4, pp. 415–21, 1994.

[39] J. Tumblin and G. Turk, "A boundary hierarchy for detail-preserving contrast reduction," in *Proc. of ACM SIGGRAPH'99*, pp. 83–90, 1990.

[40] C. Schlick, "Quantization techniques for visualization of high dynamic range pictures," in *Photorealistic rendering techniques*, pp. 7–20, Springer, 1995.

[41] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pp. 267–276, 2002.

[42] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pp. 249–256, 2002.

[43] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "A perceptual framework for contrast processing of high dynamic range images," *ACM Transactions on Applied Perception (TAP)*, vol. 3, no. 3, pp. 286–308, 2006.

[44] P. Debevec and S. Gibson, "A tone mapping algorithm for high contrast images," in *13th eurographics workshop on rendering: Pisa, Italy. Citeseer*, Citeseer, 2002.

[45] S. N. Pattanaik, J. Tumblin, H. Yee, and D. P. Greenberg, "Time-dependent visual adaptation for fast realistic image display," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pp. 47–54, 2000.

[46] J. A. Ferwerda, S. N. Pattanaik, P. Shirley, and D. P. Greenberg, "A model of visual adaptation for realistic image synthesis," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pp. 249–258, 1996.

[47] F. Durand and J. Dorsey, "Interactive tone mapping," in *Eurographics Workshop on Rendering Techniques*, pp. 219–230, Springer, 2000.

[48] P. Ledda, L. P. Santos, and A. Chalmers, "A local model of eye adaptation for high dynamic range images," in *Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, pp. 151–160, 2004.

[49] N. Zhang, C. Wang, Y. Zhao, and R. Wang, "Deep tone mapping network in hsv color space," in *2019 IEEE Visual Communications and Image Processing (VCIP)*, pp. 1–4, IEEE, 2019.

[50] K. Panetta, L. Kezebou, V. Oludare, S. Agaian, and Z. Xia, "Tmo-net: A parameter-free tone mapping operator using generative adversarial network, and performance benchmarking on large scale hdr dataset," *IEEE Access*, vol. 9, pp. 39500–39517, 2021.

[51] C.-C. Su, R. Wang, H.-J. Lin, Y.-L. Liu, C.-P. Chen, Y.-L. Chang, and S.-C. Pei, "Explorable tone mapping operators," in *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 10320–10326, IEEE, 2021.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[53] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[54] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, vol. 88, no. 2, pp. 303–338, 2010.

[55] R. Mukherjee, M. Melo, V. Filipe, A. Chalmers, and M. Bessa, "Backward compatible object detection using hdr image content," *IEEE Access*, vol. 8, 2020.

[56] M. Weiher, "Domain adaptation of hdr training data for semantic road scene segmentation by deep learning," *Master Thesis, Technical University of Munich*, 2019.

[57] R. Mukherjee, M. Bessa, P. Melo-Pinto, and A. Chalmers, "Object detection under challenging lighting conditions using high dynamic range imagery," *IEEE Access*, vol. 9, pp. 77771–77783, 2021.

[58] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*, pp. 2642–2651, PMLR, 2017.

[59] L. Liu, M. Muelly, J. Deng, T. Pfister, and L.-J. Li, "Generative modeling for small-data object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 6073–6081, 2019.

[60] H. Rashid, M. A. Tanveer, and H. A. Khan, "Skin lesion classification using gan based data augmentation," in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 916–919, IEEE, 2019.

[61] S. Vandenhende, B. De Brabandere, D. Neven, and L. Van Gool, "A three-player gan: generating hard samples to improve classification networks," in *2019 16th International Conference on Machine Vision Applications (MVA)*, pp. 1–6, IEEE, 2019.

[62] C. D. Prakash and L. J. Karam, "It gan do better: Gan-based detection of objects on images with varying quality," *arXiv preprint arXiv:1912.01707*, 2019.

[63] J. Li, X. Liang, Y. Wei, T. Xu, J. Feng, and S. Yan, "Perceptual generative adversarial networks for small object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1222–1230, 2017.

[64] K. Debattista, T. Bashford-Rogers, E. Selmanović, R. Mukherjee, and A. Chalmers, "Optimal exposure compression for high dynamic range content," *The Visual Computer*, vol. 31, no. 6-8, pp. 1089–1099, 2015.

[65] R. Mantiuk, "Pfstools, `http://pfstools.sourceforge.net/`, v2.1.0," 2017.

[66] G. Bradski, "The OpenCV Library," *Dr. Dobb's Journal of Software Tools*, 2000.

[67] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *ICCV*, 2017.

[68] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016.

[69] K. Chen, J. Wang, J. Pang, and Y. C. et al., "Mmdetection: Open mmlab detection toolbox and benchmark," 2019.

[70] C. Michaelis, B. Mitzkus, R. Geirhos, and E. e. a. Rusak, "Benchmarking robustness in object detection: Autonomous driving when winter is coming," *arXiv:1907.07484*, 2019.

[71] G. Valenzise, F. De Simone, P. Lauga, and F. Dufaux, "Performance evaluation of objective quality metrics for hdr image compression," in *Applications of Digital Image Processing XXXVII*, vol. 9217, 2014.

[72] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241, Springer, 2015.

[73] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[74] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, pp. 5998–6008, 2017.

[75] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, and Z. Wang, "Enlightengan: Deep light enhancement without paired supervision," *IEEE Transactions on Image Processing*, vol. 30, pp. 2340–2349, 2021.

[76] Z. Shen, M. Zhang, H. Zhao, S. Yi, and H. Li, "Efficient attention: Attention with linear complexities," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3531–3539, 2021.

[77] M. Ashikhmin, "A tone mapping algorithm for high contrast images," in *Rendering Techniques*, 2002.

[78] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[79] R. Mantiuk, K. J. Kim, A. G. Rempel, and W. Heidrich, "Hdr-vdp-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions," *ACM Transactions on graphics (TOG)*, vol. 30, no. 4, pp. 1–14, 2011.