Implementing Noise2Noise for Dynamic Scenes

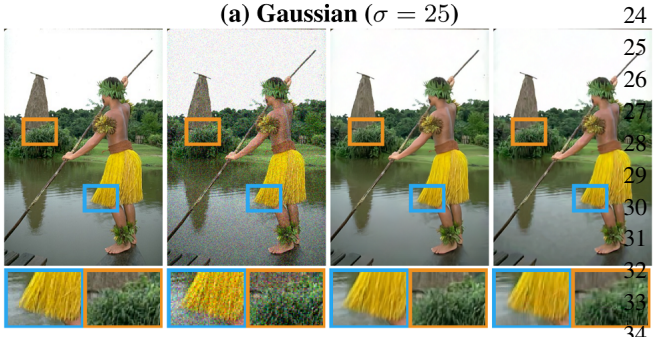
Ruijie Song M.S. [*1]

Abstract

Implementing Noise2Noise for Dynamic Scenes

1. Dataset

(a) Gaussian ($\sigma = 25$)



There's the dataset.

1.1. Codes

```
# Copyright (c) 2018, NVIDIA CORPORATION.
#
# This work is licensed under the Creative
# 4.0 International License. To view a cop
# http://creativecommons.org/licenses/by-n
# Creative Commons, PO Box 1866, Mountain

import tensorflow as tf

def parse_tfrecord_tf(record):
    features = tf.parse_single_example(rec
        'shape': tf.FixedLenFeature([3], t
        'data': tf.FixedLenFeature([], tf.
    data = tf.decode_raw(features['data'],
    return tf.reshape(data, features['shape

# [c,h,w] -> [h,w,c]
def chw_to_hwc(x):
    return tf.transpose(x, perm=[1, 2, 0])

# [h,w,c] -> [c,h,w]
def hwc_to_chw(x):
    return tf.transpose(x, perm=[2, 0, 1])

def resize_small_image(x):
    shape = tf.shape(x)
    return tf.cond(
        tf.logical_or(
            tf.less(shape[2], 256),
            tf.less(shape[1], 256)
        ),
        true_fn=lambda: hwc_to_chw(tf.image.resize_i
        false_fn=lambda: tf.cast(x, tf.float32)
    )

def random_crop_noised_clean(x, add_noise):
    cropped = tf.random_crop(resize_small_image(x),
    return (add_noise(cropped), add_noise(cropped),

def create_dataset(train_tfrecords, minibatch_size,
    print('Setting up dataset source from', train_
    buffer_mb  = 256
    num_threads = 2
    dset = tf.data.TFRecordDataset(train_tfrecords,
    dset = dset.repeat()
    buf_size = 1000
    dset = dset.prefetch(buf_size)
    dset = dset.map(parse_tfrecord_tf, num_parallel
    dset = dset.shuffle(buffer_size=buf_size)
    dset = dset.map(lambda x: random_crop_noised_cl
    dset = dset.batch(minibatch_size)
    it = dset.make_one_shot_iterator()
    return it
```

[*]Equal contribution [1]Department of Electrical and Computer Engineering, Purdue University, West Lafayette, USA. Correspondence to: Cieua Vvvvv <c.vvvvv@googol.com>, Eee Pppp <ep@eden.co.uk>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

A. Do *not* have an appendix here

Do not put content after the references. Put anything that you might normally include after the references in a separate supplementary file.

We recommend that you build supplementary material in a separate document. If you must create one PDF and cut

it up, please be careful to use a tool that doesn't alter the margins, and that doesn't aggressively rewrite the PDF file. pdftk usually works fine.

Please do not use Apple's preview to cut off supplementary material. In previous years it has altered margins, and created headaches at the camera-ready stage.