Implementation for our ICCV2021 paper: Internal Video Inpainting by Implicit Long-range Propagation

⏱ Last update: **Oct 26, 2021**

Related tags

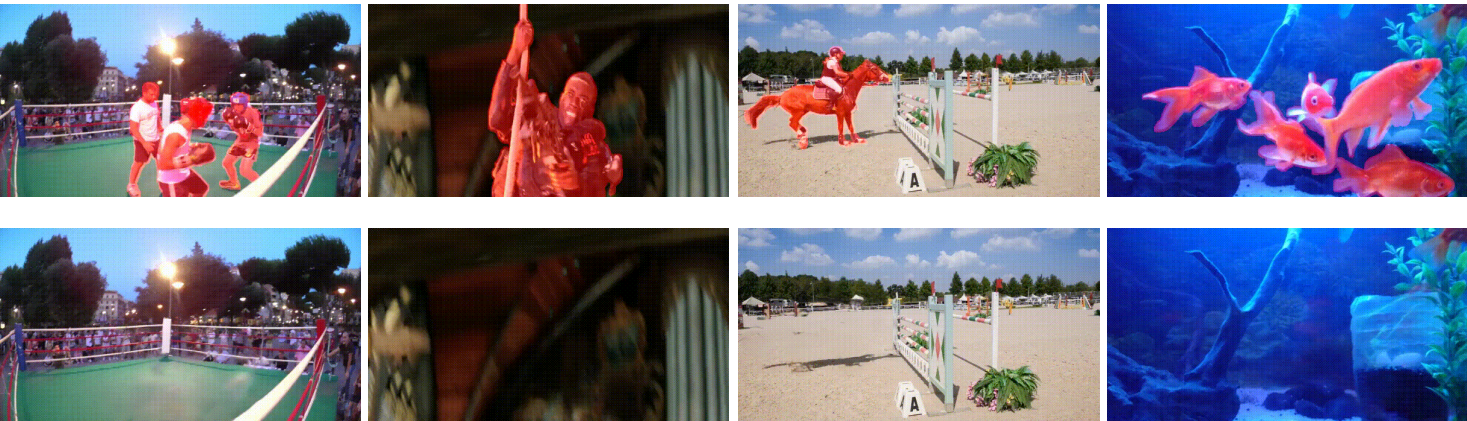computer-vision   video-processing   deeplearning   video-editing   image-inpainting   video-inpainting   object-removal

Overview

Implicit Internal Video Inpainting

Implementation for our ICCV2021 paper: Internal Video Inpainting by Implicit Long-range Propagation

paper | project website | 4K data | demo video

Introduction

Want to remove objects from a video without days of training and thousands of training videos? Try our simple but effective internal video inpainting method. The inpainting process is zero-shot and implicit, which does not need any pretraining on large datasets or optical-flow estimation. We further extend the proposed method to more challenging tasks: video object removal with limited annotated masks, and inpainting on ultra high-resolution videos (e.g., 4K videos).
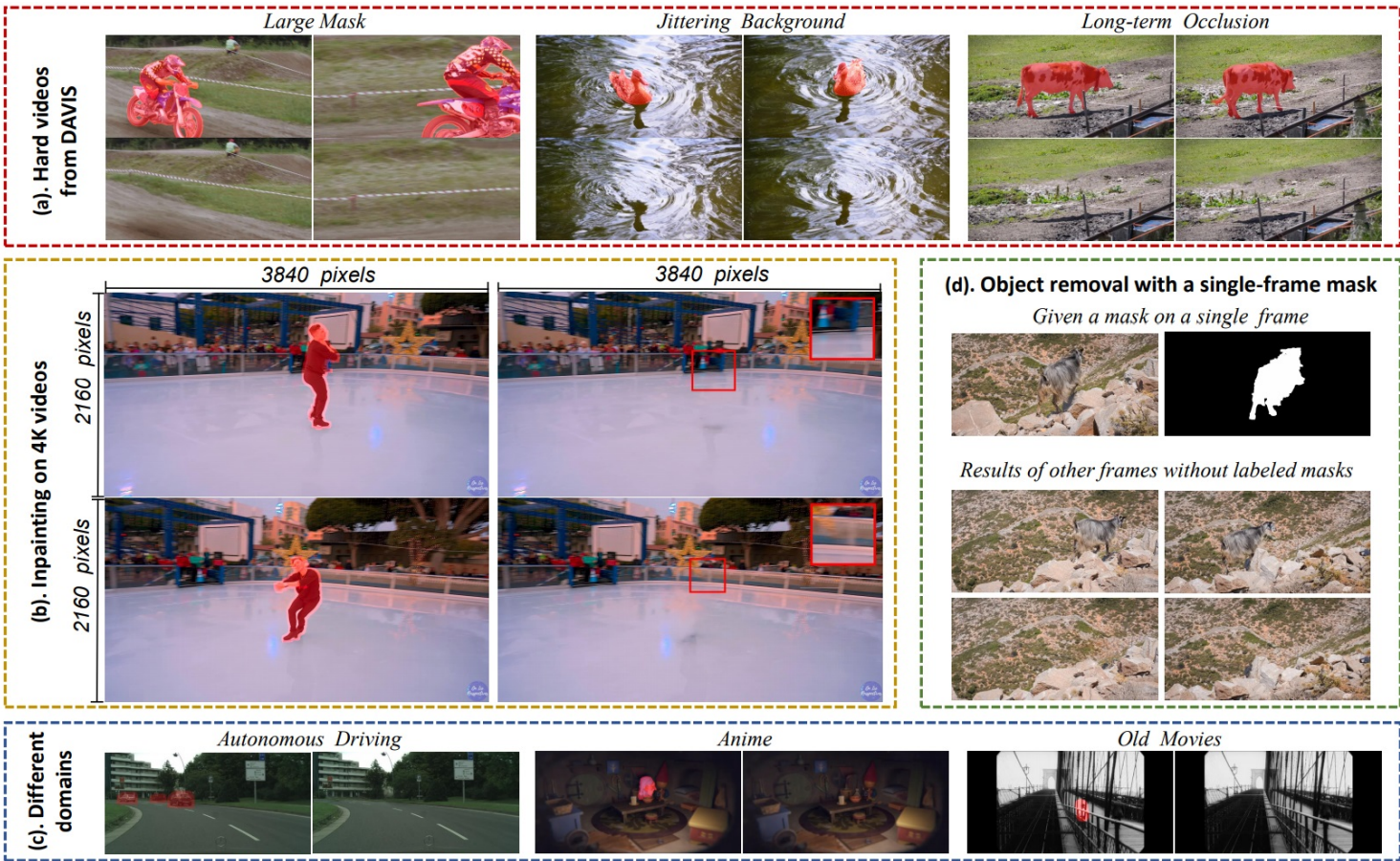

Figure 1: Ours results on (a) hard video sequences in DAVIS [29], (b) a 4K-resolution video, (c) videos from different domains, and (d) a video with a single-frame mask. Zoom in for details.

TO DO

☐ Release code for 4K video inpainting

Setup

Installation

```
git clone https://github.com/Tengfei-Wang/Implicit-Internal-Video-Inpainting.git
cd Implicit-Internal-Video-Inpainting
```

Environment

This code is based on tensorflow 2.x (tested on tensorflow 2.2, 2.4).

The environment can be simply set up by Anaconda:

```
conda create -n IIVI python=3.7
conda activate IIVI
conda install tensorflow-gpu tensorboard
pip install pyaml
pip install opencv-python
pip install tensorflow-addons
```

Or, you can also set up the environment from the provided environment.yml:

```
conda env create -f environment.yml
conda activate IIVI
```

Usage

Quick Start

We provide an example sequence 'bmx-trees' in `./inputs/` . To try our method:

```
python train.py
```

The default iterations is set to 50,000 in `config/train.yml` , and the internal learning takes ~4 hours with a single GPU. During the learning process, you can use tensorboard to check the inpainting results by:

```
tensorboard --logdir ./exp/logs
```

After the training, the final results can be saved in `./exp/results/` by:

```
python test.py
```

You can also modify 'model_restore' in `config/test.yml` to save results with different checkpoints.

Try Your Own Data

Data preprocess

Before training, we advise to dilate the object masks first to exclude some edge pixels. Otherwise, the imperfectly-annotated masks would lead to artifacts in the object removal task.

You can generate and preprocess the masks by this script:

```
python scripts/preprocess_mask.py --annotation_path inputs/annotations/bmx-trees
```

Basic training

Modify the `config/train.yml` , which indicates the video path, log path, and training iterations,etc.. The training iterations depends on the video length, and it typically takes 30,000 ~ 80,000 iterations for convergence for 100-frame videos. By default, we only use reconstruction loss for training, and it works well for most cases.

```
python train.py
```

Improve the sharpness and consistency

For some hard videos, the former training may not produce a pleasing result. You can fine-tune the trained model with another losses. To this end, modify the 'model_restore' in `config/test.yml` to the checkpoint path of basic training. Also set ambiguity_loss or stabilization_loss to True. Then fine-tune the basic checkpoint for 20,000-40,000 iterations.

```
python train.py
```

Inference

Modify the `./config/test.yml`, which indicates the video path, log path, and save path.

```
python test.py
```

Mask Propagation from A Single Frame

When you only annotate the object mask of one frame (or few frames), our method can propagate it to other frames automatically.

Modify `./config/train_mask.yml`. We typically set the training iterations to 4,000 ~ 20,000, and the learning rate to 1e-5 ~ 1e-4.

```
python train_mask.py
```

After training, modify `./config/test_mask.yml`, and then:

```
python test_mask.py
```

High-resolution Video Inpainting

Our 4K videos and mask annotations can be downloaded in 4K data.

More Results

Our results on 70 DAVIS videos (including failure cases) can be found here for your reference :)
If you need the PNG version of our uncompressed results, please contact the authors.

Citation

If you find this work useful for your research, please cite:

```
@inproceedings{ouyang2021video,
  title={Internal Video Inpainting by Implicit Long-range Propagation},
  author={Ouyang, Hao and Wang, Tengfei and Chen, Qifeng},
  booktitle={International Conference on Computer Vision (ICCV) },
  year={2021}
}
```

If you are also interested in the image inpainting or internal learning, this paper can be also helpful :)

```
@inproceedings{wang2021image,
  title={Image Inpainting with External-internal Learning and Monochromic Bottleneck},
  author={Wang, Tengfei and Ouyang, Hao and Chen, Qifeng},
  booktitle={Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition},
  pages={5120--5129},
  year={2021}
}
```

Contact

Please send emails to Hao Ouyang or Tengfei Wang if there is any question

ⓘ

AI Powered Photo Editing

Ad Topaz Labs

The Best 124 Python 3d-animation Libraries

pythonrepo.com

Hindawi Open Access Publishing

Ad Hindawi

Just playing with getting CLIP Guided Diffusion running...

pythonrepo.com

Four Seasons Maldives

Ad Four Seasons

Implementation of

pythonrepo.com

The Best 741 Python 3d-human-motion Libraries

pythonrepo.com

DeepFaceL Deep Fake

pythonrepo.com

Issues

About the pipeline

Hi, thanks for your released code.
It takes too long to train&test a video, will it possible to fastly test any input? The example "bmx-trees" takes several hours to finish. Thanks.

opened by sydney0zq 💬 5

Questoin about mask

Hi, thx for repo! Can you explain - must I make mask for each frame from video manually? or maybe has any automation tool? thx!

opened by pavelxx1 💬 2

multi-GPUs - only using vram, not processing

Hi Tengfei Wang, such a amazing reasearch and many thanks for sharing the code. Very intersting results...
I was able to reproduce some results and really liked the work flow you created of CNN and not Oflow, seams it handles perspective shifts and background better (still playing with it). The dilate mask makes totally sense...
My question is about multi-GPU to speed up training....im doing these below:
on train.py i removed the # on mirrored_strategy = tf.distribute.MirroredStrategy() line and added # on os.environ["CUDA_VISIBLE_DEVICES"] = FLAGS.GPU_ID.
With that seams that is Training is using both GPUs, but also shows that the GPU_0 is using CUDA and processing but GPU_1 only using vram, does not seams to be using CUDA and process, only VRAM. Is that correct?
Also saw @tf.function down below, but not sure if i should remove # on those lines. Also found #dist_full_ds = mirrored_strategy, tried but seams to do the same thing on second gpu, only using vram, not processing
Is that correct behavior?
Thank you Tengfei Wang and once again, amazing research.

opened by optfx 💬 1

hello，where is the pre-trained model?

opened by lainxx 💬 1

What "Mask Propagation from A Single Frame" usage?

How much annotation files I am must provide?

Must I am provide corresponding mask for them or not?

Why annotation pictures red and green and what difference?

How much frames I am must provide for video?

Is video must be without fast object movement?

---

opened by Vadim2S 💬 1

About test

Hi, I'm a little confused about this method. If I want to test 10 different videos, should I train 10 different models? and every model needs about 4 hours? or just need to train a model which can be applied to other videos?

opened by lixixin 💬 1

---

error when using train_dist.py

when I use tensorflow 2.4： AttributeError: 'MirroredStrategy' object has no attribute 'experimental_run_v2' I need to downgrade to tensorflow 2.0
when I use tensorflow 2.0： ImportError:cannot import name 'keras_tensor' from 'tensorflow.python.keras.engine' (/home/ivdai/anaconda3/envs/IIVI/lib/python3.7/site-packages/tensorflow_core/python/keras/engine/**init**.py) I need to upgrade to tensorflow 2.4
funny

opened by Altheim 💬 1

PNG version of our uncompressed results and segmentation results

Hi,
Many thanks for publishing the code for this nice work. I am very interested in your work.
May I ask could you please share
the PNG version of results

---

the segmentation results

---

inpainting results with only the first frame segmentation mask

---

of all the videos in DAVIS dataset?
Many thanks!

opened by Huihui1002 💬 1

GPU out of memory when set ambiguity_loss or stabilization_loss to True

Hi, I am trying to run your code with ambiguity loss and stabilization loss. But I met the gpu out of memory problem.
May I ask what is the batch size you set in the experiments with ambiguity loss and stabilization loss and what kind of gpu and how many gpus are used to train the model?
Many thanks!

opened by Huihui1002 💬 1

Owner

Ph.D. candidate @ HKUST / Computer Vision

○ GitHub Repository

⌂ https://tengfei-wang.github.io/Implicit-Internal-Video-Inpainting/

AOT-GAN for High-Resolution Image Inpainting (codebase for image inpainting)

AOT-GAN for High-Resolution Image Inpainting Arxiv Paper | AOT-GAN: Aggregated Contextual Transformations for High-Resolution Image Inpainting Yanhong

★ 103   ⊙ Oct 16, 2021

---

codes for Image Inpainting with External-internal Learning and Monochromic Bottleneck

Image Inpainting with External-internal Learning and Monochromic Bottleneck This repository is for the CVPR 2021 paper: 'Image Inpainting with Externa

★ 68   ⊙ Oct 19, 2021

[Automatically erase objects in the video, such as logo, text, etc.](#)

Video-Auto-Wipe Read English Introduction：Here　　　本人不定期的基于生成技术制作一些好玩有趣的算法模型，这次带来的作品是"视频擦除"方向的应用模型，它实现的功能是自动感知到视频中我们不想看见的部分（譬如广告、水印、字幕、图标等等）然后进行擦除。由于图标擦
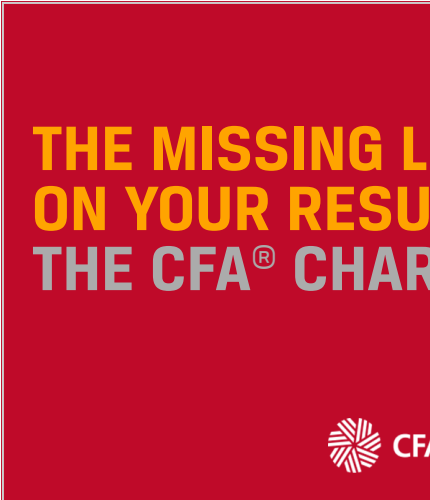
★ 68   ⊘ Oct 15, 2021

[MaRS - a recursive filtering framework that allows for truly modular multi-sensor integration](#)

The Modular and Robust State-Estimation Framework, or short, MaRS, is a recursive filtering framework that allows for truly modular multi-sensor integration
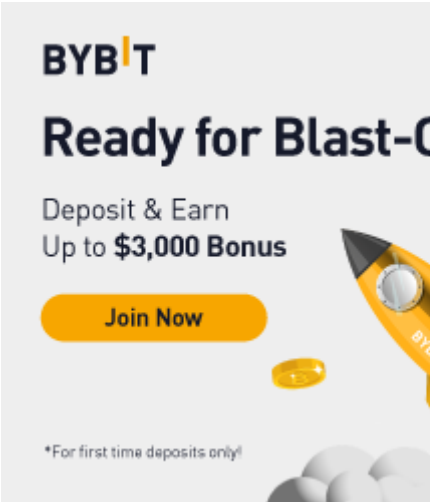
★ 97   ⊘ Oct 20, 2021

[CVPR 2021] MiVOS - Mask Propagation module. Reproduced STM (and better) with training code :star2:. Semi-supervised video object segmentation evaluation.

MiVOS (CVPR 2021) - Mask Propagation Ho Kei Cheng, Yu-Wing Tai, Chi-Keung Tang [arXiv] [Paper PDF] [Project Page] [Papers with Code] This repo impleme

★ 61   🕐 Oct 26, 2021

The pytorch implementation of the paper "text-guided neural image inpainting" at MM'2020

TDANet: Text-Guided Neural Image Inpainting, MM'2020 (Oral) MM | ArXiv This repository implements the paper "Text-Guided Neural Image Inpainting" by L

★ 30   🕐 Oct 22, 2021

CVPR 2021: "Generating Diverse Structure for Image Inpainting With Hierarchical VQ-VAE"

Diverse Structure Inpainting ArXiv | Papar | Supplementary Material | BibTex This repository is for the CVPR 2021 paper, "Generating Diverse Structure

★ 99   🕐 Oct 11, 2021

Probabilistic Tracklet Scoring and Inpainting for Multiple Object Tracking

Probabilistic Tracklet Scoring and Inpainting for Multiple Object Tracking (CVPR 2021) Pytorch implementation of the ArTIST motion model. In this repo

★ 24   🕐 Sep 20, 2021

PyTorch Implement of Context Encoders: Feature Learning by Inpainting

Context Encoders: Feature Learning by Inpainting This is the Pytorch implement of CVPR 2016 paper on Context Encoders 1) Semantic Inpainting Demo Inst

★ 293   🕐 Oct 20, 2021

This is the official repo for TransFill: Reference-guided Image Inpainting by Merging Multiple Color and Spatial Transformations at CVPR'21. According to some product reasons, we are not planning to release the training/testing codes and models. However, we will release the dataset and the scripts to prepare the dataset.

TransFill-Reference-Inpainting This is the official repo for TransFill: Reference-guided Image Inpainting by Merging Multiple Color and Spatial Transf

★ 49   ⊘ Oct 17, 2021

Implementation for "Seamless Manga Inpainting with Semantics Awareness" (SIGGRAPH 2021 issue)

Seamless Manga Inpainting with Semantics Awareness [SIGGRAPH 2021](To appear) | Project Website | BibTex Introduction: Manga inpainting fills up the d

★ 50   ⊘ Oct 11, 2021

A library for preparing, training, and evaluating scalable deep learning hybrid recommender systems using PyTorch.

collie Collie is a library for preparing, training, and evaluating implicit deep learning hybrid recommender systems, named after the Border Collie do

★ 80   ⊘ Oct 12, 2021

[CVPR 2021] Modular Interactive Video Object Segmentation: Interaction-to-Mask, Propagation and Difference-Aware Fusion

[CVPR 2021] Modular Interactive Video Object Segmentation: Interaction-to-Mask, Propagation and Difference-Aware Fusion

★ 217   ⊘ Oct 19, 2021

Awesome Long-Tailed Learning

Awesome Long-Tailed Learning This repo pays specially attention to the long-tailed distribution, where labels follow a long-tailed or power-law distri

★ 129   ⊘ Oct 22, 2021

patchmatch和patchmatchstereo算法的python实现

patchmatch patchmatch以及patchmatchstereo算法的python版实现 patchmatch参考 github patchmatchstereo参考李迎松博士的c++版代码 由于patchmatchstereo没有做任何优化，并且是python的代码，主要是方便解析算

★ 6   ⊘ Sep 19, 2021

A curated (most recent) list of resources for Learning with Noisy Labels

A curated (most recent) list of resources for Learning with Noisy Labels

★ 99   ⊘ Oct 17, 2021

official Pytorch implementation of ICCV 2021 paper FuseFormer: Fusing Fine-Grained Information in Transformers for Video Inpainting.

FuseFormer: Fusing Fine-Grained Information in Transformers for Video Inpainting By Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu

★ 16   ⊘ Oct 21, 2021

《Geo Word Clouds》 paper implementation

《Geo Word Clouds》 paper implementation

★ 1   ⊘ Oct 15, 2021

This repository contains the code for the CVPR 2020 paper "Differentiable Volumetric Rendering: Learning Implicit 3D Representations without 3D Supervision"

Differentiable Volumetric Rendering Paper | Supplementary | Spotlight Video | Blog Entry | Presentation | Interactive Slides | Project Page This repos

★ 513   ⊘ Oct 22, 2021

PythonRepo   2021.PythonRepo

f t in P ◎

★ 513   ⊘ Oct 22, 2021

PythonRepo   2021.PythonRepo

f t in P ◎