4/11/2022 8:01:28 PM

GitHub - openvinotoolkit/anoma ×

github.com/openvinotoolkit/anomalib

Product Team Enterprise Explore Marketplace Pricing Search / Sign in Sign up

openvinotoolkit / anomalib Public

Notifications | Fork 88 | Star 743

Code | Issues 22 | Pull requests 7 | Discussions | Actions | Projects | Wiki | Security | Insights

development | 28 branches | 7 tags | Go to file | Code

dreaquil WIP Fix/m1 mac (#158) ... ✓ b785f7d 8 hours ago 137 commits

.devcontainer	Initial Commit	5 months ago
.github	Fix docs workflow (#200)	5 days ago
anomalib	WIP Fix/m1 mac (#158)	8 hours ago
docs	Updated documentation for development on Docker (#217)	9 hours ago
requirements	WIP Fix/m1 mac (#158)	8 hours ago
tests	Minor fixes: Update callbacks to AnomalyModule (#208)	14 hours ago
tools	Directory streaming (#210)	10 hours ago
.dockerignore	Initial Commit	5 months ago
.gitattributes	Initial Commit	5 months ago
.gitignore	Feature/data/btad (#120)	28 days ago
.pre-commit-config.yaml	Remove `freia` as dependency and include it in `anomalib/models/com...	5 days ago
CHANGELOG.md	Updated changelog (#49)	4 months ago
CITATION.cff	Update `anomalib` version and requirements (#163)	17 days ago
CODE_OF_CONDUCT.md	Create CODE_OF_CONDUCT.md (#86)	2 months ago
CONTRIBUTING.md	Refactor (#87)	2 months ago
Dockerfile	fixed Dockerfile (#172)	6 days ago
LICENSE	Update LICENSE	4 months ago
MANIFEST.in	Organize `anomalib` dependencies (#32)	4 months ago
README.md	Added MVTec license to the repo (#177)	12 days ago
pyproject.toml	CI: Nightly Build (#66)	2 months ago
setup.py	Update `anomalib` version and requirements (#163)	17 days ago
third-party-programs.txt	Remove `freia` as dependency and include it in `anomalib/models/com...	5 days ago
tox.ini	Remove `freia` as dependency and include it in `anomalib/models/com...	5 days ago

About

An anomaly detection library comprising state-of-the-art algorithms and features such as experiment management, hyper-parameter optimization, and edge inference.

openvinotoolkit.github.io/anomalib/

unsupervised-learning anomaly-detection neural-network-compression openvino anomaly-segmentation anomaly-localization

Readme
Apache-2.0 License
Code of conduct
Cite this repository
743 stars
14 watching
88 forks

Releases 6

v.0.2.5 Latest
17 days ago
+ 5 releases

Packages

No packages published

Contributors 11

Languages

Python 99.0% Other 1.0%

README.md

ANOMALiB

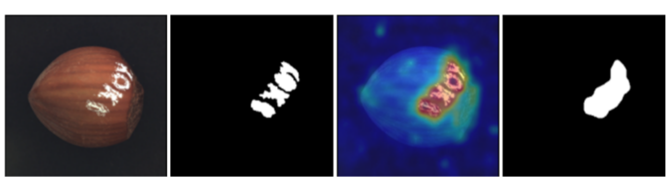
A library for benchmarking, developing and deploying deep learning anomaly detection algorithms

Key Features • Getting Started • Docs • License

python 3.8+ | pytorch 1.8.1+ | openvino 2021.4.2 | code style black | Nightly-regression Test failing | Pre-merge Checks passing | Build Docs passing

Introduction

Anomalib is a deep learning library that aims to collect state-of-the-art anomaly detection algorithms for benchmarking on both public and private datasets. Anomalib provides several ready-to-use implementations of anomaly detection algorithms described in the recent literature, as well as a set of tools that facilitate the development and implementation of custom models. The library has a strong focus on image-based anomaly detection, where the goal of the algorithm is to identify anomalous images, or anomalous pixel regions within images in a dataset. Anomalib is constantly updated with new algorithms and training/inference extensions, so keep checking!



Key features:

- The largest public collection of ready-to-use deep learning anomaly detection algorithms and benchmark datasets.
- PyTorch Lightning based model implementations to reduce boilerplate code and limit the implementation efforts to the bare essentials.
- All models can be exported to OpenVINO Intermediate Representation (IR) for accelerated inference on intel hardware.
- A set of inference tools for quick and easy deployment of the standard or custom anomaly detection models.

Getting Started

To get an overview of all the devices where `anomalib` as been tested thoroughly, look at the Supported Hardware section in the documentation.

PyPI Install

You can get started with `anomalib` by just using pip.

```
pip install anomalib
```

> *NOTE:* Due to ongoing fast pace of development, we encourage you to use editable install until we release v0.2.5.

Local Install

It is highly recommended to use virtual environment when installing anomalib. For instance, with anaconda, `anomalib` could be installed as,

```
yes | conda create -n anomalib_env python=3.8
conda activate anomalib_env
git clone https://github.com/openvinotoolkit/anomalib.git
cd anomalib
pip install -e .
```

Training

By default `python tools/train.py` runs PADIM model on `leather` category from the MVTec AD (CC BY-NC-SA 4.0) dataset.

```
python tools/train.py    # Train PADIM on MVTec AD leather
```

Training a model on a specific dataset and category requires further configuration. Each model has its own configuration file, `config.yaml` , which contains data, model and training configurable parameters. To train a specific model on a specific dataset and category, the config file is to be provided:

```
python tools/train.py --model_config_path <path/to/model/config.yaml>
```

For example, to train PADIM you can use

```
python tools/train.py --model_config_path anomalib/models/padim/config.yaml
```

Alternatively, a model name could also be provided as an argument, where the scripts automatically finds the corresponding config file.

```
python tools/train.py --model padim
```

where the currently available models are:

- CFlow
- PatchCore
- PADIM
- STFPM
- DFM
- DFKDE
- GANomaly

Custom Dataset

It is also possible to train on a custom folder dataset. To do so, `data` section in `config.yaml` is to be modified as follows:

```
dataset:
  name: <name-of-the-dataset>
  format: folder
  path: <path/to/folder/dataset>
  normal: normal # name of the folder containing normal images.
  abnormal: abnormal # name of the folder containing abnormal images.
  task: segmentation # classification or segmentation
  mask: <path/to/mask/annotations> #optional
  extensions: null
  split_ratio: 0.2  # ratio of the normal images that will be used to create a test split
  seed: 0
  image_size: 256
  train_batch_size: 32
  test_batch_size: 32
  num_workers: 8
  transform_config: null
  create_validation_set: true
  tiling:
    apply: false
    tile_size: null
    stride: null
    remove_border_count: 0
    use_random_tiling: False
    random_tile_count: 16
```

Inference

Anomalib contains several tools that can be used to perform inference with a trained model. The script in `tools/inference` contains an example of how the inference tools can be used to generate a prediction for an input image.

If the specified weight path points to a PyTorch Lightning checkpoint file (`.ckpt`), inference will run in PyTorch. If the path points to an ONNX graph (`.onnx`) or OpenVINO IR (`.bin` or `.xml`), inference will run in OpenVINO.

The following command can be used to run inference from the command line:

```
python tools/inference.py \
    --model_config_path <path/to/model/config.yaml> \
    --weight_path <path/to/weight/file> \
    --image_path <path/to/image>
```

As a quick example:

```
python tools/inference.py \
    --model_config_path anomalib/models/padim/config.yaml \
    --weight_path results/padim/mvtec/bottle/weights/model.ckpt \
```

```
    --image_path datasets/MVTec/bottle/test/broken_large/000.png
```

If you want to run OpenVINO model, ensure that `openvino` `apply` is set to `True` in the respective model `config.yaml`.

```
optimization:
  openvino:
    apply: true
```

Example OpenVINO Inference:

```
python tools/inference.py \
    --model_config_path  \
    anomalib/models/padim/config.yaml  \
    --weight_path  \
    results/padim/mvtec/bottle/compressed/compressed_model.xml  \
    --image_path  \
    datasets/MVTec/bottle/test/broken_large/000.png  \
    --meta_data  \
    results/padim/mvtec/bottle/compressed/meta_data.json
```

> Ensure that you provide path to `meta_data.json` if you want the normalization to be applied correctly.

Datasets

`anomalib` supports MVTec AD (CC BY-NC-SA 4.0) and BeanTech (CC-BY-SA) for benchmarking and `folder` for custom dataset training/inference.

MVTec AD Dataset

MVTec AD dataset is one of the main benchmarks for anomaly detection, and is released under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License (CC BY-NC-SA 4.0).

Image-Level AUC

Model		Avg	Carpet	Grid	Leather	Tile	Wood	Bottle	Cable	Capsule
PatchCore	**Wide ResNet-50**	**0.980**	0.984	0.959	1.000	**1.000**	0.989	1.000	**0.990**	**0.982**
PatchCore	ResNet-18	0.973	0.970	0.947	1.000	0.997	0.997	1.000	0.986	0.965
CFlow	Wide ResNet-50	0.962	0.986	0.962	**1.0**	0.999	**0.993**	**1.0**	0.893	0.945
PaDiM	Wide ResNet-50	0.950	**0.995**	0.942	1.0	0.974	**0.993**	0.999	0.878	0.927
PaDiM	ResNet-18	0.891	0.945	0.857	0.982	0.950	0.976	0.994	0.844	0.901
STFPM	Wide ResNet-50	0.876	0.957	0.977	0.981	0.976	0.939	0.987	0.878	0.732
STFPM	ResNet-18	0.893	0.954	**0.982**	0.989	0.949	0.961	0.979	0.838	0.759
DFM	Wide ResNet-50	0.891	0.978	0.540	0.979	0.977	0.974	0.990	0.891	0.931
DFM	ResNet-18	0.894	0.864	0.558	0.945	0.984	0.946	0.994	0.913	0.871
DFKDE	Wide ResNet-50	0.774	0.708	0.422	0.905	0.959	0.903	0.936	0.746	0.853
DFKDE	ResNet-18	0.762	0.646	0.577	0.669	0.965	0.863	0.951	0.751	0.698

Pixel-Level AUC

Model		Avg	Carpet	Grid	Leather	Tile	Wood	Bottle	Cable	Capsule
PatchCore	**Wide ResNet-50**	**0.980**	0.988	0.968	0.991	0.961	0.934	0.984	**0.988**	**0.988**
PatchCore	ResNet-18	0.976	0.986	0.955	0.990	0.943	0.933	0.981	0.984	0.986
CFlow	Wide ResNet-50	0.971	0.986	0.968	0.993	**0.968**	0.924	0.981	0.955	**0.988**
PaDiM	Wide ResNet-50	0.979	**0.991**	0.970	0.993	0.955	**0.957**	**0.985**	0.970	**0.988**
PaDiM	ResNet-18	0.968	0.984	0.918	**0.994**	0.934	0.947	0.983	0.965	0.984
STFPM	Wide ResNet-50	0.903	0.987	**0.989**	0.980	0.966	0.956	0.966	0.913	0.956
STFPM	ResNet-18	0.951	0.986	0.988	0.991	0.946	0.949	0.971	0.898	0.962

Image F1 Score

Model		Avg	Carpet	Grid	Leather	Tile	Wood	Bottle	Cable	Capsule
PatchCore	**Wide ResNet-50**	**0.976**	0.971	0.974	**1.000**	**1.000**	0.967	**1.000**	0.968	**0.982**
PatchCore	ResNet-18	0.970	0.949	0.946	**1.000**	0.98	**0.992**	**1.000**	**0.978**	0.969
CFlow	Wide ResNet-50	0.944	0.972	0.932	**1.0**	0.988	0.967	**1.0**	0.832	0.939
PaDiM	Wide ResNet-50	0.951	**0.989**	0.930	**1.0**	0.960	0.983	0.992	0.856	**0.982**
PaDiM	ResNet-18	0.916	0.930	0.893	0.984	0.934	0.952	0.976	0.858	0.960
STFPM	Wide ResNet-50	0.926	0.973	0.973	0.974	0.965	0.929	0.976	0.853	0.920
STFPM	ResNet-18	0.932	0.961	**0.982**	0.989	0.930	0.951	0.984	0.819	0.918
DFM	Wide ResNet-50	0.918	0.960	0.844	0.990	0.970	0.959	0.976	0.848	0.944
DFM	ResNet-18	0.919	0.895	0.844	0.926	0.971	0.948	0.977	0.874	0.935
DFKDE	Wide ResNet-50	0.875	0.907	0.844	0.905	0.945	0.914	0.946	0.790	0.914
DFKDE	ResNet-18	0.872	0.864	0.844	0.854	0.960	0.898	0.942	0.793	0.908

Reference

If you use this library and love it, use this to cite it 🤗

```
@misc{anomalib,
      title={Anomalib: A Deep Learning Library for Anomaly Detection},
      author={Samet Akcay and
              Dick Ameln and
              Ashwin Vaidya and
              Barath Lakshmanan and
              Nilesh Ahuja and
              Utku Genc},
      year={2022},
      eprint={2202.08341},
      archivePrefix={arXiv},
      primaryClass={cs.CV}
}
```