Print to PDF

FastFlow

Contents

FastFlow Lightning Model Implementation.

https://arxiv.org/abs/2111.07677

class **anomalib.models.image.fastflow.lightning_model.Fastflow(*backbone='resnet18', pre_trained=True, flow_steps=8, conv3x3_only=False, hidden_ratio=1.0*)**

Bases: `AnomalyModule`

PL Lightning Module for the FastFlow algorithm.

Parameters:

- **backbone** (*str*) – Backbone CNN network Defaults to `resnet18`.
- **pre_trained** (*bool, optional*) – Boolean to check whether to use a pre_trained backbone. Defaults to `True`.
- **flow_steps** (*int, optional*) – Flow steps. Defaults to `8`.
- **conv3x3_only** (*bool, optinoal*) – Use only conv3x3 in fast_flow model. Defaults to `False`.
- **hidden_ratio** (*float, optional*) – Ratio to calculate hidden var channels. Defaults to ``1.0`.

configure_optimizers()

Configure optimizers for each decoder.

Returns:

Adam optimizer for each decoder

Return type:

Optimizer

property* learning_type: *LearningType

Return the learning type of the model.

Returns:

Learning type of the model.

Return type:

LearningType

property* trainer_arguments: *dict[str, Any]

Return FastFlow trainer arguments.

training_step(*batch*, **args*, ***kwargs*)

Perform the training step input and return the loss.

Parameters:

- **(batch** (*batch*) – dict[str, str | torch.Tensor]): Input batch
- **args** – Additional arguments.
- **kwargs** – Additional keyword arguments.

Returns:

Dictionary containing the loss value.

Return type:

STEP_OUTPUT

validation_step(*batch*, **args*, ***kwargs*)

Perform the validation step and return the anomaly map.

Parameters:

- **batch** (*dict[str, str | torch.Tensor]*) – Input batch
- **args** – Additional arguments.
- **kwargs** – Additional keyword arguments.

Returns:

batch dictionary containing anomaly-maps.

Return type:

STEP_OUTPUT | None

FastFlow Torch Model Implementation.

class **anomalib.models.image.fastflow.torch_model.FastflowModel(*input_size*, *backbone*, *pre_trained=True*, *flow_steps=8*, *conv3x3_only=False*, *hidden_ratio=1.0*)**

Bases: `Module`

FastFlow.

Unsupervised Anomaly Detection and Localization via 2D Normalizing Flows.

Parameters:

- **input_size** (*tuple[int, int]*) – Model input size.
- **backbone** (*str*) – Backbone CNN network
- **pre_trained** (*bool, optional*) – Boolean to check whether to use a pre_trained backbone. Defaults to `True`.
- **flow_steps** (*int, optional*) – Flow steps. Defaults to `8`.
- **conv3x3_only** (*bool, optinoal*) – Use only conv3x3 in fast_flow model. Defaults to `False`.
- **hidden_ratio** (*float, optional*) – Ratio to calculate hidden var channels. Defaults to `1.0`.

Raises:

ValueError – When the backbone is not supported.

forward(*input_tensor*)

Forward-Pass the input to the FastFlow Model.

Parameters:

input_tensor (*torch.Tensor*) – Input tensor.

Returns:

During training, return

(hidden_variables, log-of-the-jacobian-determinants). During the validation/test, return the anomaly map.

Return type:

Tensor | list[torch.Tensor] | tuple[list[torch.Tensor]]

Loss function for the FastFlow Model Implementation.

class anomalib.models.image.fastflow.loss.**FastflowLoss**(**args*, ***kwargs*)

Bases: `Module`

FastFlow Loss.

forward(*hidden_variables*, *jacobians*)

Calculate the Fastflow loss.

Parameters:

- **hidden_variables** (*list[torch.Tensor]*) – Hidden variables from the fastflow model. f: X -> Z
- **jacobians** (*list[torch.Tensor]*) – Log of the jacobian determinants from the fastflow model.

Returns:

Fastflow loss computed based on the hidden variables and the log of the Jacobians.

Return type:

Tensor

FastFlow Anomaly Map Generator Implementation.

class

anomalib.models.image.fastflow.anomaly_map.AnomalyMapGenerator(*input_size*)

Bases: `Module`

Generate Anomaly Heatmap.

Parameters:

input_size (*ListConfig | tuple*) – Input size.

forward(*hidden_variables*)

Generate Anomaly Heatmap.

This implementation generates the heatmap based on the flow maps computed from the normalizing flow (NF) FastFlow blocks. Each block yields a flow map, which overall is stacked and averaged to an anomaly map.

Parameters:

hidden_variables (*list[torch.Tensor]*) – List of hidden variables from each NF FastFlow block.

Returns:

Anomaly Map.

Return type:

Tensor

Previous Efficient AD

Next GANomaly