Deep Learning Pipeline for Automated Visual Moth Monitoring: Insect Localization and Species Classification

Dimitri Korsch, Paul Bodesheim, Joachim Denzler

27. September 2021

FRIEDRICH-SCHILLER-UNIVERSITÄT JENA
Fakultät für Mathematik und Informatik
Computer Vision Group

AMMOD Automated Biodiversity Monitoring

Automated Multisensor Stations for Monitoring of BioDiversity

AMMOD Automated Biodiversity Monitoring

SPONSORED BY THE

Federal Ministry of Education and Research

FONA

Research for Sustainability

Graphic by V.ALTOUNIAN/SCIENCE.
From "Where have all the insects gone?"
by Gretchen Vogel, SCIENCE
May 10, 2017 (doi:10.1126/science.aal1160).
Used with permission from AAAS.

FRIEDRICH-SCHILLER-UNIVERSITÄT JENA
Fakultät für Mathematik und Informatik

Computer Vision Group

Automated Visual Moth Monitoring

"Moth" = nocturnal Lepidoptera

Moth Scanner

Graphic by V.ALTOUNIAN/SCIENCE.
From "Where have all the insects gone?"
by Gretchen Vogel, SCIENCE
May 10, 2017 (doi:10.1126/science.aal1160).
Used with permission from AAAS.

Automated Visual Moth Monitoring - Example Images

"Moth" = nocturnal Lepidoptera

Automated Visual Moth Monitoring - Challenges

"Moth" = nocturnal Lepidoptera

1. Detection
2. Species Classification

Graphic by V.ALTOUNIAN/SCIENCE.
From "Where have all the insects gone?"
by Gretchen Vogel, SCIENCE
May 10, 2017 (doi:10.1126/science.aal1160).
Used with permission from AAAS.

Outline

1. Dataset
2. Methods
   a. Detection
   b. Classification
3. Experiments

Dataset

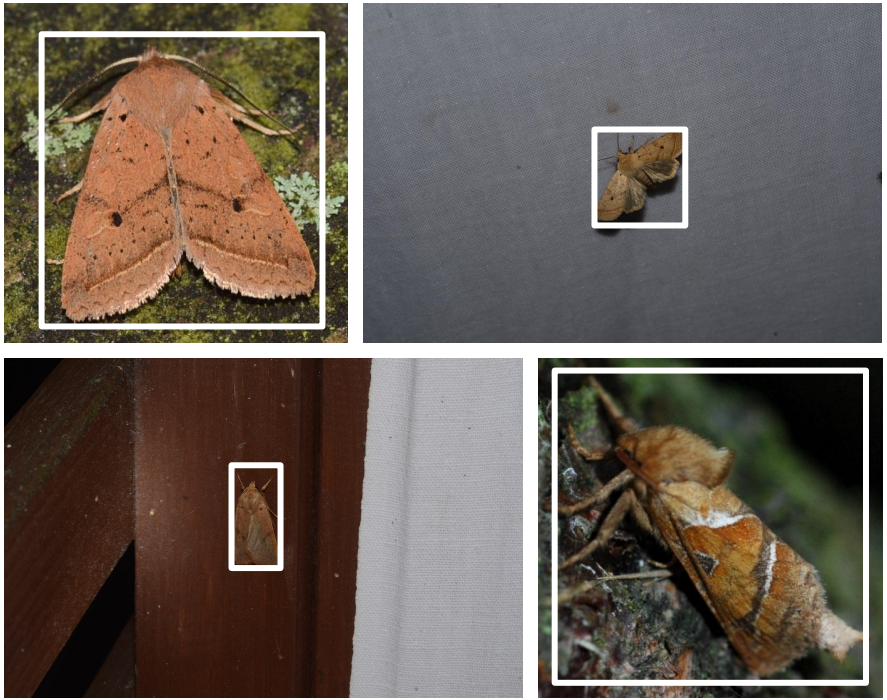
Dataset - EU-Moths [1]




- 200 moth species
- 2200 images
  11 images per species

[1] https://www.inf-cv.uni-jena.de/eu_moths_dataset

Dataset - EU-Moths [1]



- 200 moth species
- 2200 images
  11 images per species
- Manual BBox Annotations

[1] https://www.inf-cv.uni-jena.de/eu_moths_dataset

Methods

Moth Detector

Single Shot Multibox Detector (SSD)
[Liu et al., 2016]

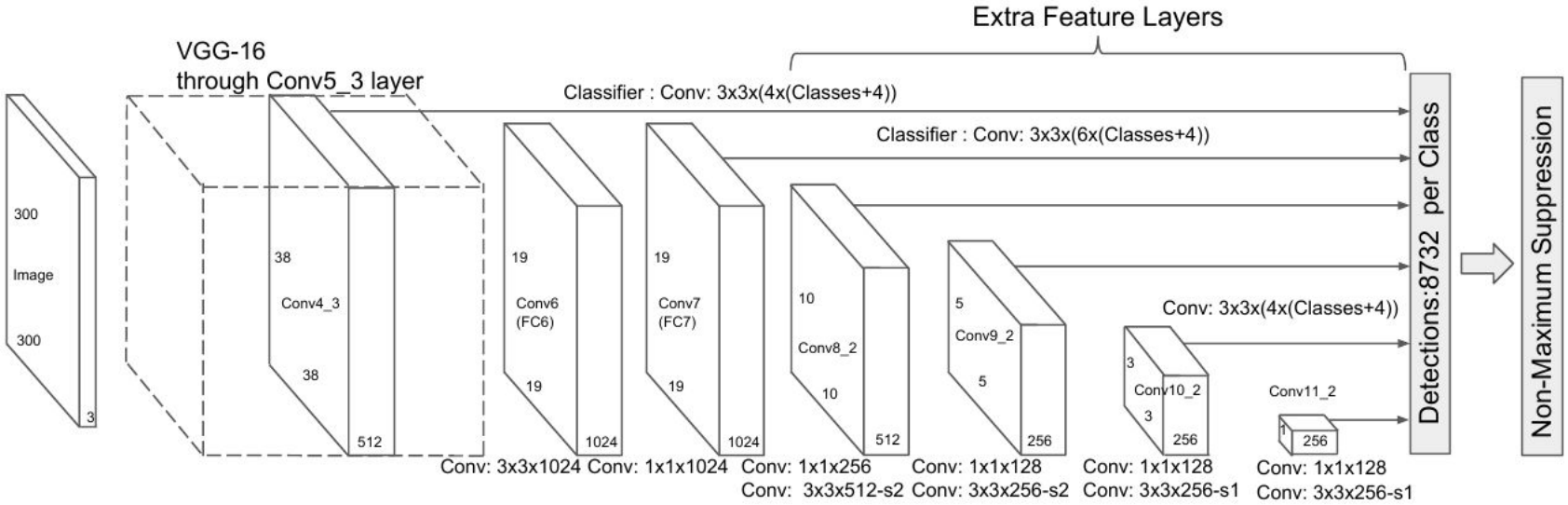
- Single Pass
- MultiBox Detection and Classification
- ~ 46 FPS @ 300x300 px

Basic idea:

- multiple feature maps
- for every local feature:
  - set of prior boxes (8732)
  - predict **offsets** for each box
  - predict **class confidences**

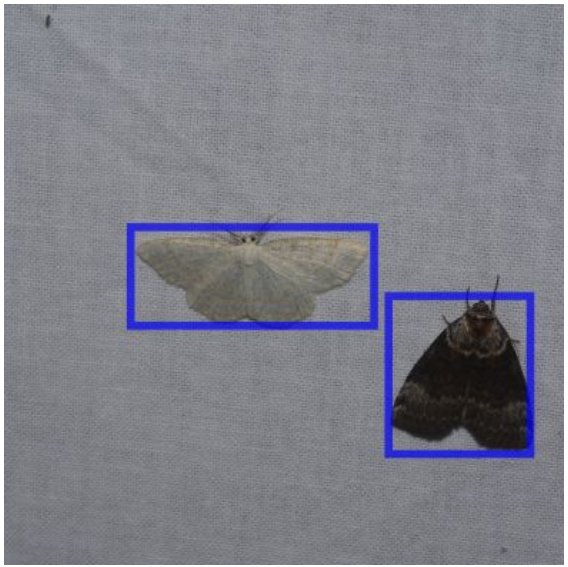
Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "SSD: Single shot multibox detector." ECCV. 2016

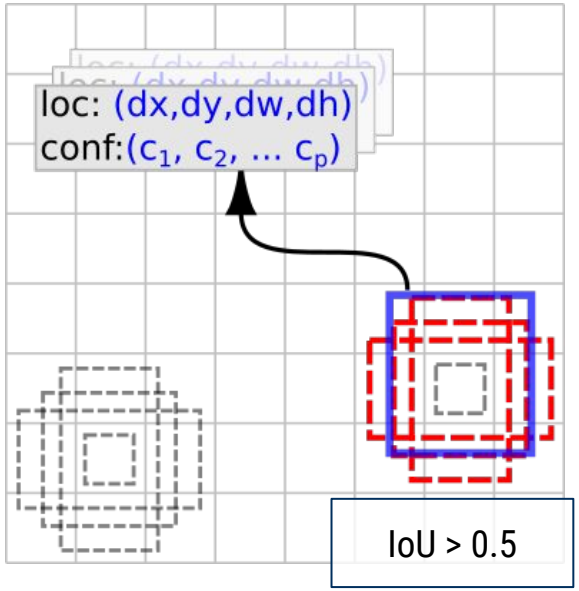
Moth Detector: (1) multi-scale feature maps

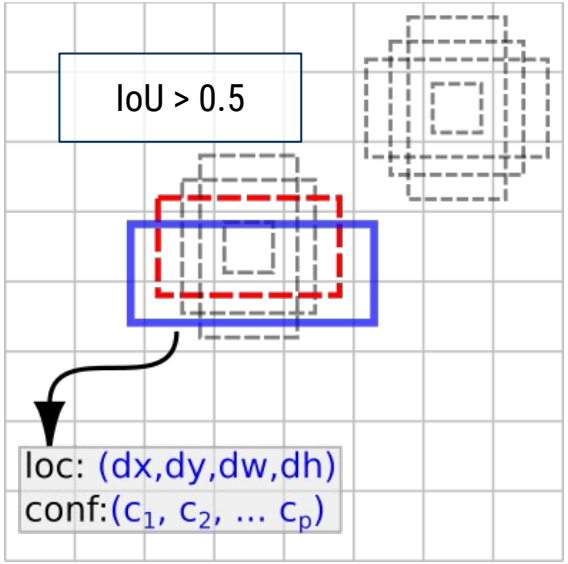


Simonyan, Karen, and Andrew Zisserman.
"Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556. 2014.

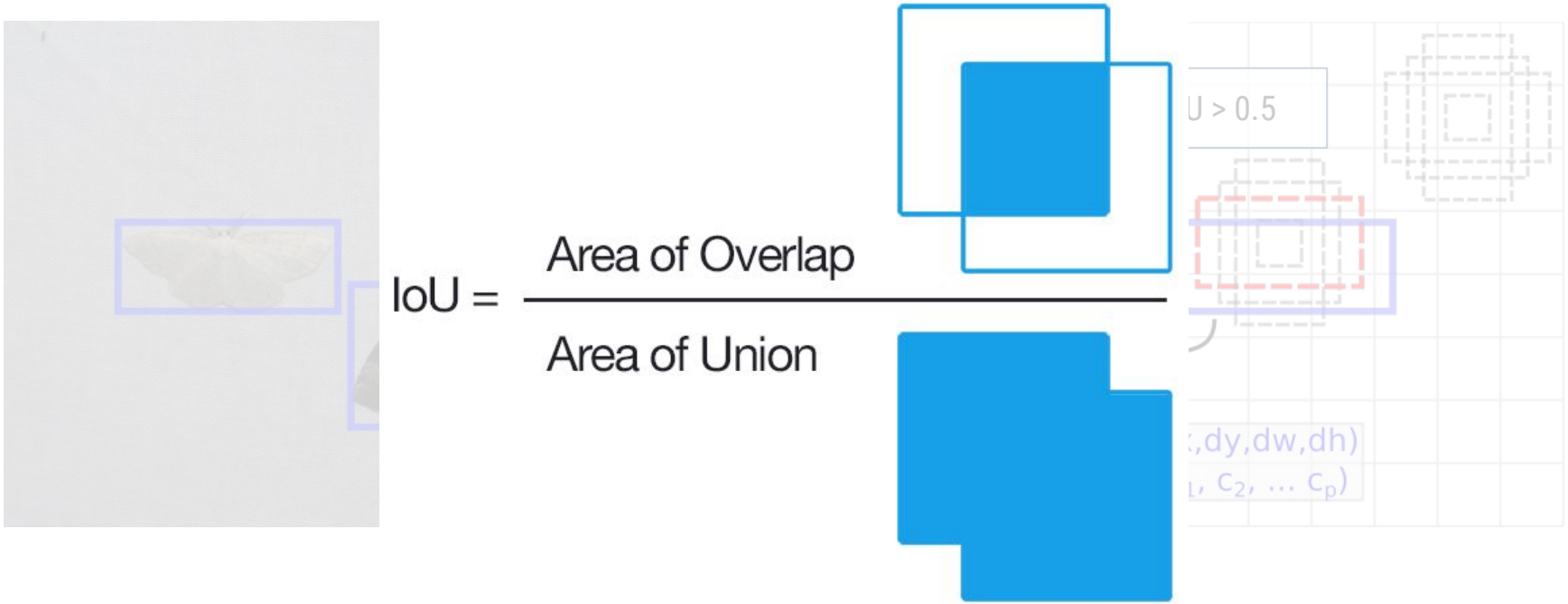
Moth Detector: (2) prior bounding boxes



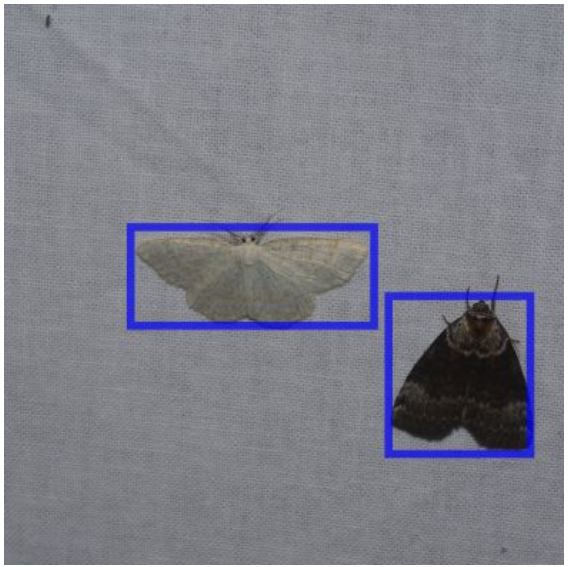


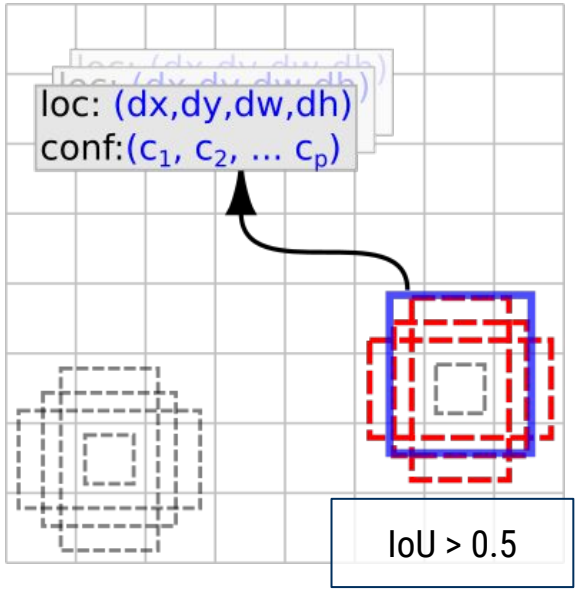


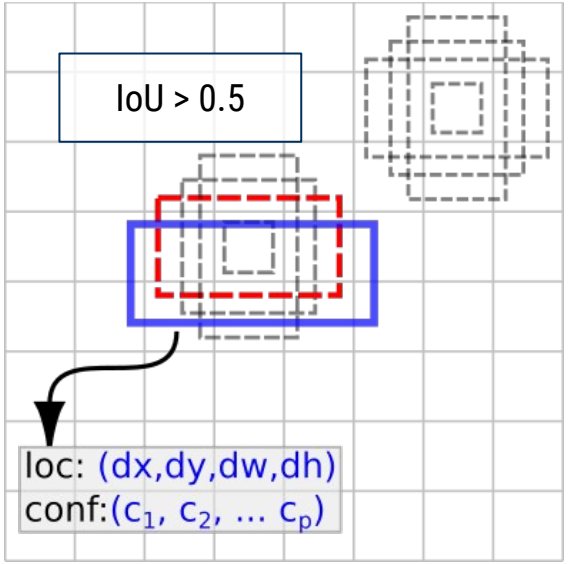
Moth Detector: (2) prior bounding boxes



Moth Detector: (2) prior bounding boxes

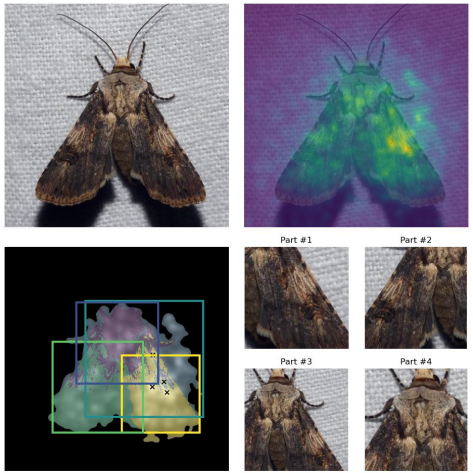


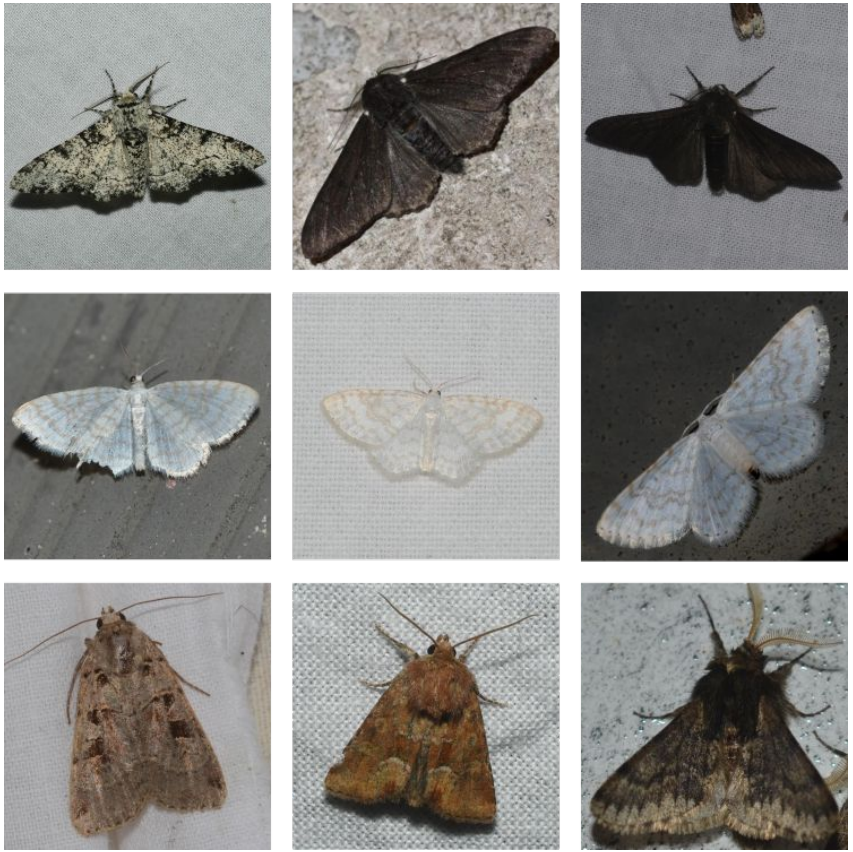




Species Classification

1. Classification of the image
2. Part-based enhancement:





Species Classification - Unsupervised Part Estimation [Korsch et al., 2019]

CS-Parts (**c**lassification-**s**pecific)

1. Initial classification

$d \in \mathcal{D}$

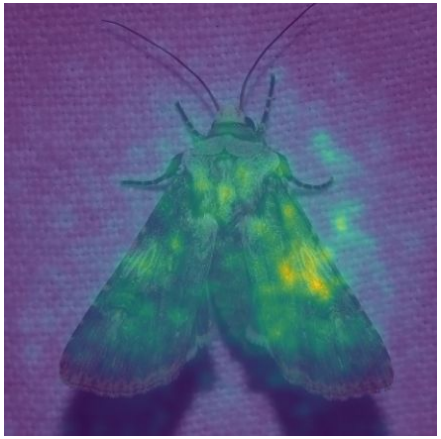
Korsch, Dimitri, Paul Bodesheim, Joachim Denzler. "Classification-Specific Parts for Improving Fine-Grained Visual Categorization" GCPR. 2019

Species Classification - Unsupervised Part Estimation [Korsch et al., 2019]

CS-Parts (**c**lassification-**s**pecific)

1. Initial classification
2. Gradient of the used features w.r.t the input ⇒ “saliency”

$$M_{x,y}(\mathbf{I}) = \frac{1}{|\mathfrak{D}|} \sum_{d \in \mathfrak{D}} \left| \frac{\partial}{\partial I_{x,y}} f^{(d)}(\mathbf{I}) \right|$$



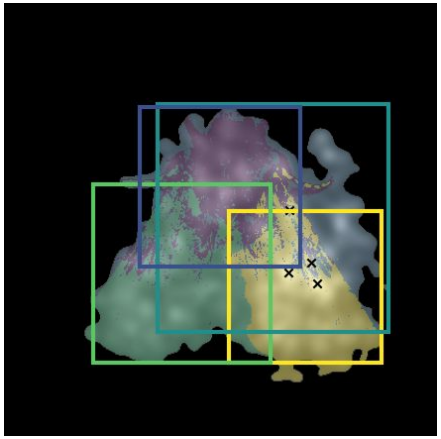
Korsch, Dimitri, Paul Bodesheim, Joachim Denzler. "Classification-Specific Parts for Improving Fine-Grained Visual Categorization" GCPR. 2019

Species Classification - Unsupervised Part Estimation [Korsch et al., 2019]

CS-Parts (**c**lassification-**s**pecific)

1. Initial classification
2. Gradient of the used features w.r.t the input ⇒ “saliency”
3. Post-process of the saliency
4. Clustering and BBox creation

$$M_{x,y}(\mathbf{I}) = \frac{1}{|\mathfrak{D}|} \sum_{d \in \mathfrak{D}} \left| \frac{\partial}{\partial I_{x,y}} f^{(d)}(\mathbf{I}) \right|$$



Korsch, Dimitri, Paul Bodesheim, Joachim Denzler. "Classification-Specific Parts for Improving Fine-Grained Visual Categorization" GCPR. 2019

Species Classification - Unsupervised Part Estimation [Korsch et al., 2019]

$$M_{x,y}(\mathbf{I}) = \frac{1}{|\mathfrak{D}|} \sum_{d \in \mathfrak{D}} \left| \frac{\partial}{\partial I_{x,y}} f^{(d)}(\mathbf{I}) \right|$$

CS-Parts (**c**lassification-**s**pecific)

1. Initial classification
2. Gradient of the used features w.r.t the input ⇒ “saliency”
3. Post-process of the saliency
4. Clustering and BBox creation
5. Crop parts and classify

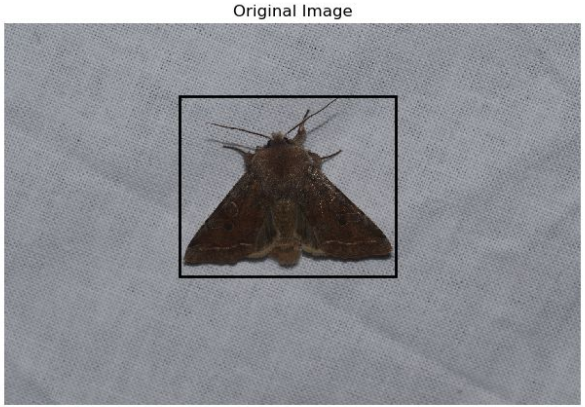
Korsch, Dimitri, Paul Bodesheim, Joachim Denzler. "Classification-Specific Parts for Improving Fine-Grained Visual Categorization" GCPR. 2019

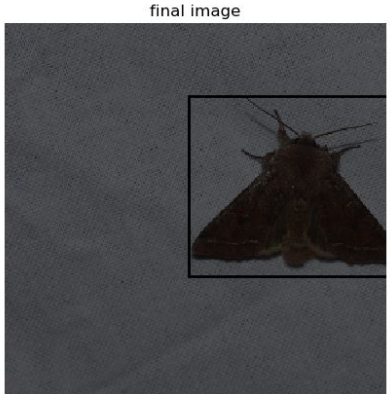
Experiments

Moth Detector

Training

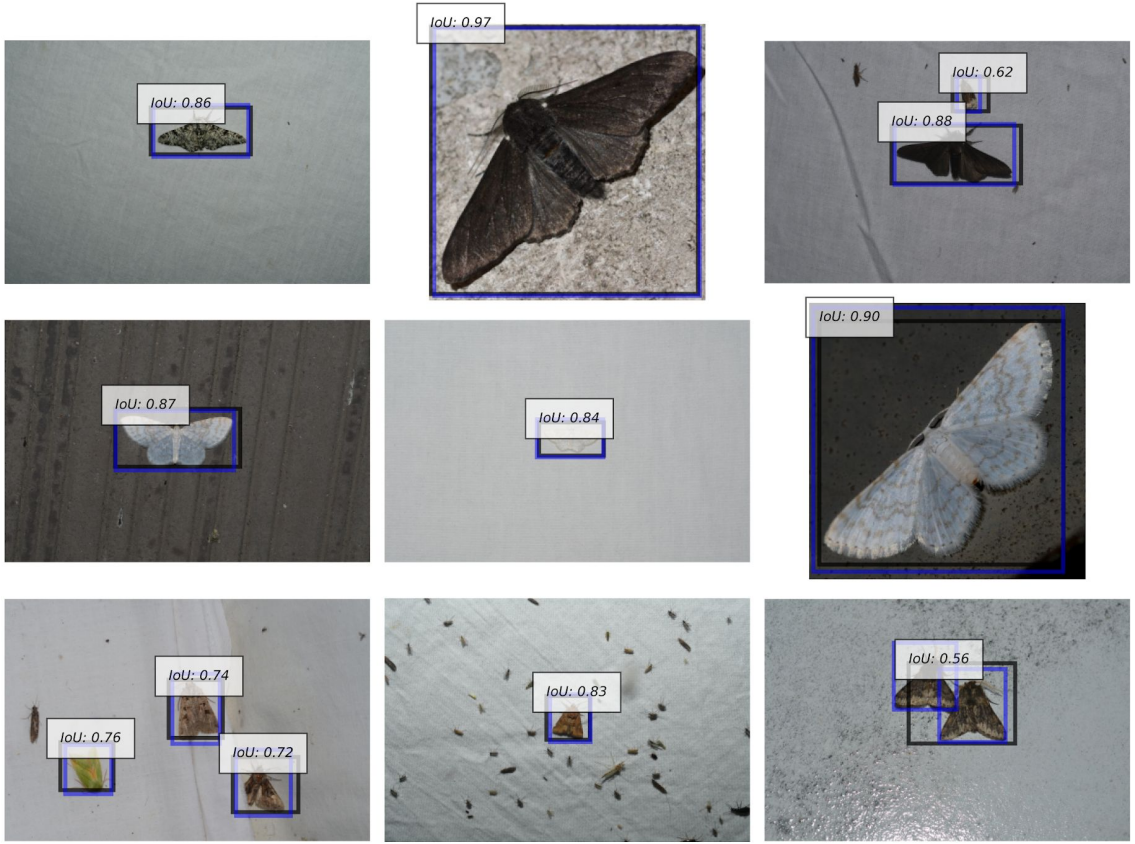
- RMSProp, 60 epochs, 5e-4 L2-Reg
- LR: 1e-4 LR; reduce after 20 epochs
- Augmentations:
  - color jittering
    (brightness, contrast, saturation, hue)
  - random cropping
  - random horizontal flipping





Moth Detector

IoU threshold	mAP in %
0.75	88.9 (± 0.8)
0.50	99.0 (± 0.1)



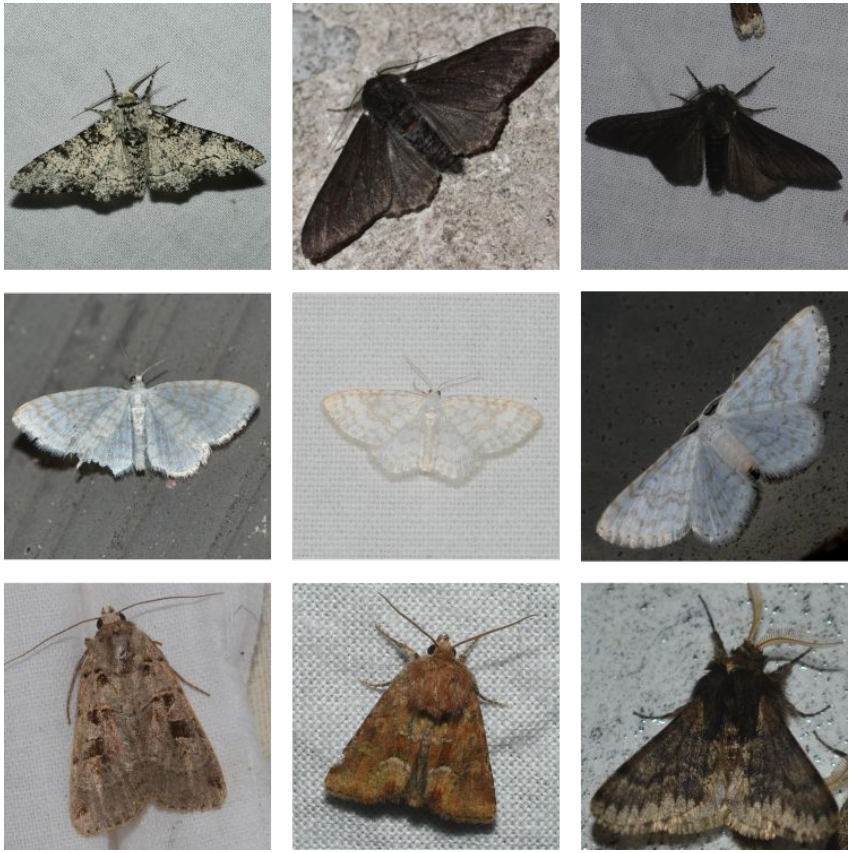
Ground-truth boxes

Predicted boxes

Moth Classifier

Training

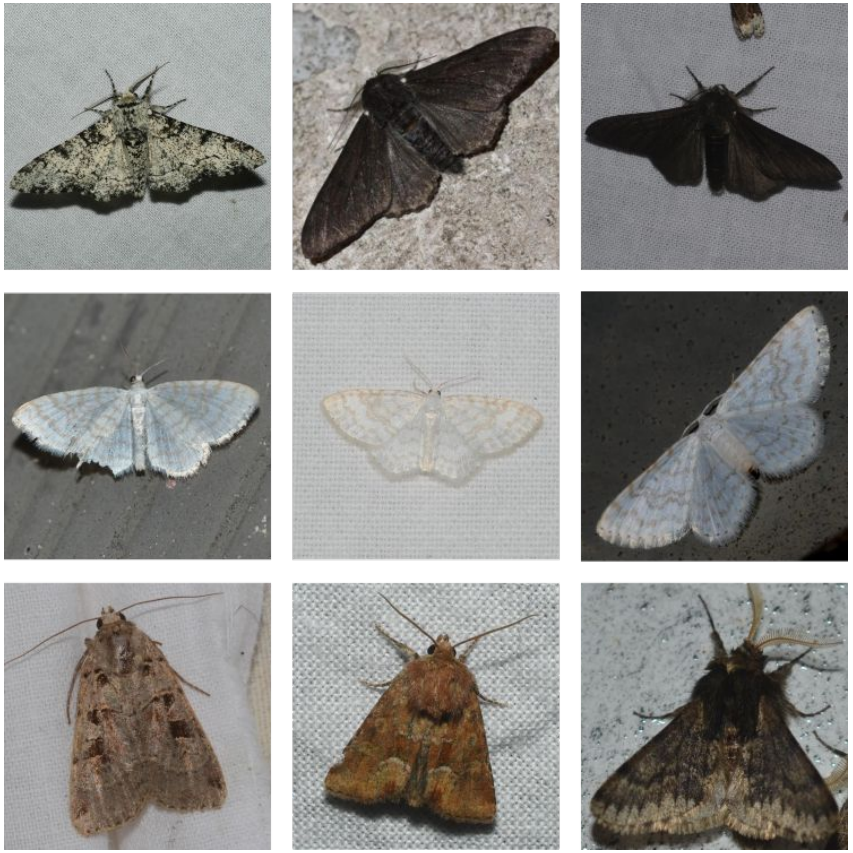
- RMSProp, 60 epochs, 5e-4 L2-Reg
- LR: 1e-4 LR; reduce after 20 epochs
- Augmentations:
  - color jittering
    (brightness, contrast, saturation, hue)
  - random cropping
  - random horizontal flipping



Moth Classifier

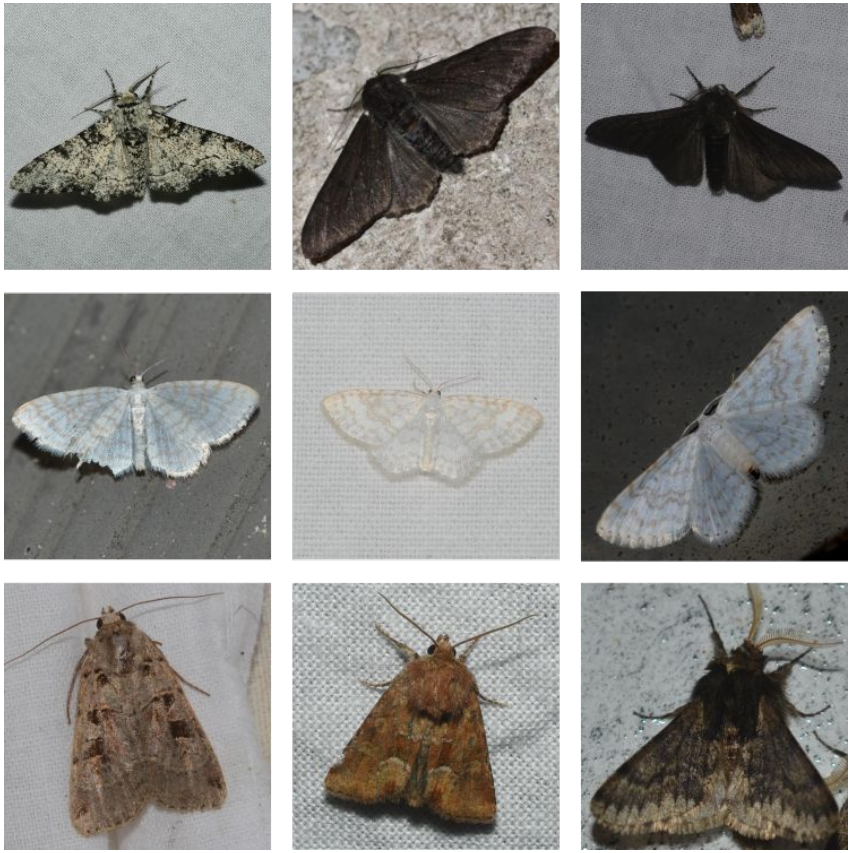
Setups

1. last FC layer vs. entire CNN
2. pre-training on
   ImageNet vs. iNaturalist 2017
3. no parts vs. CS-parts



Moth Classifier

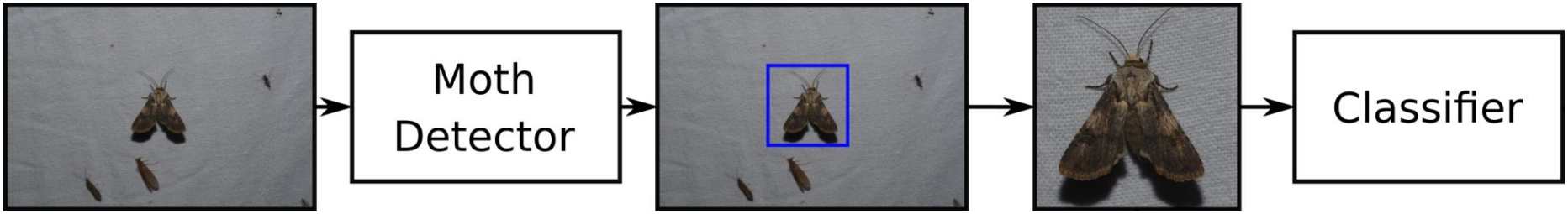
		Accuracy in %	
	Fine-tuning	ImageNet	iNaturalist
No Parts	*only FC Layer*	63.3 (± 0.4)	86.6 (± 0.4)
	entire CNN	89.5 (± 0.9)	90.5 (± 0.1)
With CS-Parts	*only FC Layer*	71.8 (± 0.4)	88.0 (± 0.4)
	entire CNN	**91.5 (± 0.6)**	**93.1 (± 0.8)**



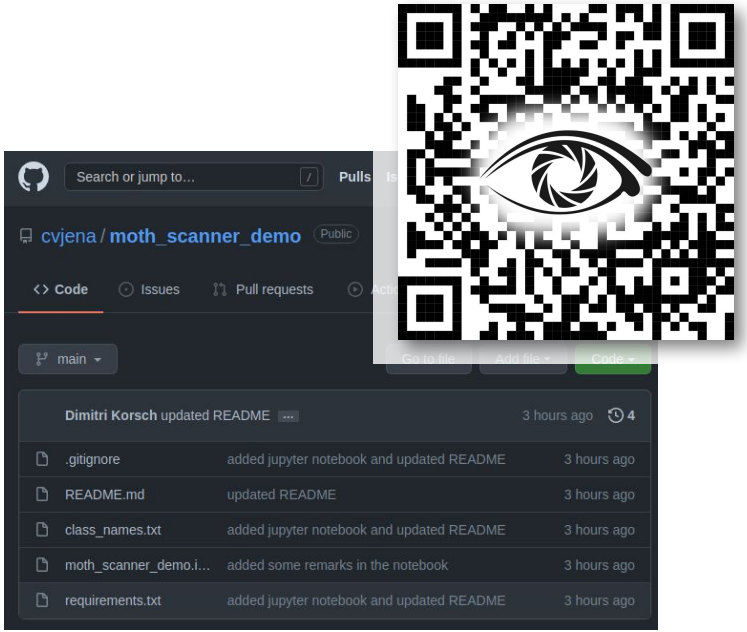
Entire Pipeline

- combination of 10 detectors and 10 classifiers
- baseline: 10 classifiers trained on uncropped images

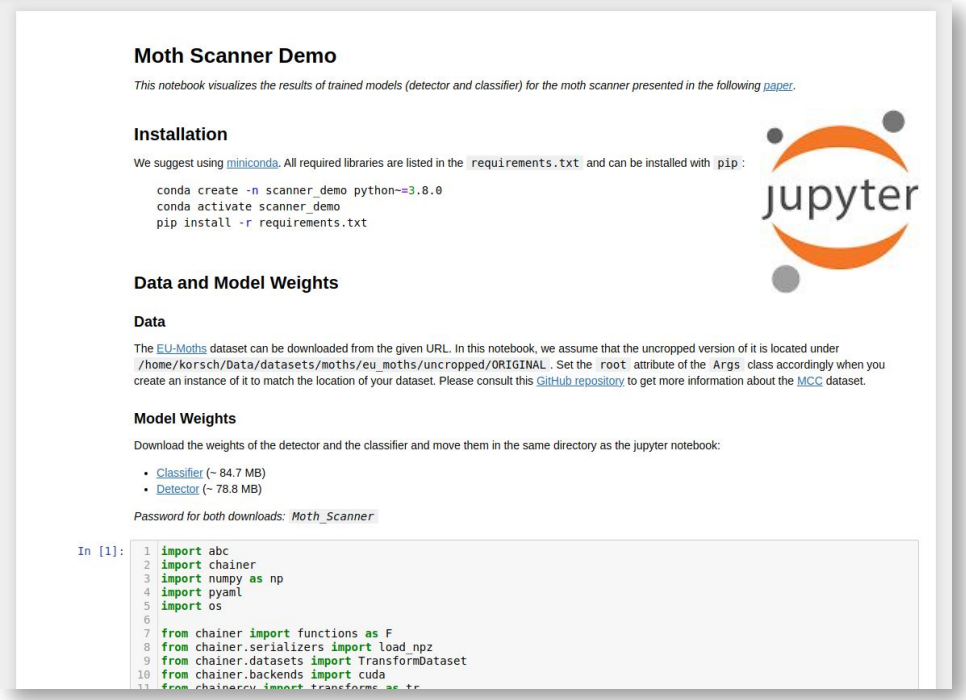
	Accuracy in %
Classifier only	79.6 (± 1.1)
Detector + Classifier	88.1 (± 0.6)



Demo Code



https://github.com/cvjena/moth_scanner_demo



Moth Scanner Demo

This notebook visualizes the results of trained models (detector and classifier) for the moth scanner presented in the following paper.

Installation

We suggest using miniconda. All required libraries are listed in the `requirements.txt` and can be installed with `pip`:

```
conda create -n scanner_demo python~=3.8.0
conda activate scanner_demo
pip install -r requirements.txt
```

Data and Model Weights

Data

The EU-Moths dataset can be downloaded from the given URL. In this notebook, we assume that the uncropped version of it is located under `/home/korsch/Data/datasets/moths/eu_moths/uncropped/ORIGINAL`. Set the `root` attribute of the `Args` class accordingly when you create an instance of it to match the location of your dataset. Please consult this GitHub repository to get more information about the MCC dataset.

Model Weights

Download the weights of the detector and the classifier and move them in the same directory as the jupyter notebook:

- Classifier (~ 84.7 MB)
- Detector (~ 78.8 MB)

Password for both downloads: `Moth_Scanner`

In [1]:

```
import abc
import chainer
import numpy as np
import pyaml
import os

from chainer import functions as F
from chainer.serializers import load_npz
from chainer.datasets import TransformDataset
from chainer.backends import cuda
```

SPONSORED BY THE

Federal Ministry
of Education
and Research

FONA

Research for Sustainability

Thank you! Questions?