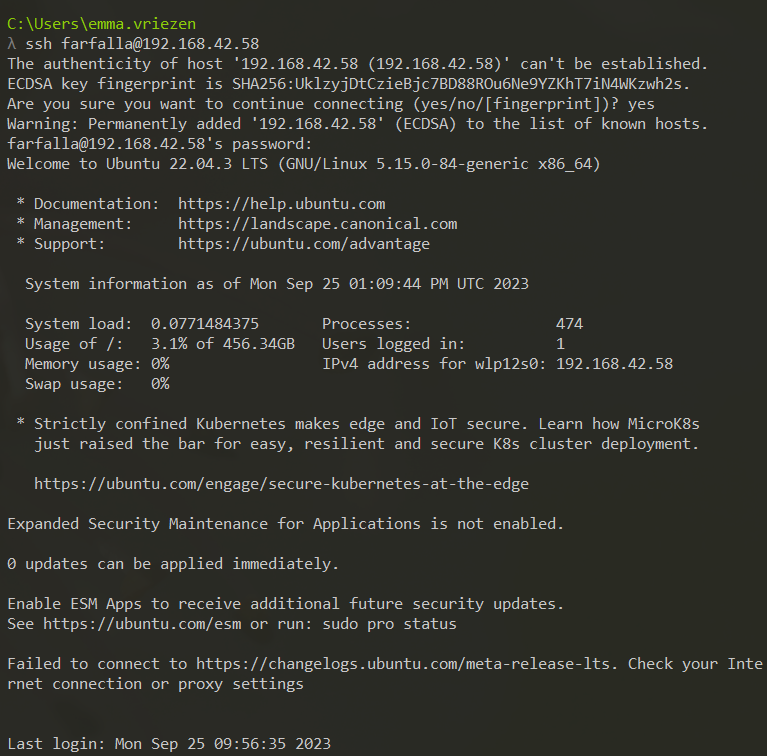
Aan de slag met Beeldherkenning

Uitleg bij de nachtvlinderbeeldherkenning van De Vlinderstichting (DVS).

Repository: [GitHub – moth-classification](https://github.com/EmmaDVS/moth-classification)

# Verbinden met Farfalla

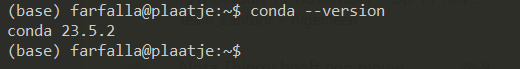
1. Bevind je in het DVS netwerk op kantoor of verbind via VPN
2. Eerste keer verbinden:

ssh -L 8080:localhost:8080 -L 6006:localhost:6006 farfalla@192.168.42.63

# Installatie (voor op een andere computer)

Op Farfalla is alles al geïnstalleerd. Hier staan wel de stappen van het installatieproces, om het te repliceren:

1. (Alleen voor nieuwe installatie) Installeer miniconda

* Maak een folder: mkdir miniconda 3
* Download: wget [https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86\_64.sh -O ~/miniconda3/miniconda.sh](https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh%20-O%20~/miniconda3/miniconda.sh)
* Installeer: bash ~/miniconda3/miniconda.sh -b -u -p ~/miniconda3
* Verwijder installer: rm -rf ~/miniconda3/miniconda.sh
* Initialise: ~/miniconda3/bin/conda init bash
* Herstart de computer. Bij het opstarten zal de zgn. base environment geactiveerd zijn en werkt de conda command:

1. (Alleen voor nieuwe installatie) Maak een environment aan

* Lege environment aanmaken: conda create -n ml python
  + -n = name
  + python = installeer de nieuwste python
* Activeer: conda activate ml

1. (Alleen voor nieuwe installatie) Gebruik environment.yml van mothRecognition om een environment ‘ml’ (machine learning) aan te maken. Gebruik envrionment.yml van mothDetection of een envrionment ‘yolo’ (het model voor object detectie) aan te maken.

*Geen environment.yml gevonden?*

3.1. (Alleen voor nieuwe installatie) Installeer jupyter

* Conda install jupyter
  + Dit is inclusief ipykernel
* Voeg de environment toe aan jupyter kernels: ipython kernel install –user –name=ml
* Start een notebook server: jupyter notebook
  + Deze opent automatisch in je browser

3.2 (Alleen voor nieuwe installatie) Installeer de benodigde packages:

* conda install pytorch torchvision torchaudio pytorch-cuda=11.8 -c pytorch -c nvidia
  + torch.cuda.is\_available() == False? Verkeerde CUDA versie:
  + conda install pytorch torchvision torchaudio pytorch-cuda=12.1 -c pytorch -c nvidia
* conda install -c conda-forge torchinfo
* conda install matplotlib
* conda install -c conda-forge tensorboard
* conda install -c conda-forge scikit-learn

1. (Alleen voor nieuwe installatie) REMOTE jupyter:

* Dit is alleen relevant als je via SSH op een andere computer werkt
* Sudo apt install screen
* Screen + enter
* Navigeer naar folder waarin je jupyter wilt runnen
* Conda activate ml
* Jupyter notebook –no-browser –port=8080
* Ctrl + A + D
  + Detacht het huidige scherm en gaat terug naar origineel
* Exit
* Maak een nieuwe ssh verbinding met port-forwarding: ssh -L 8080:localhost:8080 [farfalla@168.192](mailto:farfalla@168.192)...
* Ga in je lokale browser naar localhost:8080
  + Dit verbindt met de jupyter server
* In de ssh en de correcte env: jupyter notebook list
  + Kopiëer de token
* Voer de token in onderaan de pagina van Jupyter notebook, en kies een nieuw wachtwoord. Schrijf deze op, want die is later misschien weer nodig.

# Environments

Op Farfalla staan twee environments: ml en yolo.

‘ml’ is voor alles in de folder mothRecognition. ‘yolo’ is voor de folder mothDetection.

In Jupyter moet de juiste environment geselecteerd staan. Dit is al het geval. Als er iets misgaat met het importeren van modules, kijk dan welke environment actief staat (rechtsboven in de Notebook). Los het binnen die environment op (of wissel naar de correcte environment, als de verkeerde geselecteerd is).

# Uitleg van principes beeldherkenning

* Om AI te begrijpen komt er eigenlijk wat wiskunde om de hoek kijken. Het is echter niet nodig om de onderliggende theorie onder de knie te hebben, om er wel mee aan de slag te gaan (voor simpele applicaties). Toch is het essentieel om je in de basis in te lezen. Een goede plek om te beginnen is deze lijst met uitleg over veel basis termen die bij AI komen kijken is hier te vinden: [What is a Neural Network?](https://aws.amazon.com/what-is/neural-network/)
* Het automatisch classificeren van foto’s van nachtvlinders is een vorm van ‘image recognition’ (= beeldherkenning). Het model waarop de nachtvlinderbeeldherkenning gebouwd is, heet EfficientNet. Dit is een convolutional network. Het neemt afbeeldingen als input. Elke afbeelding heeft een label. Het model probeert patronen in de afbeeldingen te herkennen, die het mogelijk maken om ze in de juiste categorie te plaatsen. Een duidelijke uitleg van convolutional networks is te vinden in deze YouTube video: [Convolutional Neural Networks (CNNs) explained](https://www.youtube.com/watch?v=YRhxdVk_sIs).
* Het idee is om zo veel mogelijk foto’s aan het model te laten zien. Hoe meer, hoe beter. Het is belangrijk dat een dataset divers is. De foto’s hoeven bijvoorbeeld niet allemaal van perfecte kwaliteit te zijn. Gevarieerde data zorgt ervoor dat het model generieker wordt, d.w.z. dat het redelijk tot goed presteert onder veel omstandigheden en minder afhankelijk is van de kwaliteit van de input.

# Veranderingen 2023 (Emma) t.o.v. 2022 (Kay)

* Alles is verplaatst naar Jupyter Notebooks in plaats van verschillende Python projecten. Hierdoor is het project kleiner geworden en staat er documentatie bij de code.
* Een kleiner model wordt gebruikt. EfficientNet komt in 8 maten: B0 t/m B7. In 2022 werd B7 gebruikt. Dit paste echter net op de GPU van Belle (een computer van DVS), waardoor er geen grote batch sizes konden worden gebruikt (grotere batches hebben meer videogeheugen nodig). Hierdoor duurde het trainen heel lang. De kleinere EfficientnetB1 bleek beter te presteren dan B7, trainde sneller (kleiner model = minder parameters), en nam ook nog eens minder ruimte in op de GPU. Bovendien zijn kleinere modellen ook sneller tijdens inferentie.
* Alle soorten worden meegenomen. In de oude versie werden alle soorten met minder dan 30 foto’s op één stapel gegooid. Dit veroorzaakte dat er één grote, heterogene categorie aan foto’s ontstond. Een neuraal netwerk kan niet optimaliseren voor zo’n ‘rest’klasse, want er zijn simpel gezegd geen gedeelde kenmerken tussen alle foto’s binnen die klasse. In de huidige versie zijn alle soorten tot een eigen categorie gemaakt, onafhankelijk van het aantal foto’s dat er beschikbaar is voor deze soort.
* Een ‘object detection’ model is toegevoegd, om de locaties van motten in een foto te bepalen. Dit model is gebaseerd op YOLOv5. Dit soort model kan getrained worden om ‘bounding boxes’ om doelobjecten te tekenen. Dus in het geval van nachtvlinders, leert het model nachtvlinders te omlijnen met een rechthoek op een foto die je als input geeft. De output van het model (i.e. de rechthoek binnen een foto) kan dan uitgesneden worden. Meer informatie over YOLO is hier te lezen: [YOLO Object Detection Explained](https://www.datacamp.com/blog/yolo-object-detection-explained).