05/12/2024

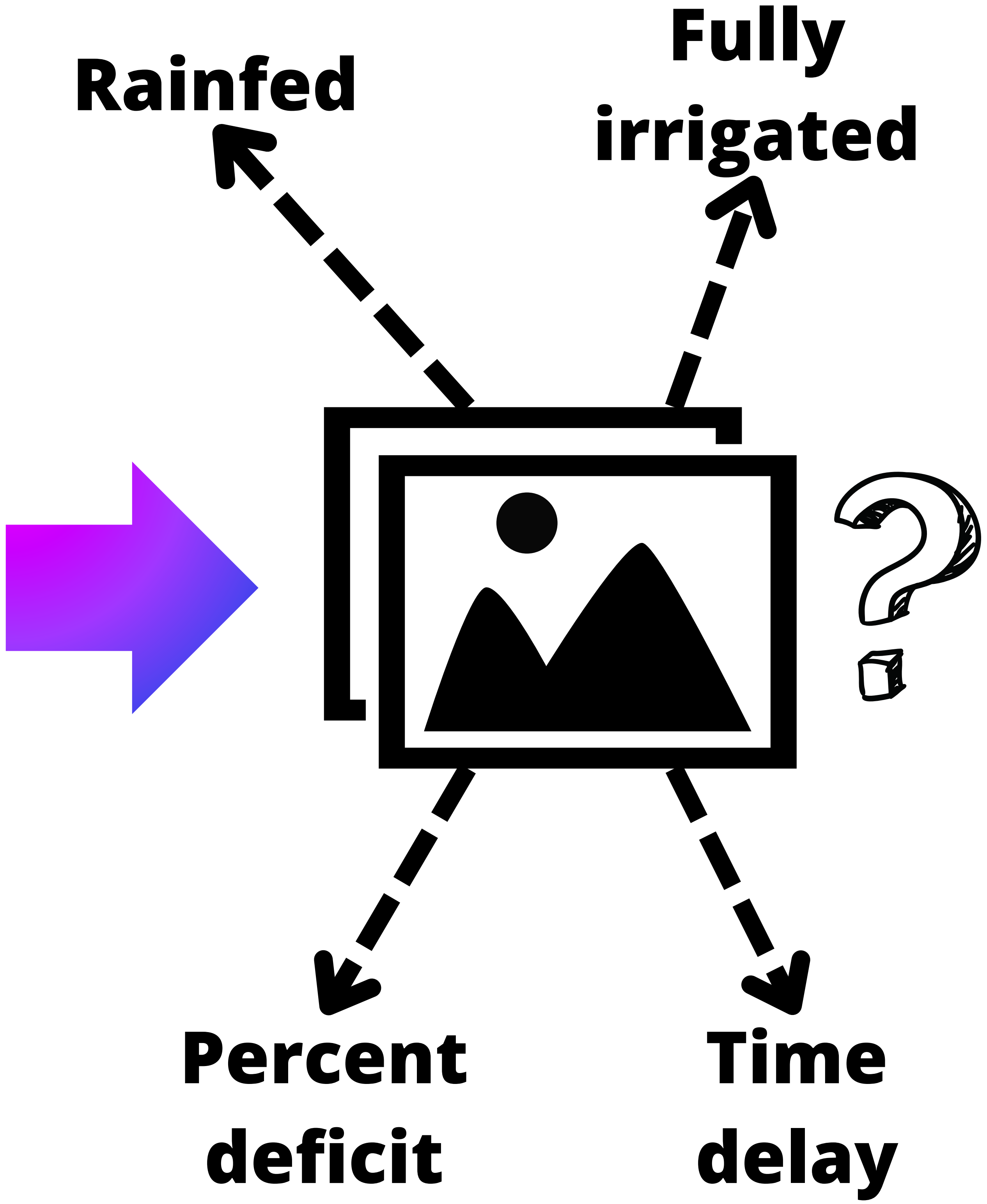
Presentation TP - DL

Title: Classification of cotton water stress using inception models and UAV-based RGB imagery

Presented by : Lydia Mouhoun

Context


Rainfed
Fully irrusied
Fully irrigied
Perser deficit
Time Deficict
Time Dela


Rainfed
Fully irrigated
Percent deficit
Time delay

[1] Classification of cotton water stress using convolutional neural networks and UAV-based RGB imagery

Date: 01/02/2024
Type: Research paper

Keywords: cotton; irrigation; water stress; UAV; RGB; evapotranspiration; convolutional neural networks; random forest

Problematic: How can convolutional neural networks (CNN) and UAV-based RGB imagery be effectively utilized to classify water stress in cotton crops?

Purpose: analyze and classify the effects of different irrigation strategies on cotton water stress, thereby providing insights into optimal irrigation practices for enhancing plant growth.

Key findings:

- **CNN** model achieved high accuracy (91%) in classifying different irrigation treatments ("rainfed", "full irrigation", "percent deficit ", "time delay")
- Analysis of canopy-only and soil-only images provided insights but did not outperform the original RGB dataset.
- Feature importance analysis, using **random forest** model, identified key features such as red and blue channels and canopy cover (i.e physiological changes) as significant for water stress assessment.

Limits & Future Work:

- The study suggests integrating environmental data (temperature, humidity, solar radiation) with RGB imagery to enhance classification accuracy.
- Future research should focus on training CNN models on larger, diverse datasets and applying transfer learning techniques to improve generalization across different regions and cultivars.

Designing CNNs is not that simple and effecient

Source: [2]

Problematic

Adding layers for better feature extraction is more costly, prone to overfitting and not necessarily effective !

How to make CNNs more efficient without going deeper ?

Problematic

Adding layers for better feature extraction is more costly, prone to overfitting and not necessarily effective !

How to make CNNs more efficient without going deeper ?

Inception Models

PLAN

1. Evolution of Inception Models
2. Proposed Inception Model
3. Discover Noise & Denoiser models
4. Final Proposed Model
5. Limits & Future Work

1

Evolution of Inception Models

Source: [2][3]

INCEPTION V1 2014

- Filters with multiple sizes operate parallelly
- Adding an extra 1x1 convolution
- Add auxilary classifers
- ex: GoogLeNet

INCEPTION V2 2015

- Reducing the dimensions too much may cause loss of information, known as a “representational bottleneck”
- Using smart factorization methods
- Or, asymmetric convolutions

INCEPTION V3 2015

- Auxiliary classifiers didn’t contribute much until near the end of the training process
- Adding RMSProp Optimizer, Factorized 7x7 convolutions, BatchNorm, Label Smoothing

INCEPTION V4 2016

- Some of the modules were more complicated than necessary.
- Make the modules more uniform -> boost performance
- Introduced “Reduction Blocks”

INCEPTION RESNET 2016

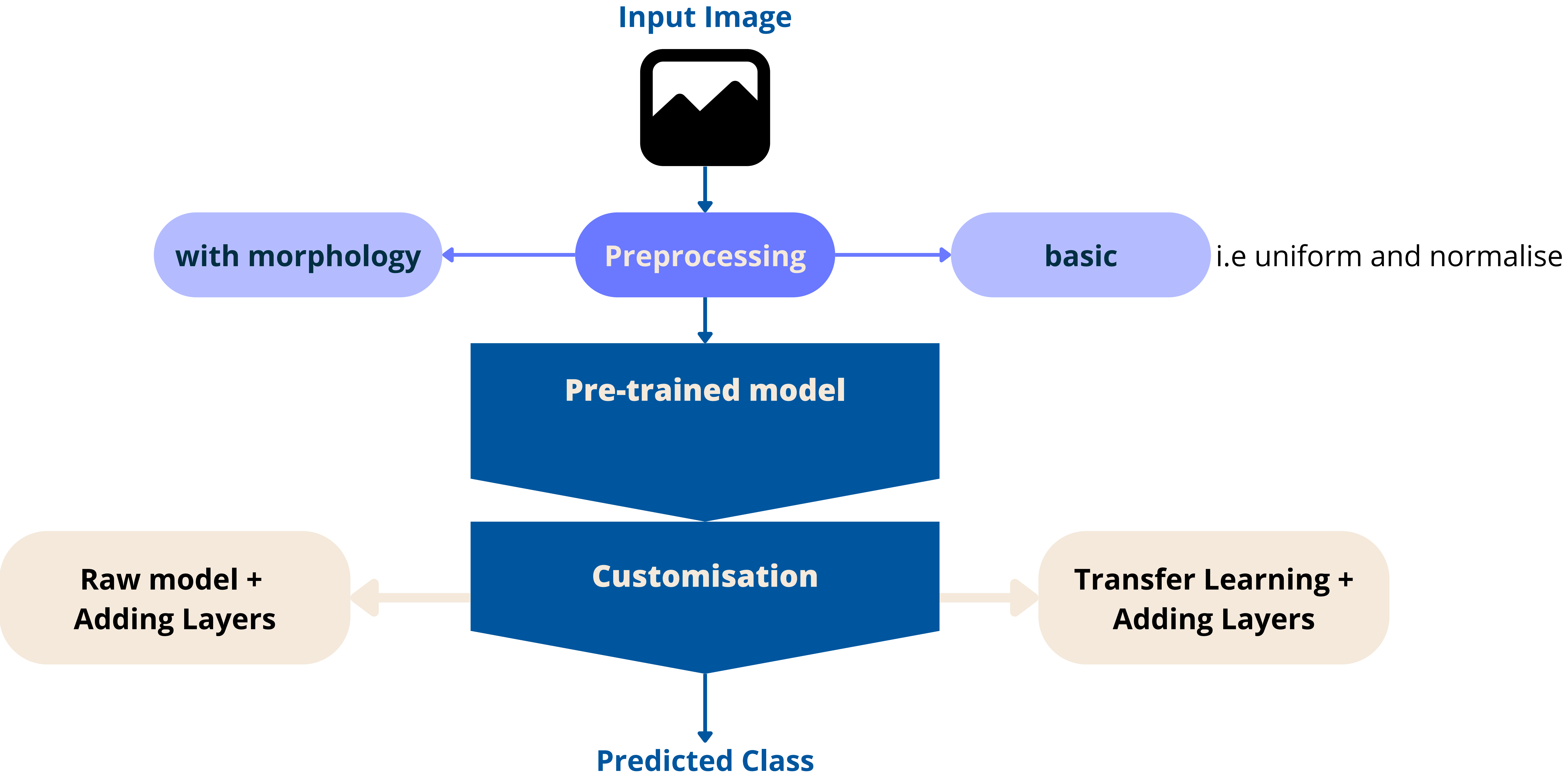
V1

V2

- Introduce residual connections (output=input)
- Higher accuracy with less epochs

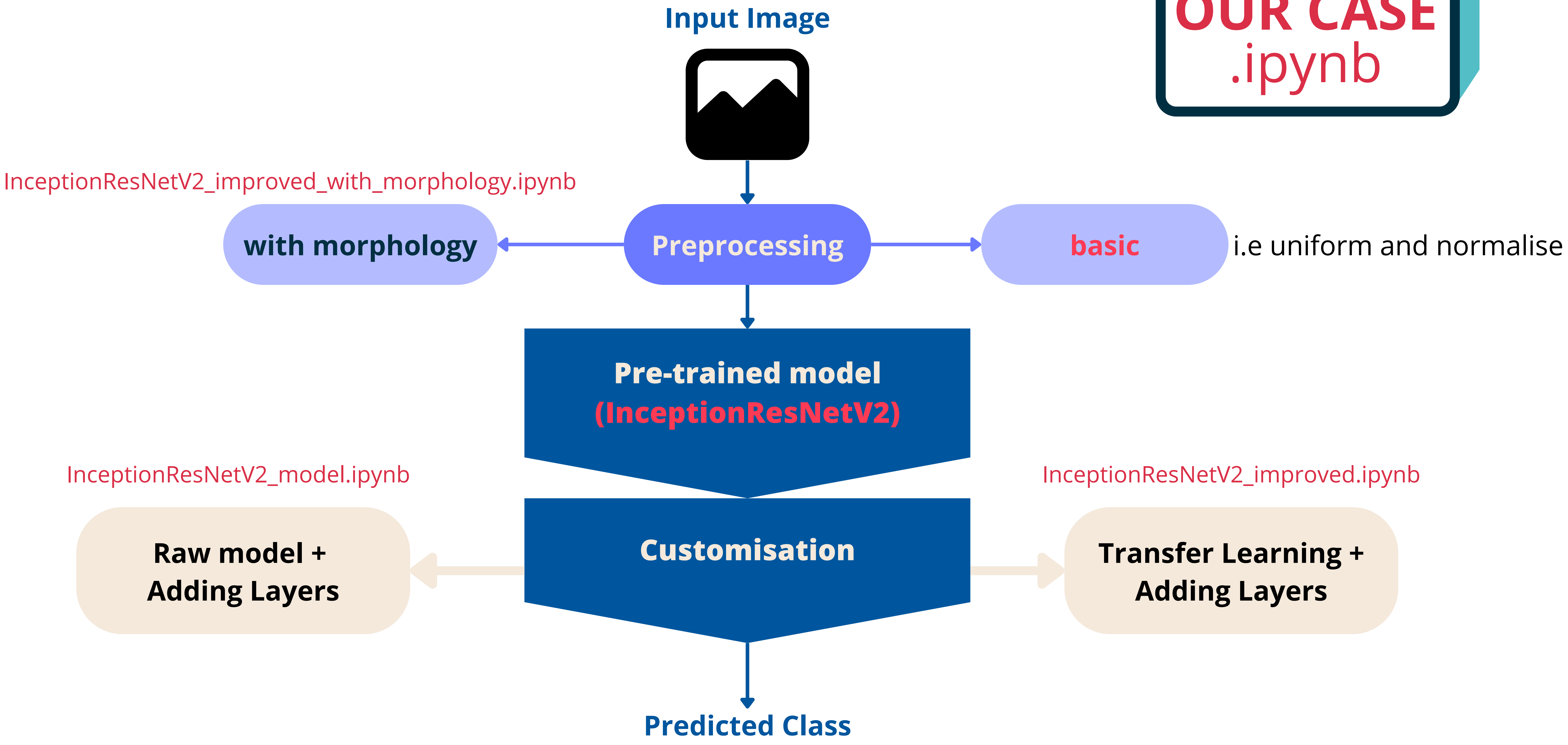
2

Proposed Inception Model



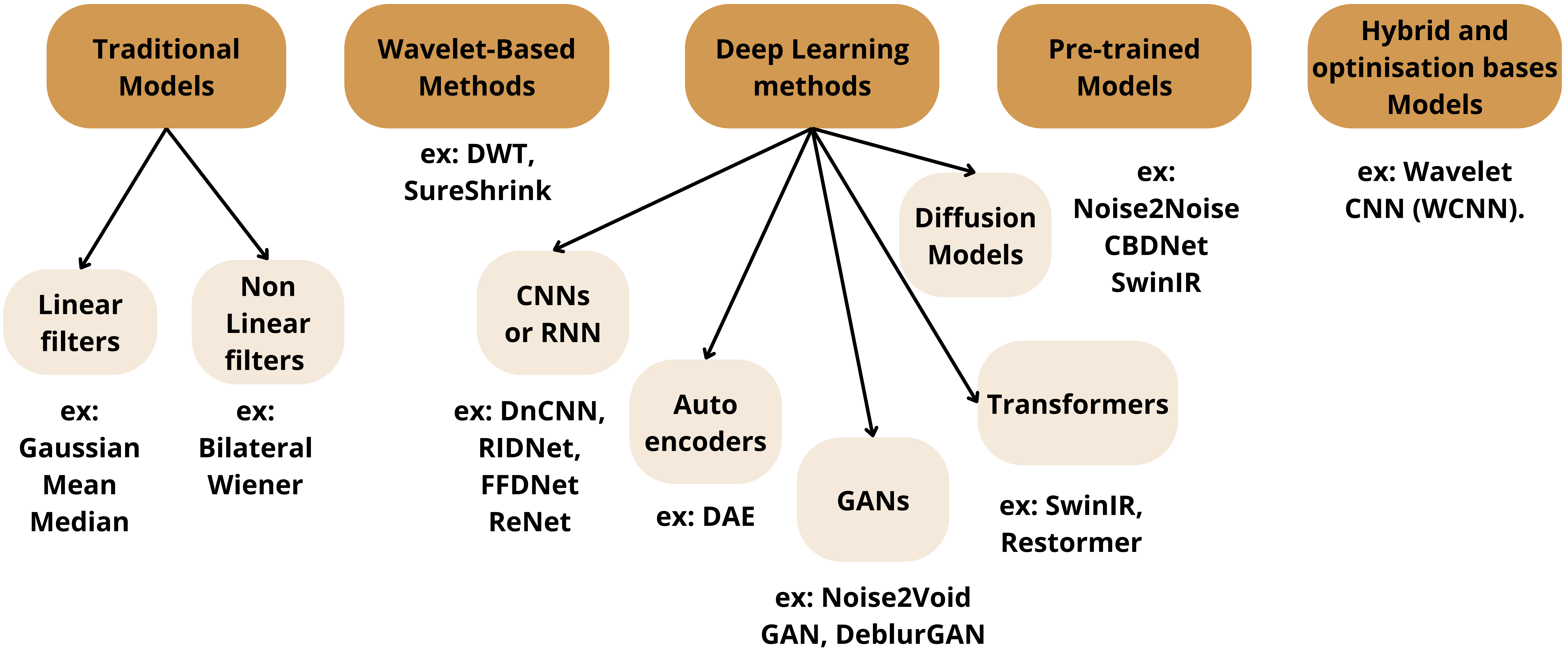
2

Proposed Inception Model



3

Discover Noise & Denoiser models

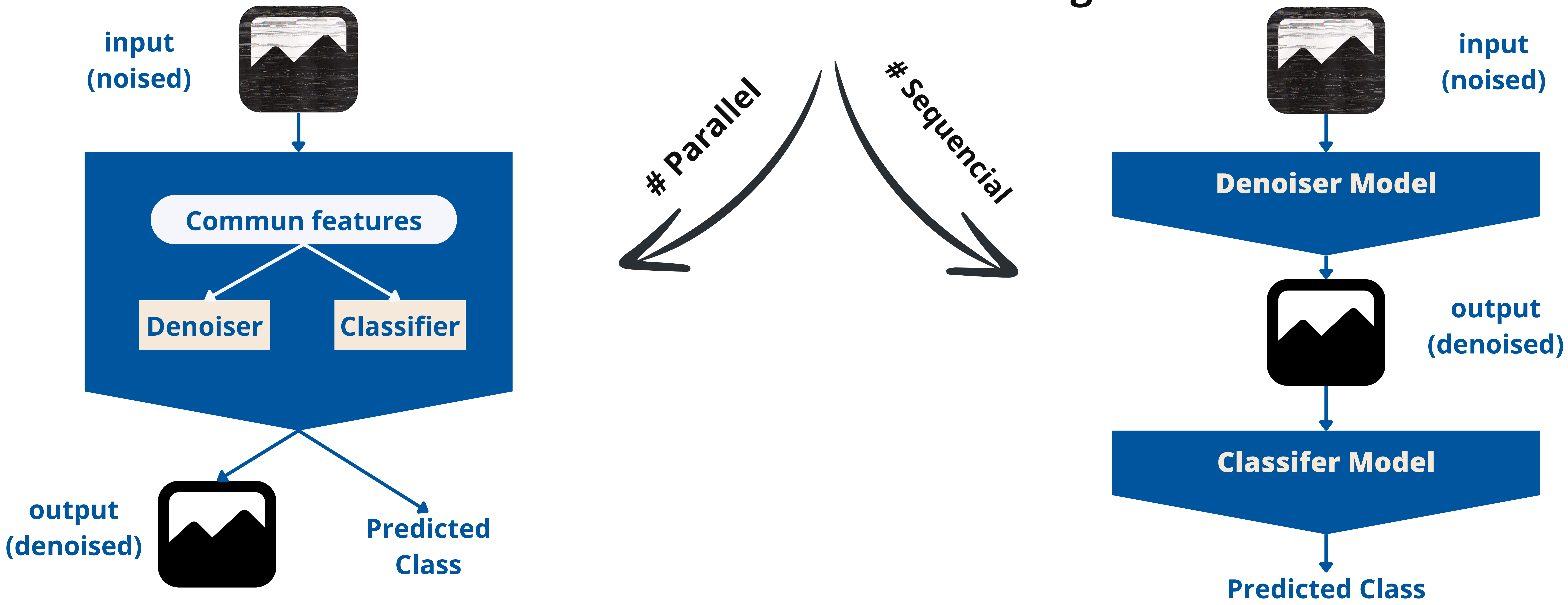


4

Final Proposed Model

How to integrate two different tasks:

classification + **denoising**

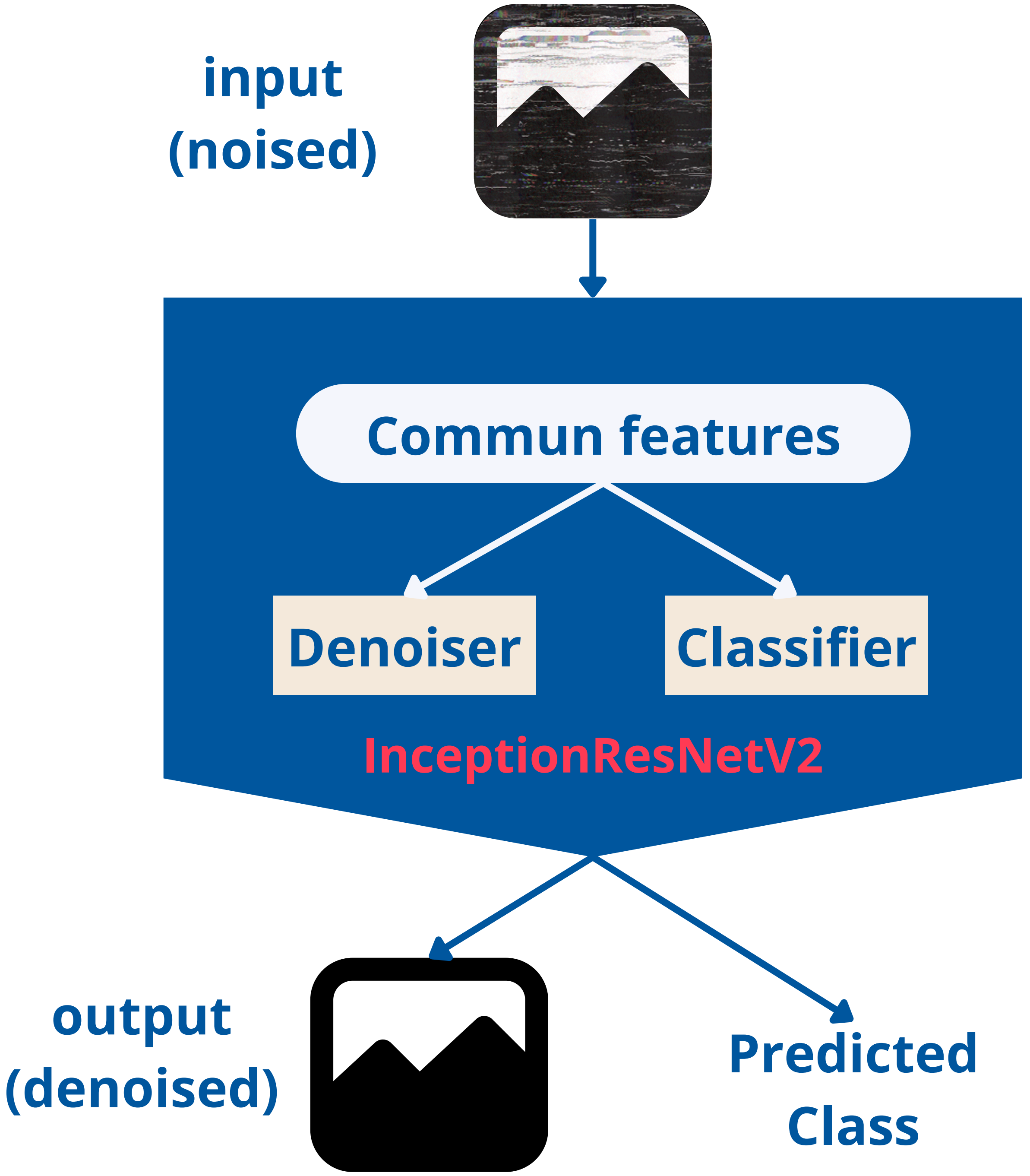


4

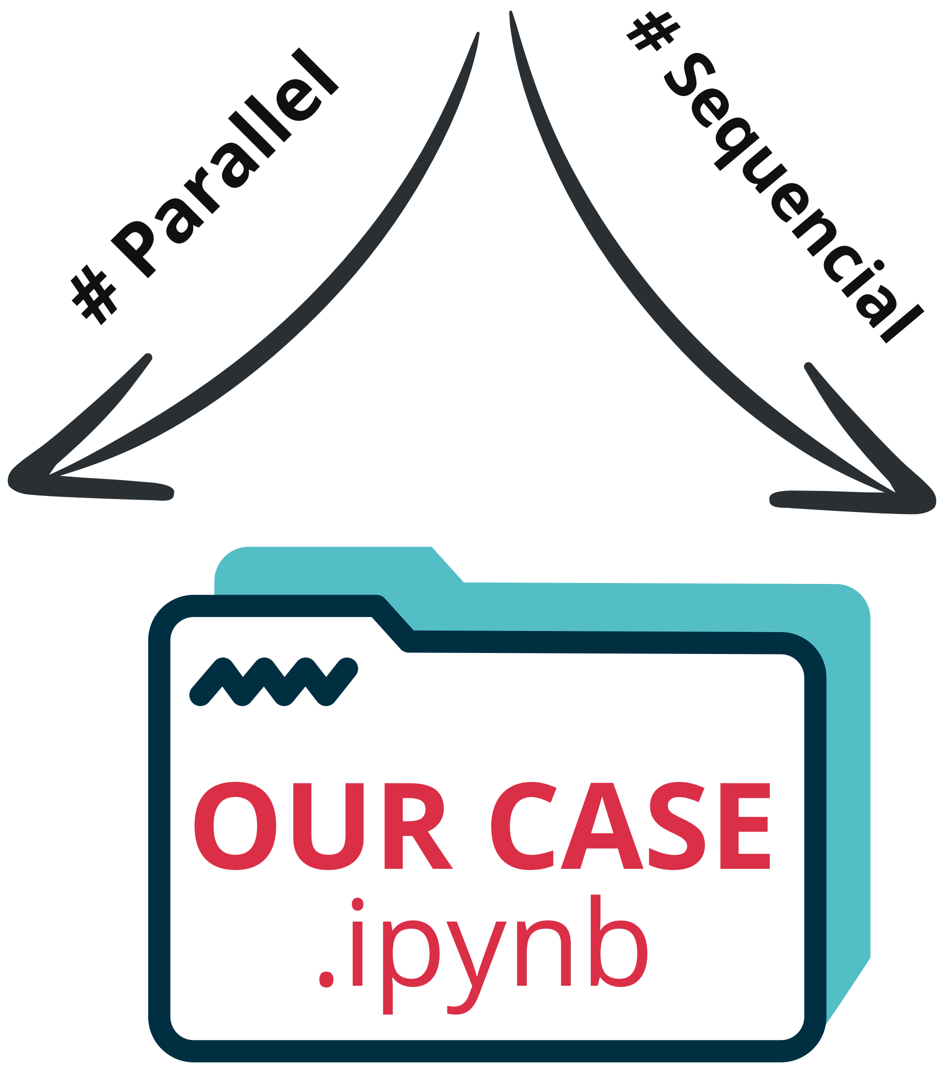
Final Proposed Model

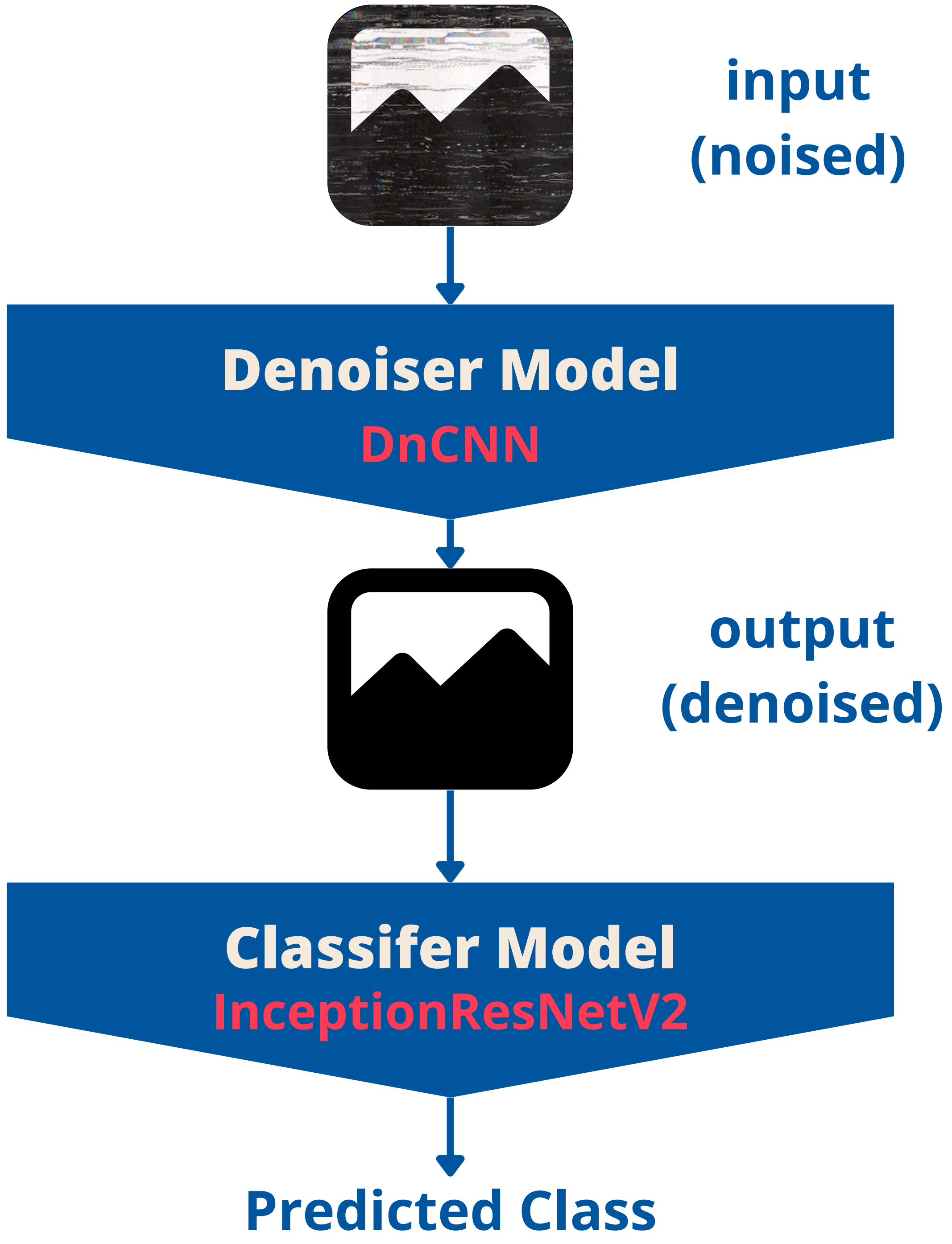
How to integrate two different tasks:

classification + denoising



InceptionResNetV2_with_denoising_MTL{_improved}.ipynb





InceptionResNetV2_with_denoising_DnCNN.ipynb

RECAP

Tested only on one dataset: 18 August 2022 / nb_epochs = 30

Filename	Description	Training time	Training accuracy%	Validation accuracy%
InceptionResNetV2_model.ipynb	Based on InceptionResNetV2 pre-trained model with adding final layers	52 min	58	60
InceptionResNetV2_improved.ipynb	Based on InceptionResNetV2 pre-trained model with transfer learning (TL) + final layers	60 min	98	70
InceptionResNetV2_improved_with_morphology.ipynb	Similar to the previous but by preprocessing the input images as described in [1]	64 min	97	72
InceptionResNetV2_with_denoising_DnCNN.ipynb	Sequencial process: DnCNN denoiser model feeds the TL Inception model	100 (5e) + 63 (20e)	mse: 0.004 acc: 98%	mse: 0.004 acc: 72%
InceptionResNetV2_with_denoising_MTL.ipynb	Parallel process: the inception model performs both denoising and classification	65 min	mse: 0.012 acc: 99%	mse: 0.012 acc: 73%

Machine: **RAM** 32GO, **CPU** Intel(R) Core(TM) Ultra 7 155H, **GPU** Intel(R) Arc(TM) Graphics (16GO)

Limits & Perspectives

- Transfer learning approach is prone to **overfitting**
- Better investigate the **dataset imbalances and noises** for additional preprocessing
- Add parameter/hyperparameter **turning** for optimal results
- Explore the potential of **transformers and attention**-based mechanisms
- Gather and include all spatial, spectral, and temporal **context** for better model accuracy

References

[1] Niu H, Landivar J, Duffield N. Classification of cotton water stress using convolutional neural networks and UAV-based RGB imagery. Advances in Modern Agriculture. 2024; 5(1): 2457. https://doi.org/10.54517/ama.v5i1.24 57
[2] Bharath, R. (2019, May 29). A Simple Guide to the Versions of the Inception Network. Towards Data Science. Retrieved from https://towardsdatascience.com/a-simple-guide-to-the-versions-of-the-inception-network-7fc52b863202
[3] Alifia, G. (2022, May 21). Understanding Architecture Of Inception Network & Applying It To A Real-World Dataset. Retrieved from https://gghantiwala.medium.com/understanding-the-architecture-of-the-inception-network-and-applying-it-to-a-real-world-dataset-169874795540

Thank you

Contact me

Lydia Mouhoun
3CS SIQ2 student

kl_mouhoun@esi.dz