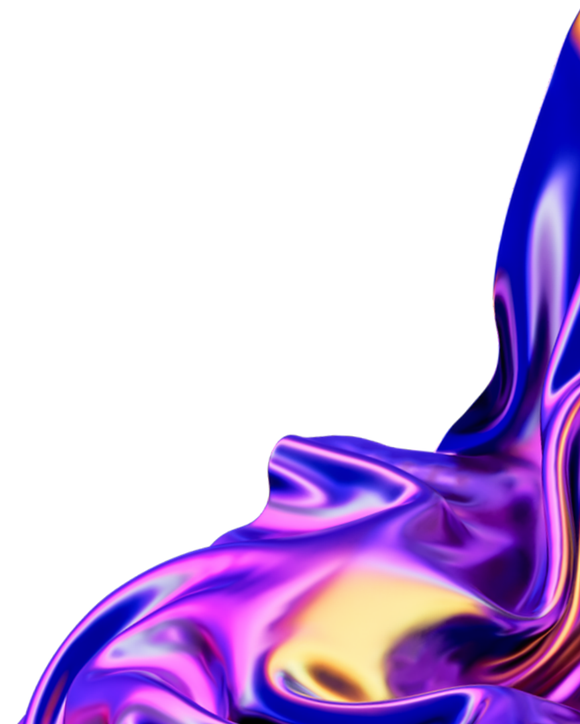
Pixel-aware Semantic Segmentation for self-driving cars

Featuring:
David Moe
Tommaso Grandi

Table of contents

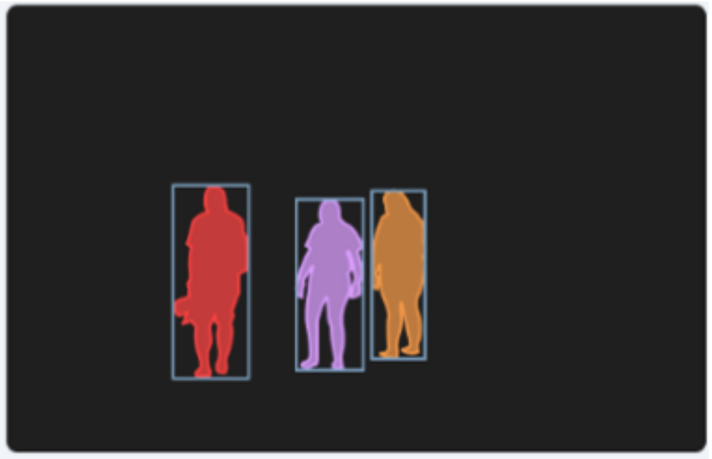
- Task presentation
- Dataset
- Preprocessing
- Unet
- Architecture and blocks
- Models training
- Results and comparison



Task presentation

Image

Semantic Segmentation



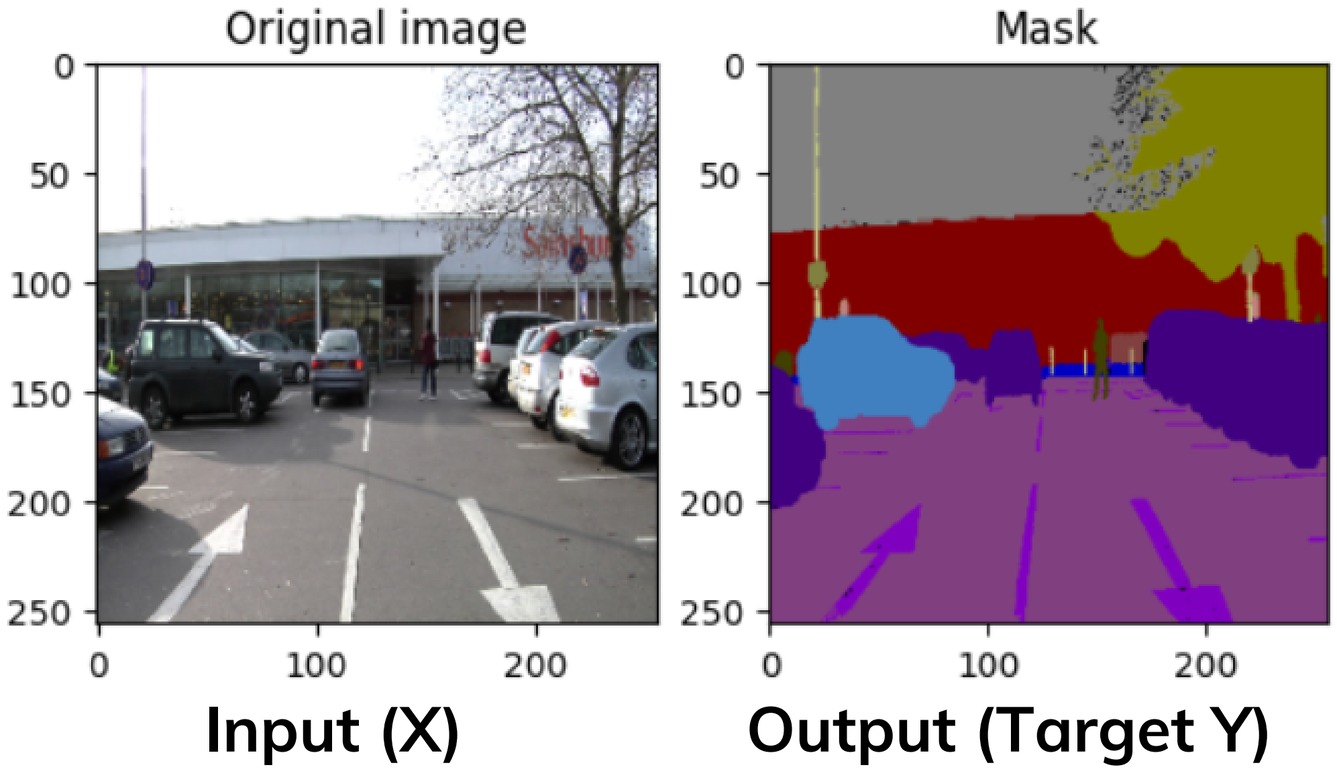
Instance Segmentation

Panoptic Segmentation

Dataset

- Road Image Semantic Segmentation for **Self-driving cars**
- Leveraged **CNN encoder-decoder** architectures for training and inference

- CamVid dataset with 30 classes (**Multiclass**)
- OG images paired with ground truth mask
- Number of image-mask pairs:
  - Train: 369
  - Validation: 232
  - Test: 100
- **Small sample size** and high class **imbalance**



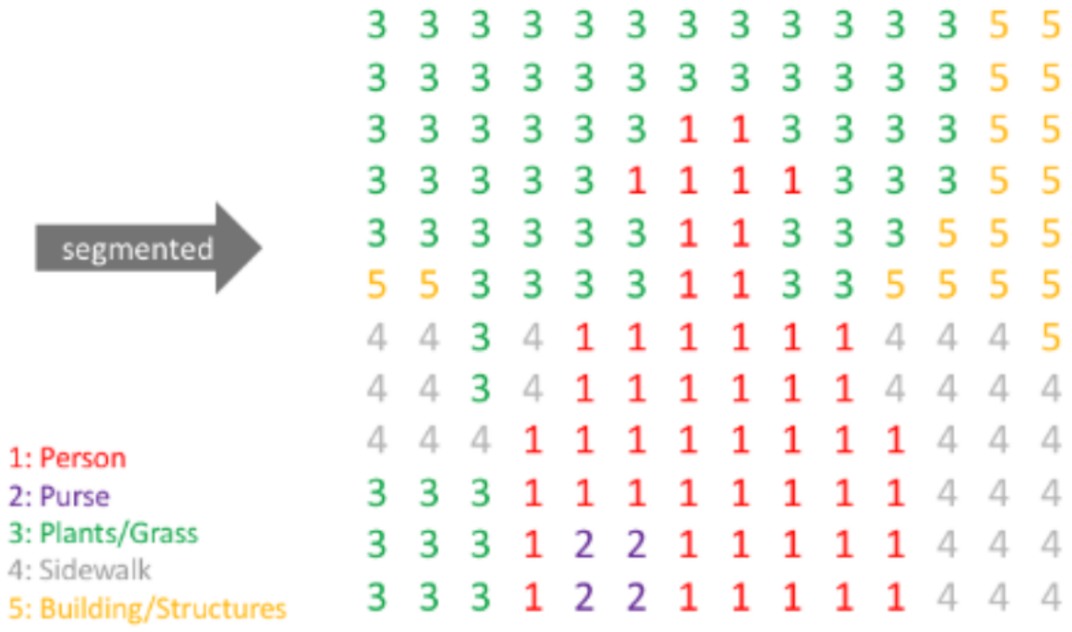
Mask Encoding

- Masks were formatted in **RGB** (as the images)
- **classes** dataframe contains the RGB mapping for each class
- Need to convert each pixel in a single channel format indicating the class membership; using the classes df as a mapping to maps RGB channel to a 1d categorical encoding

Classes dataframe:

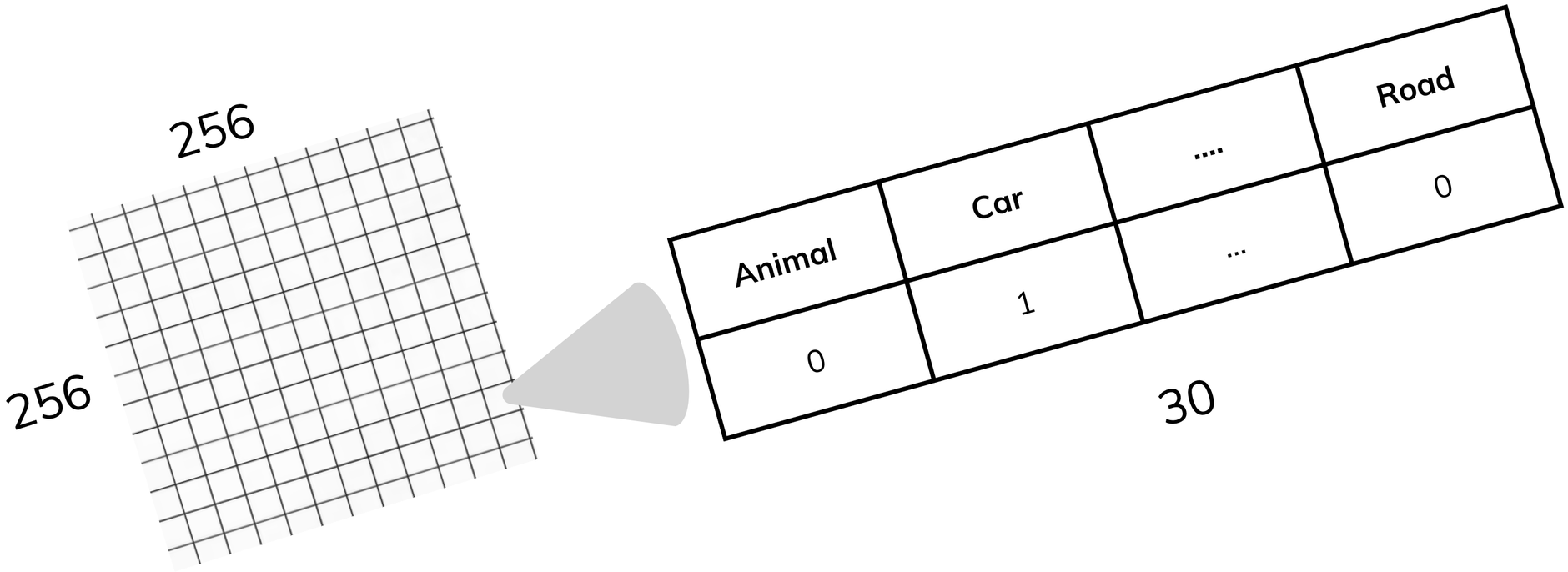
name	r	g	b	Class
Animal	64	128	64	0
Archway	192	0	128	1
Bicyclist	0	128	192	2
Bridge	0	128	64	3
Building	128	0	0	4

```
adj_masks(class_mapping, masks)
```

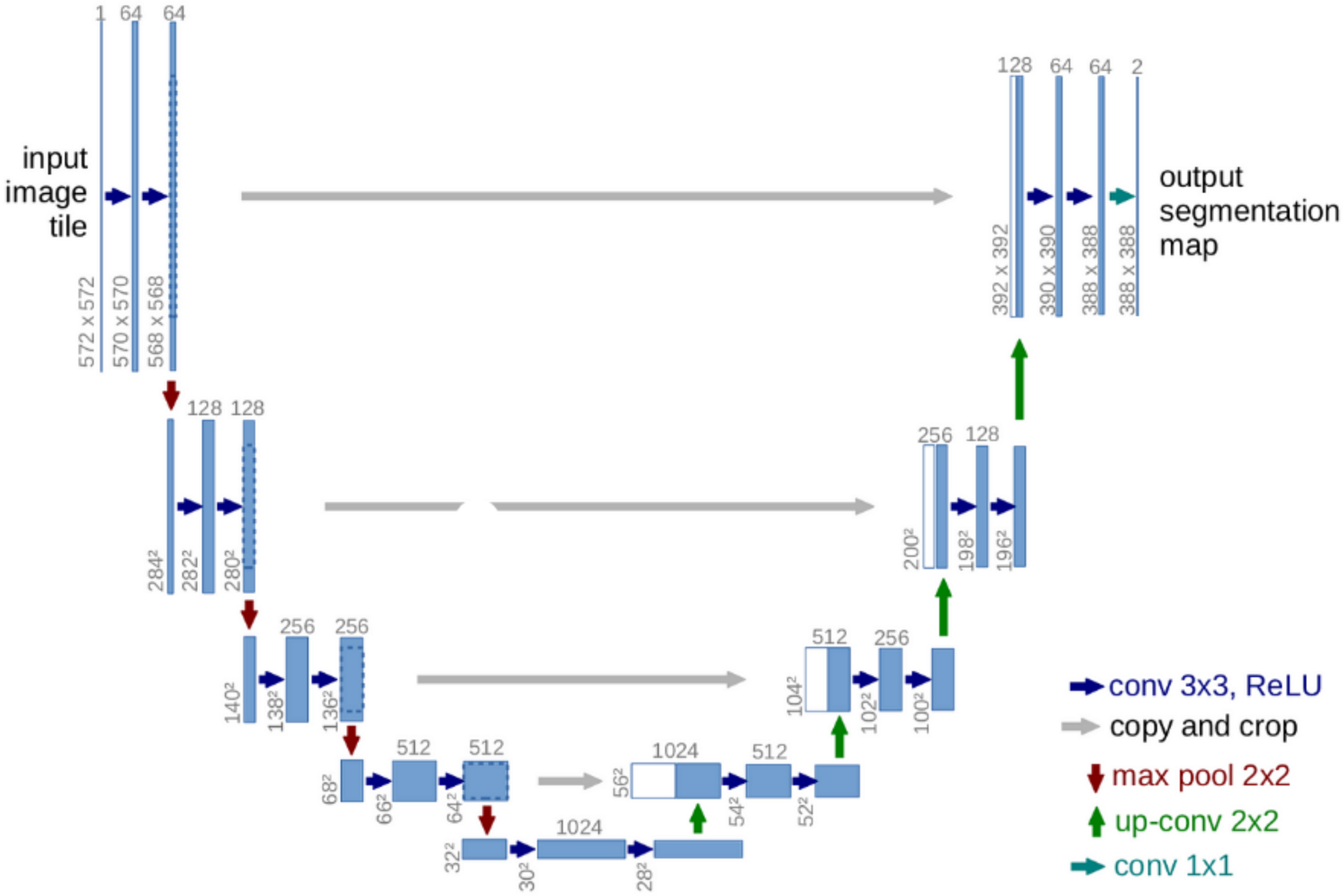


Mask Encoding

- After this we converted the single-channel images into a 30-channel images with to_categorical function of Keras
- Each image becomes an array with dimensions (256, 256, 30)
- The last axis is in the form of **one-hot-encoding** : each pixel have 30 channels in binary format with only one with a positive value

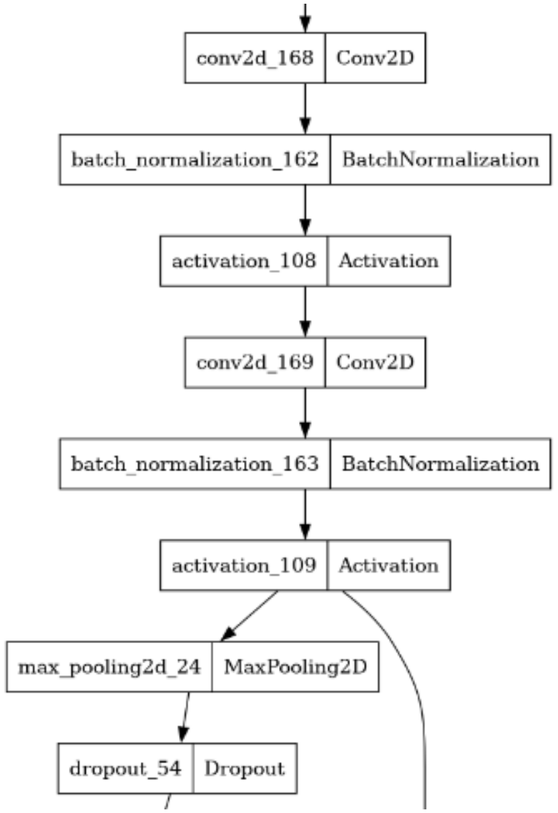


Model - Unet

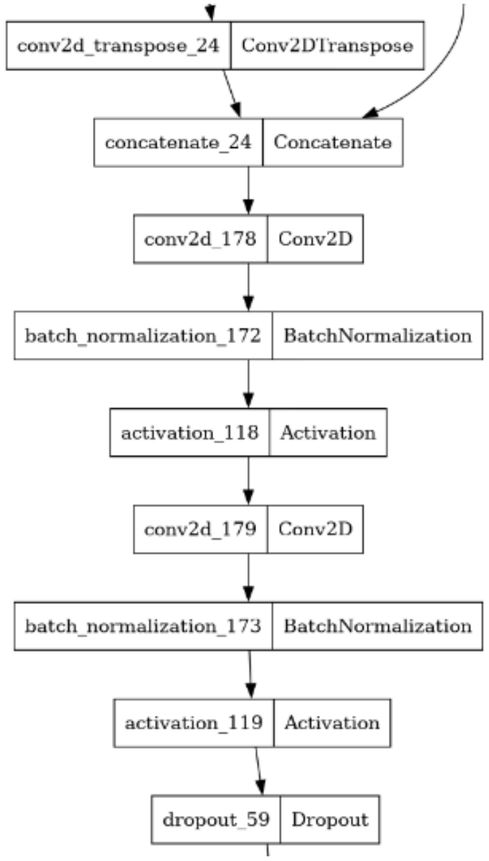

input image tile
output segmentation map
conv 3x3, ReLU
copy and crop
max pool 2x2
up-conv 2x2
conv 1x1

Architecture Blocks

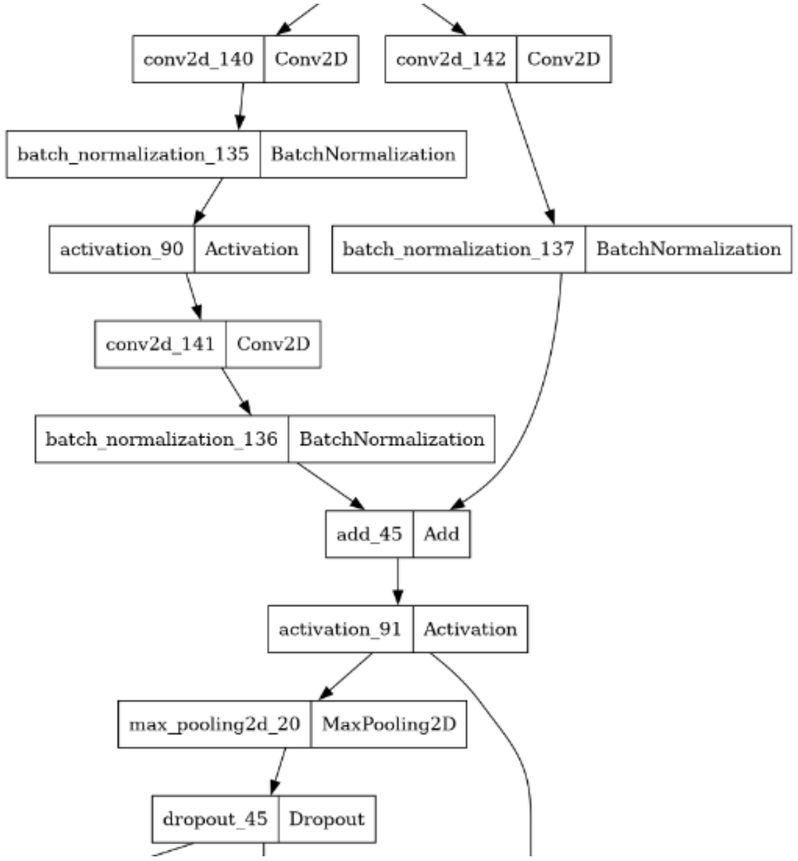
Contraction Block



Expansive Block

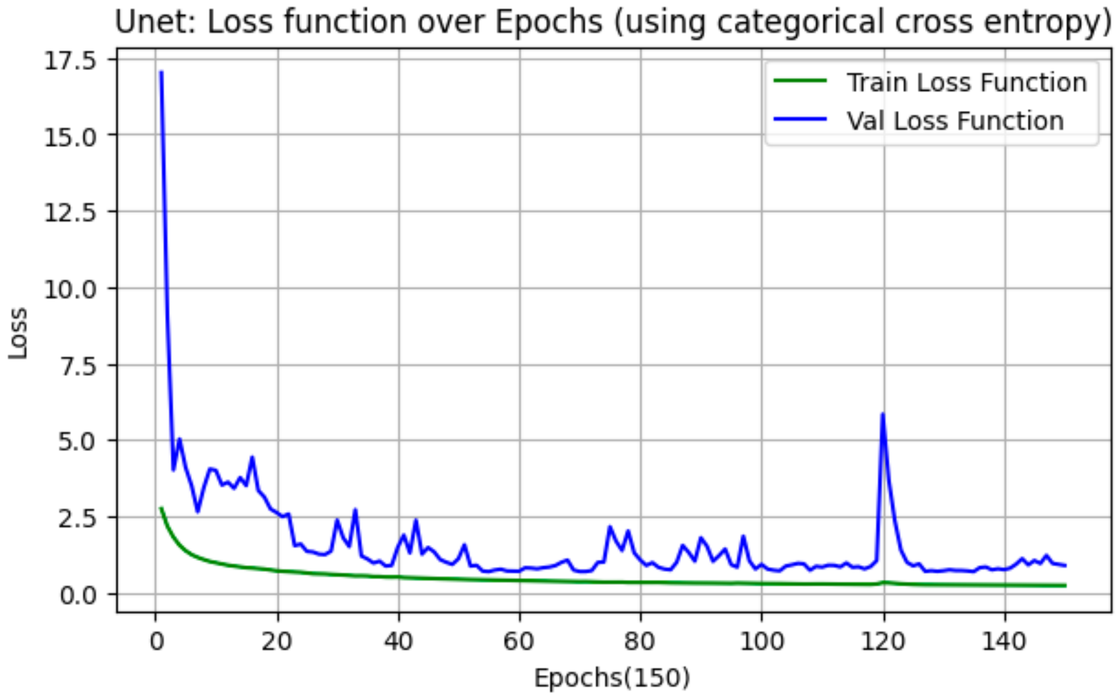


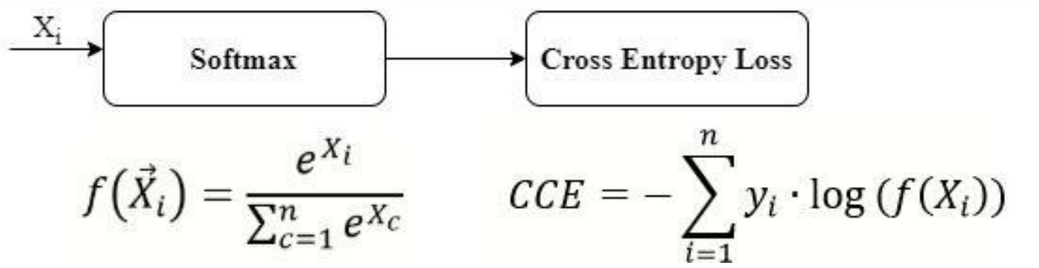
Residual Block



for ResUnet architecture

Unet - Training

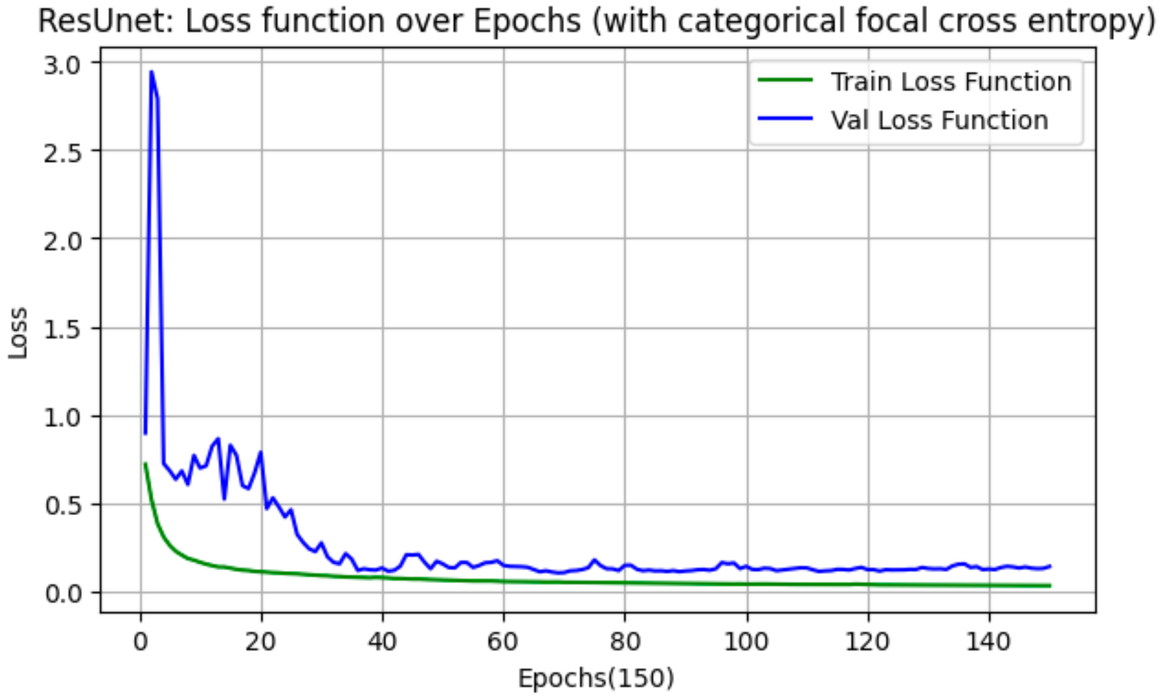




- **Optimizer:** Adam
- **Final activation:** softmax
- **Loss:** Categorical Cross-Entropy
- **Total Params:**
  - 2162142 trainable
  - 2944 non-trainable
- **Batch size:** 16
- **Epochs:** 150

	Loss	Accuracy
Train	0.322114	0.903724
Val	0.887552	0.807691
Test	0.640518	0.848687

ResUnet - Training



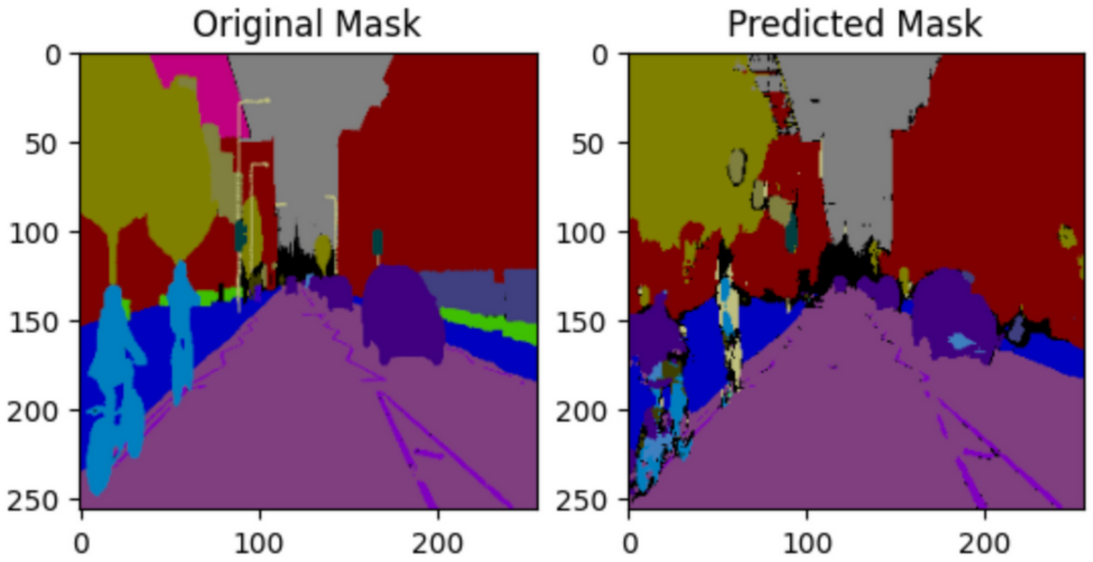
$$\mathrm{FL}(p_t) = -\alpha_t (1 - p_t)^{\gamma} \log(p_t).$$

- **Optimizer:** Adam
- **Final activation:** softmax
- **Loss:** Categorical Focal CE
- **Total Params:**
  - 2251438 trainable
  - 4416 non-trainable
- **Batch size:** 16
- **Epochs:** 150

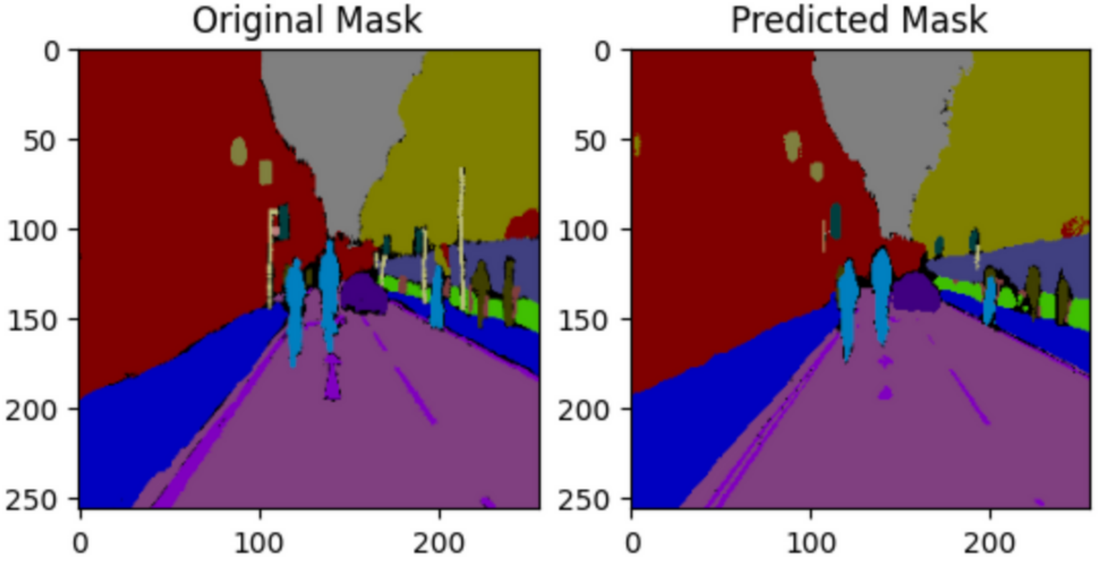
	Loss	Accuracy
Train	0.027418	0.924424
Val	0.138663	0.829723
Test	0.089534	0.868367

Results - comparison

Unet

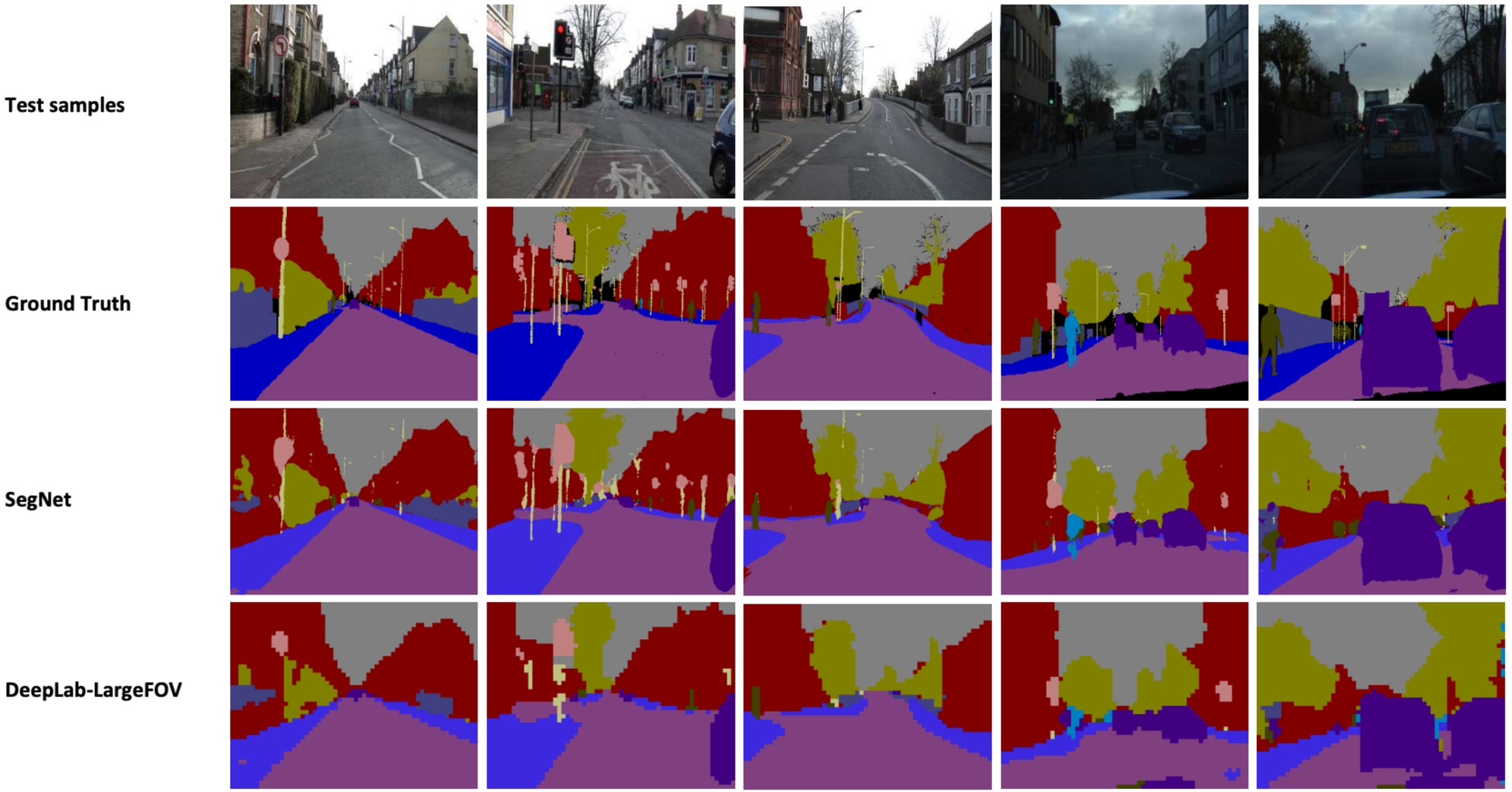


ResUnet



	Loss	Dice	IoU	Accuracy
ResUnet	0.0895	0.7710	0.6274	0.8684
Unet	0.6405	0.8356	0.7177	0.8487

Comparison with SOTA Models



Ref: *"SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation "*

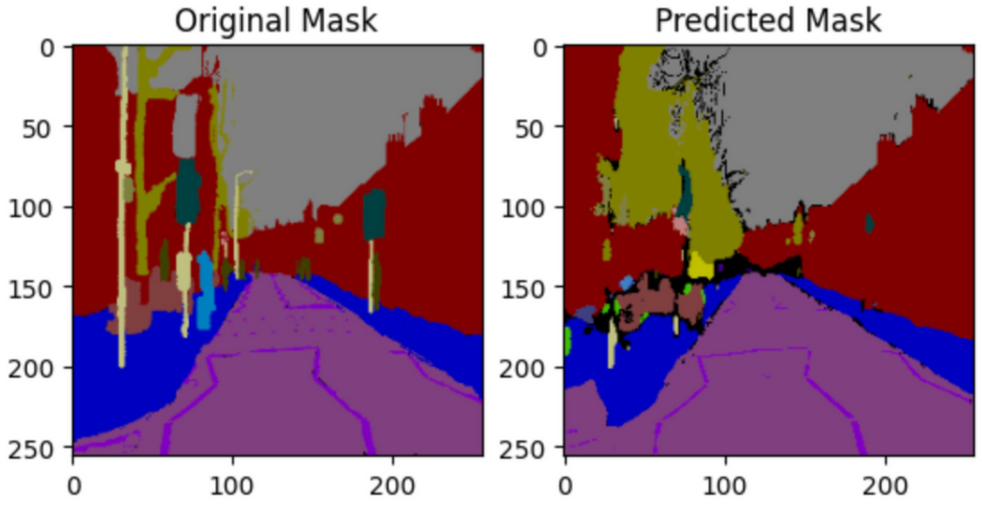
Conclusions - a few comments

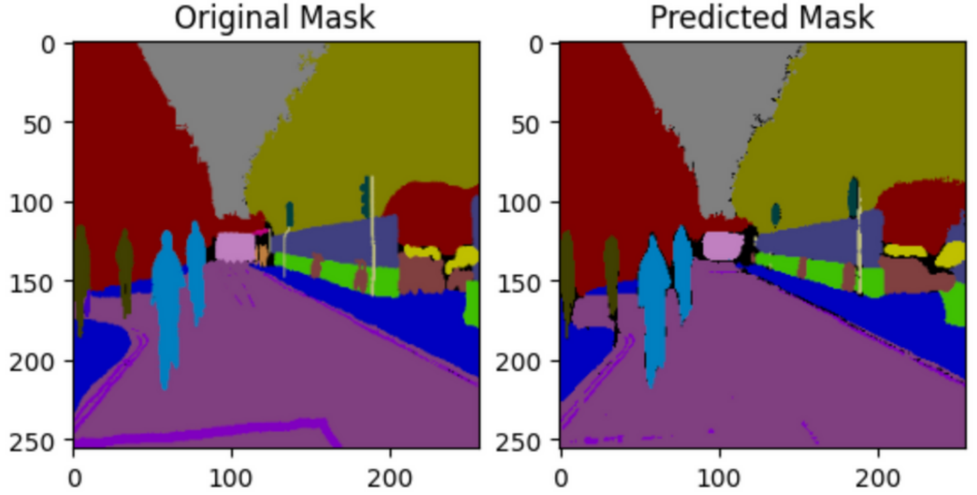
Model weaknesses:

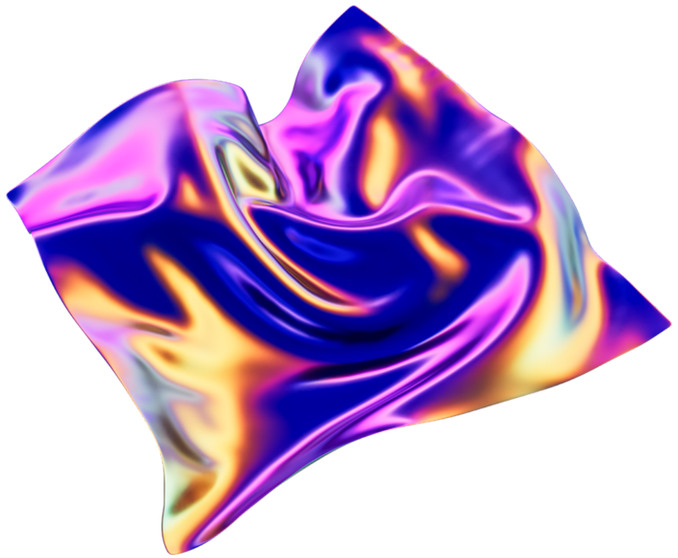
- Struggle with images with a lot of classes (very crowded by objects not confined but extremely mixed)

Model strengths:

- Performs notably better with images in which objects are well-confined
- Very good generalization on unseen images
- Really good at capturing small details (ResUnet)
- Impressive capabilities given the limited sample size



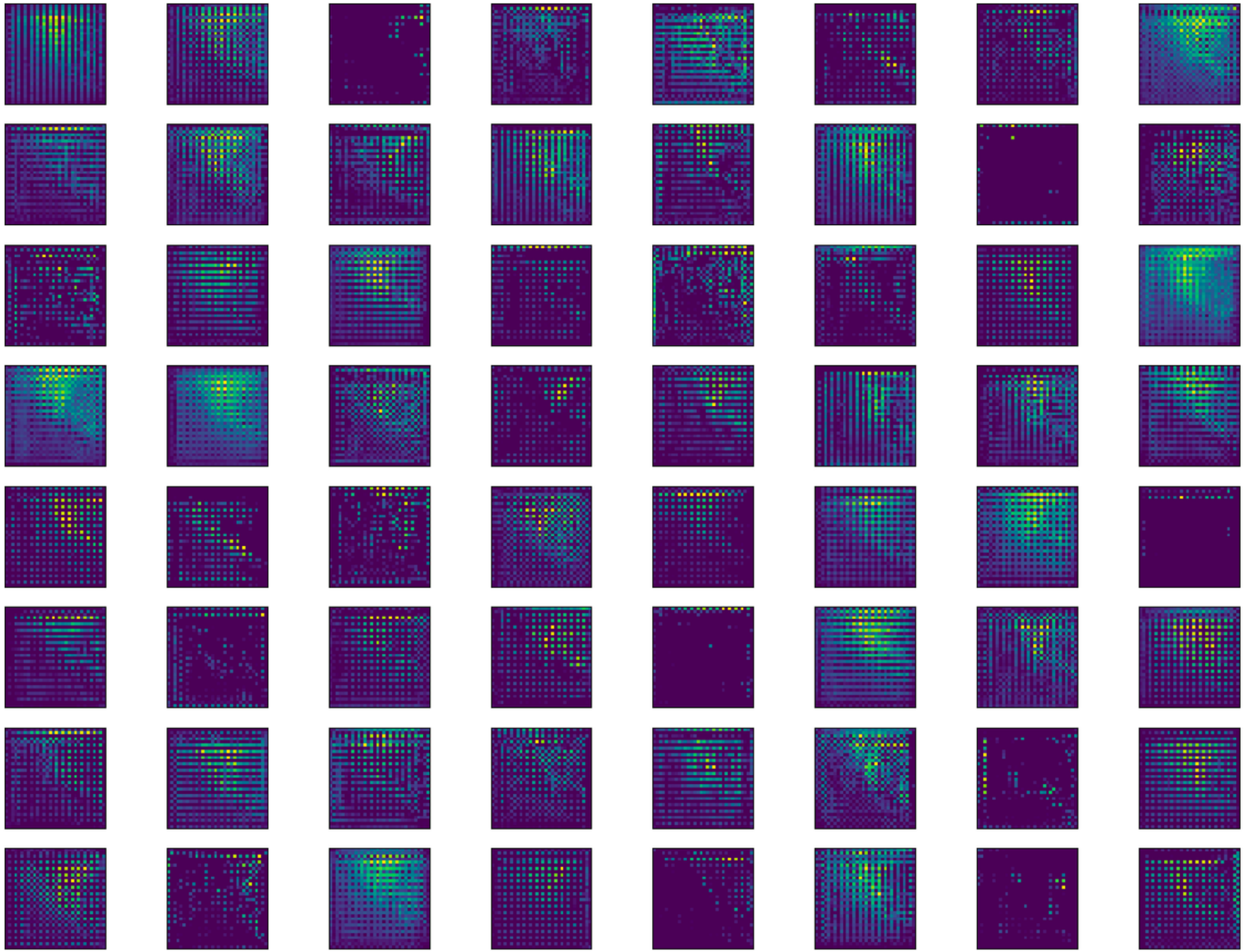




two more things...

A transfer learning-based approach

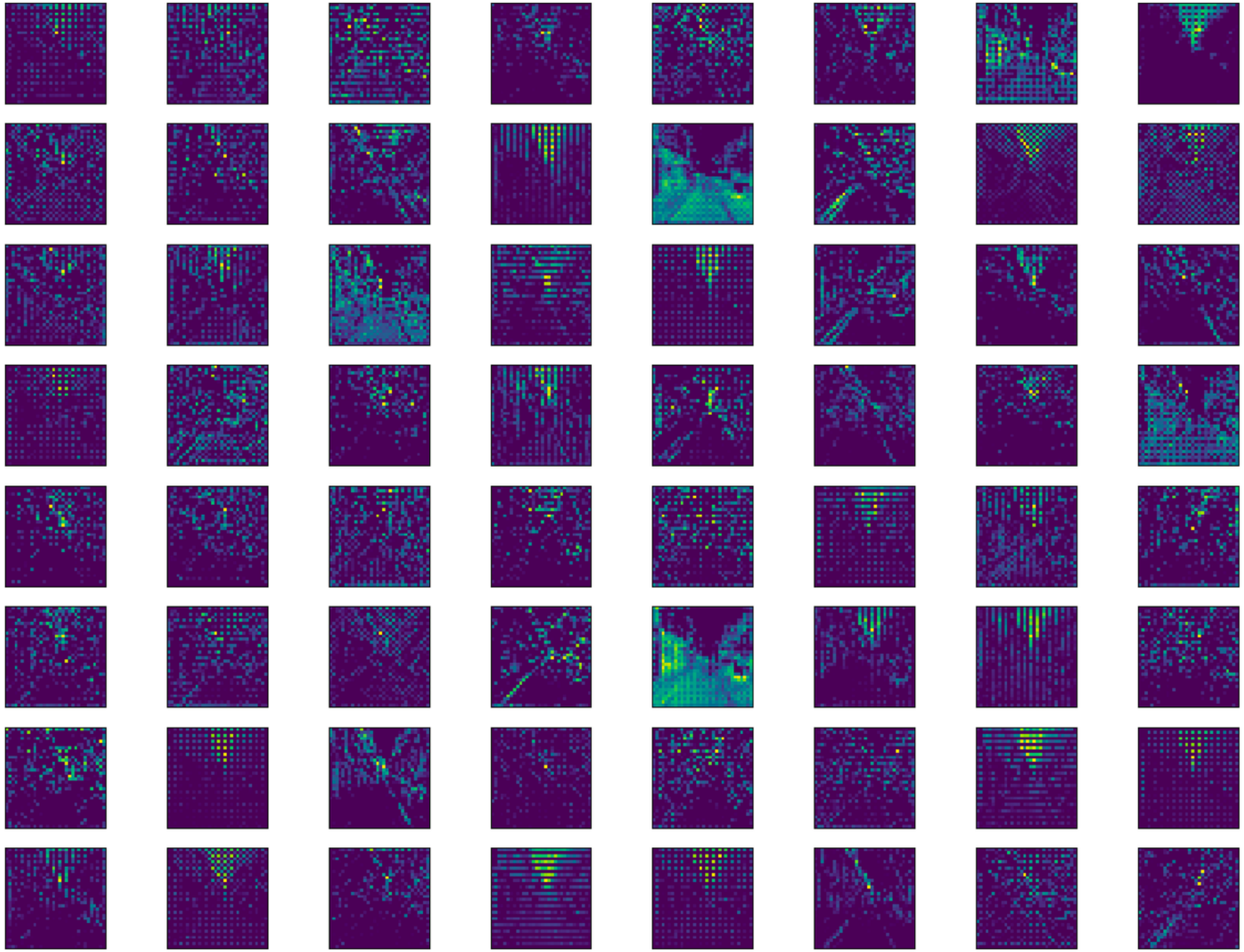
- featuring AUTOENCODERS



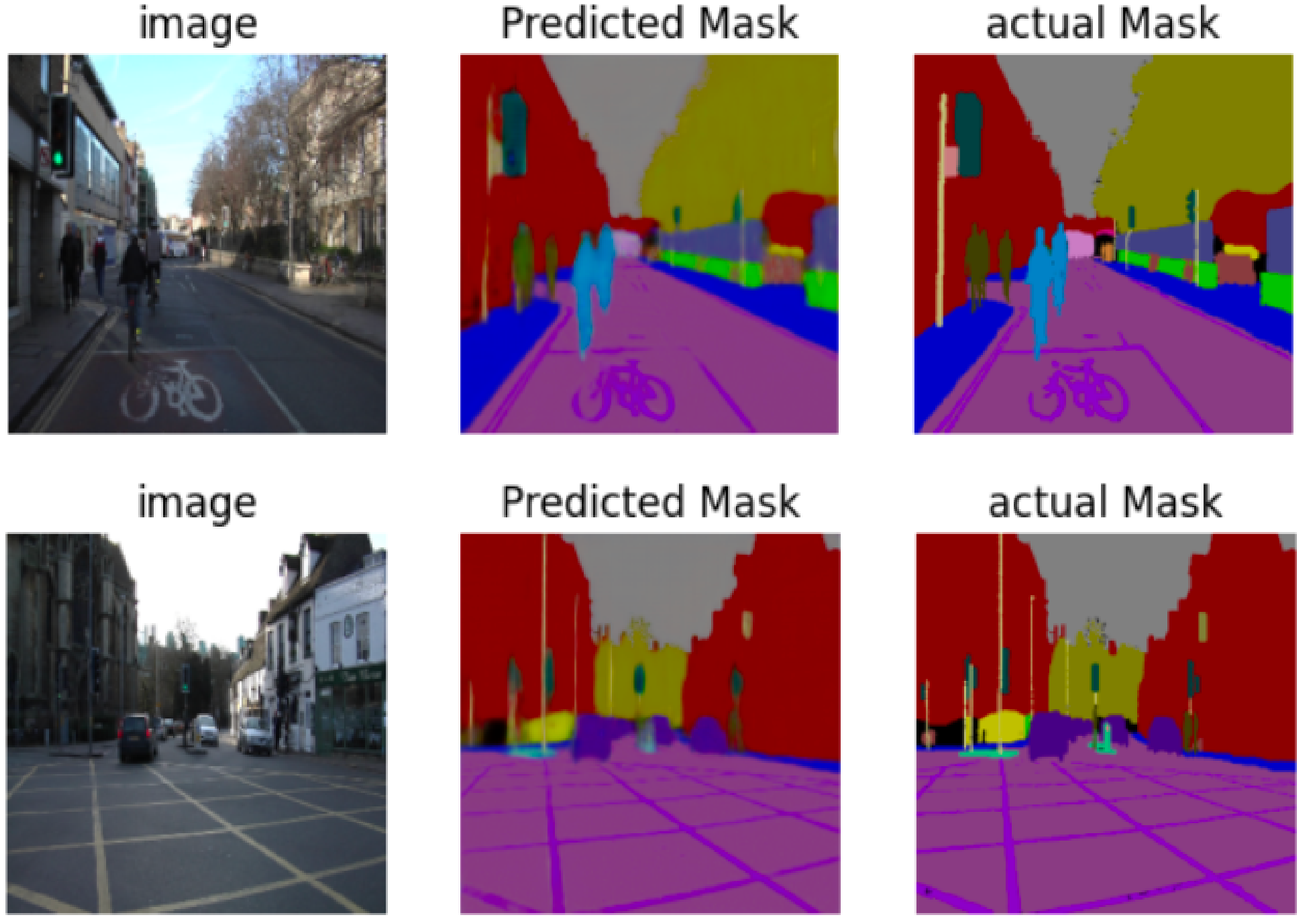
Random

A transfer learning-based approach

Pre-trained



A different approach



- No MASKS encoding
- Target: prediction of the RGB channel
- Architecture
  - ResUnet
  - Loss: Binary Cross Entropy
  - Activation function: sigmoid