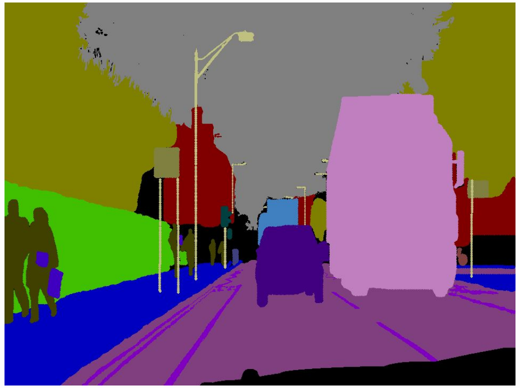
CamVid Dataset Segmentation with Deep Learning

Julius Maliwat
16/07/2024

Introduction to Semantic Segmentation



Definition: Semantic segmentation is the process of partitioning an image into segments where each pixel is assigned a label corresponding to a specific object or class.

Goal: The primary goal is to understand and label every pixel in an image.

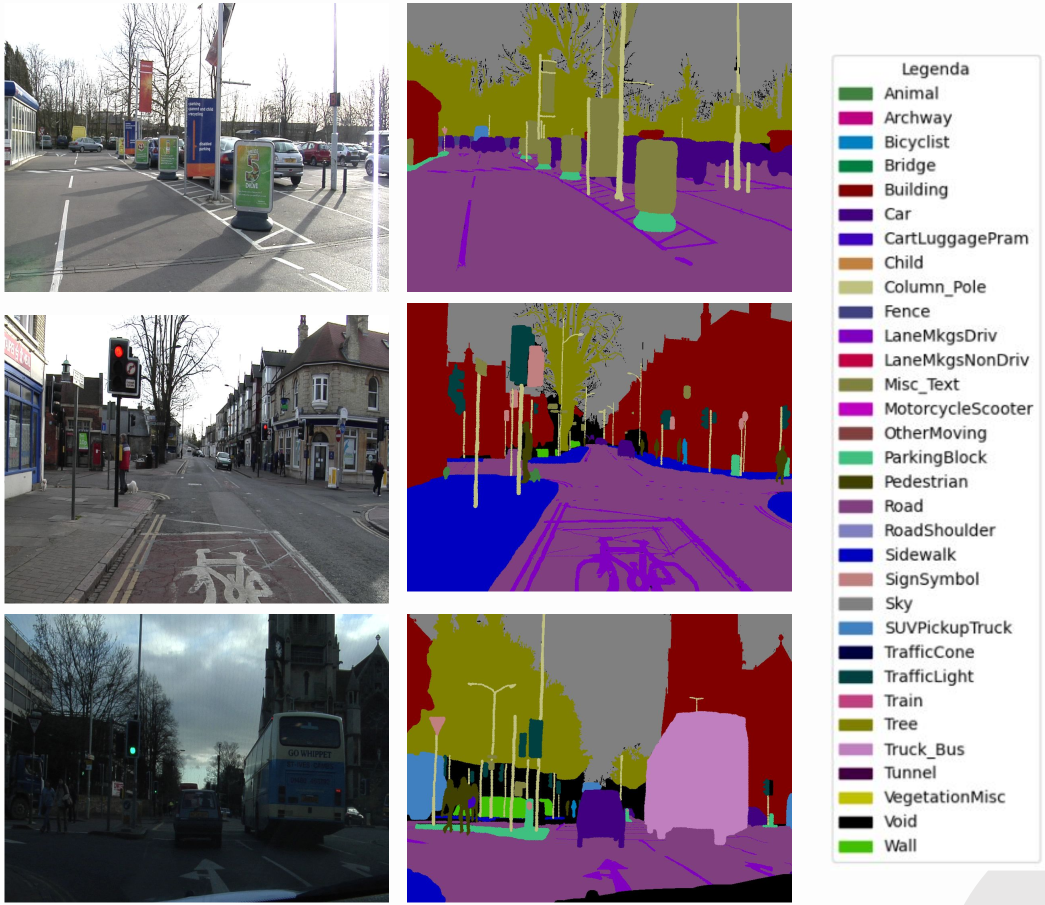
Dataset: CamVid

Origin: The CamVid dataset is a collection of images extracted from videos with corresponding pixel-wise annotations, containing urban scenes captured from a driving perspective.

Classes: The dataset includes 32 different classes such as cars, pedestrians, trees, roads, etc.

Number of Images: The dataset contains 701 labeled frames

Resolution: Images are provided at a resolution of 720x960 pixels.



Study objectives

Objective 1

Train a semantic segmentation model to classify all 32 classes in the CamVid dataset.

Objective 2

Develop a model to classify a standardized set of 11 classes, as commonly used in research, to ensure comparability with state-of-the-art methods.

Grouped Class Categories

The 11 classes were chosen to align with common practices in research papers, facilitating direct comparison with state-of-the-art methods. This approach is prevalent in semantic segmentation tasks to reduce model complexity and to focus on the most relevant and significant classes for the given application.

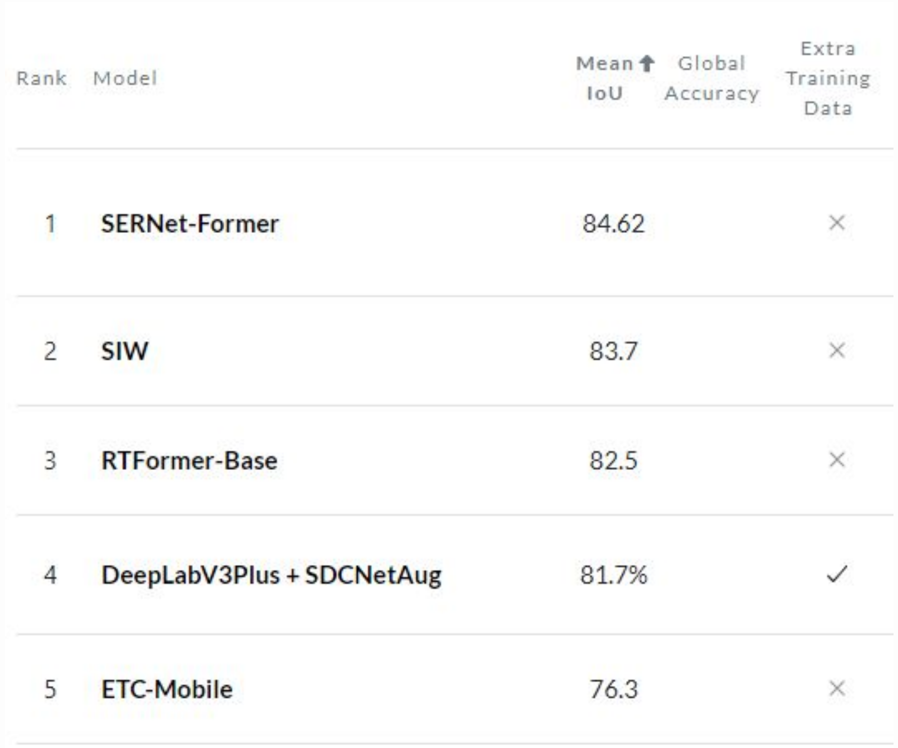
- **Sky**: Sky
- **Building**: Archway, Bridge, Building, Tunnel, Wall
- **Pole**: Column_Pole, TrafficCone
- **Road**: Road, LaneMkgsDriv, LaneMkgsNonDriv
- **Sidewalk**: Sidewalk, ParkingBlock, RoadShoulder
- **Tree**: Tree, VegetationMisc
- **Sign**: TrafficLight, Misc_Text, SignSymbol
- **Fence**: Fence
- **Car**: Car, OtherMoving, SUVPickupTruck, Train, Truck_Bus
- **Pedestrian**: Animal, CartLuggagePram, Child, Pedestrian
- **Bicyclist**: Bicyclist, MotorcycleScooter

Rationale for Simplifying to 11 Classes

Reduced Complexity: Simpler model, faster training, and inference.

Improved Performance: Better focus on significant classes.

Enhanced Relevance: Emphasis on critical elements like cars, poles, pedestrians, and roads.

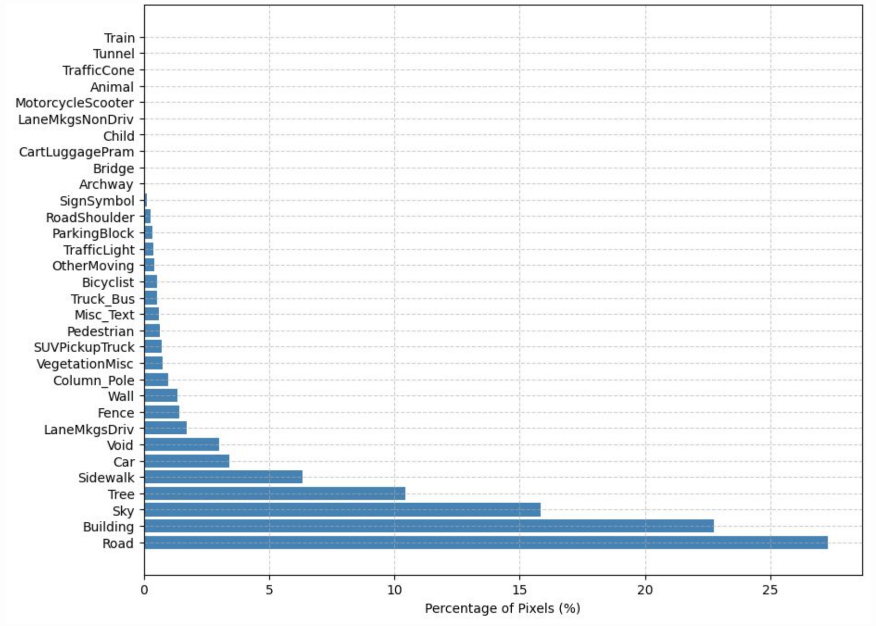


Rank	Model	Mean ↑ IoU	Global Accuracy	Extra Training Data
1	SERNet-Former	84.62		×
2	SIW	83.7		×
3	RTFormer-Base	82.5		×
4	DeepLabV3Plus + SDCNetAug	81.7%		✓
5	ETC-Mobile	76.3		×

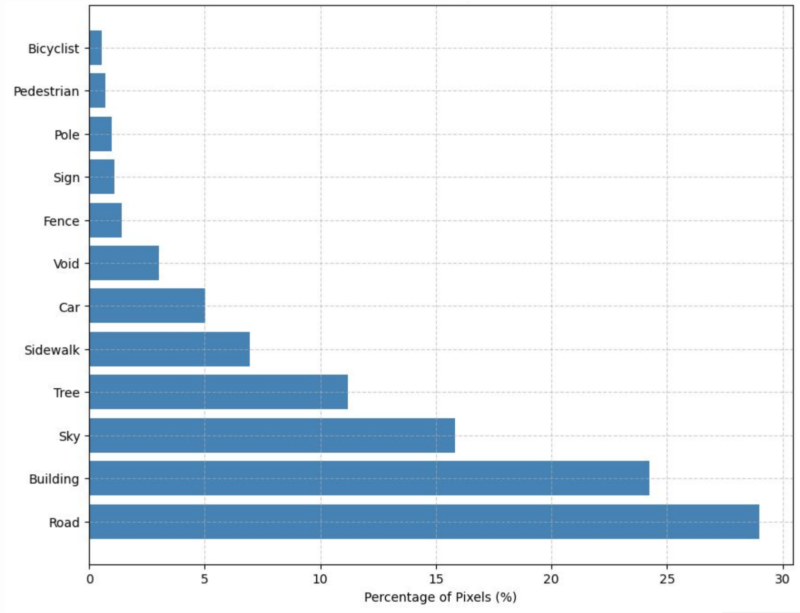
Pixel Class Distribution

Number of observations: 701
Resolution : 720x960 pixels

32 Classes



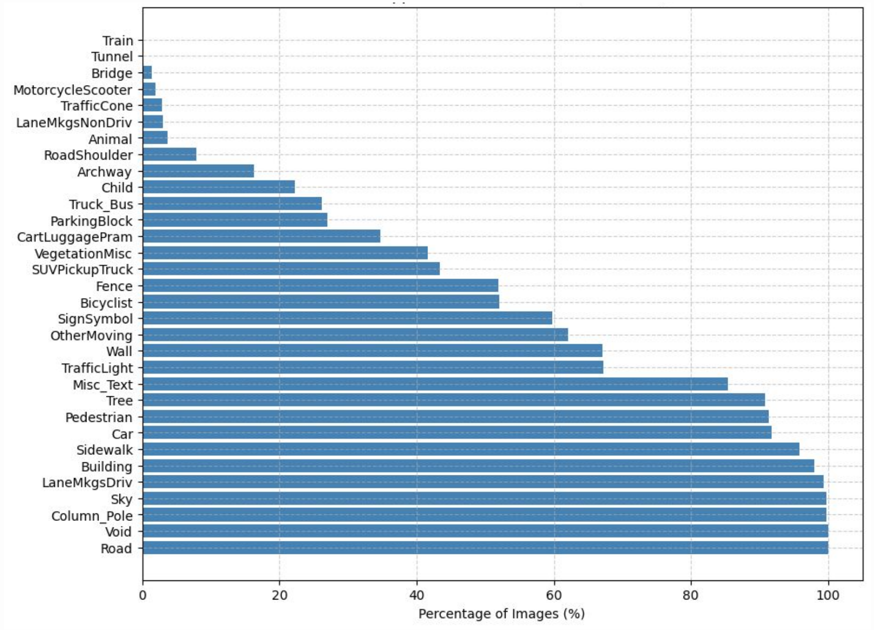
11 Classes (+ void)



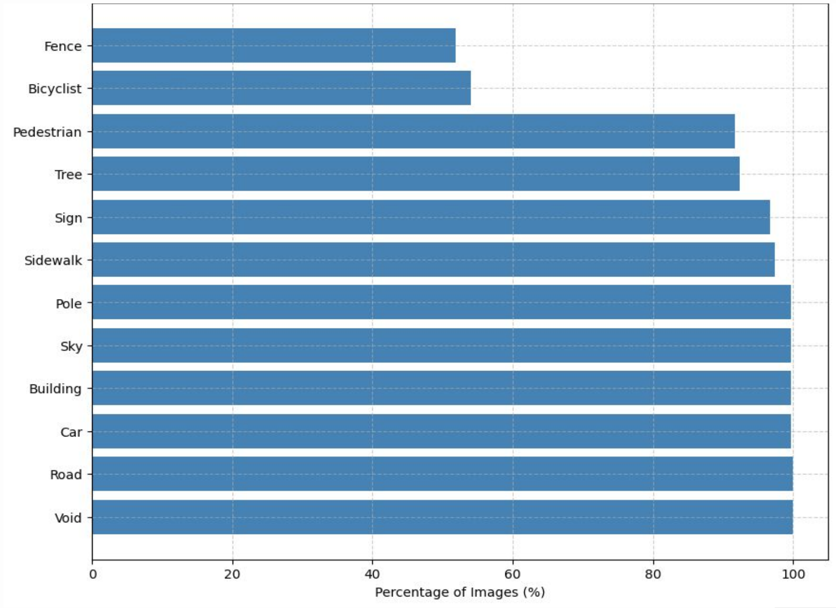
Class Appearance Distribution

Number of observations: 701
Resolution : 720x960 pixels

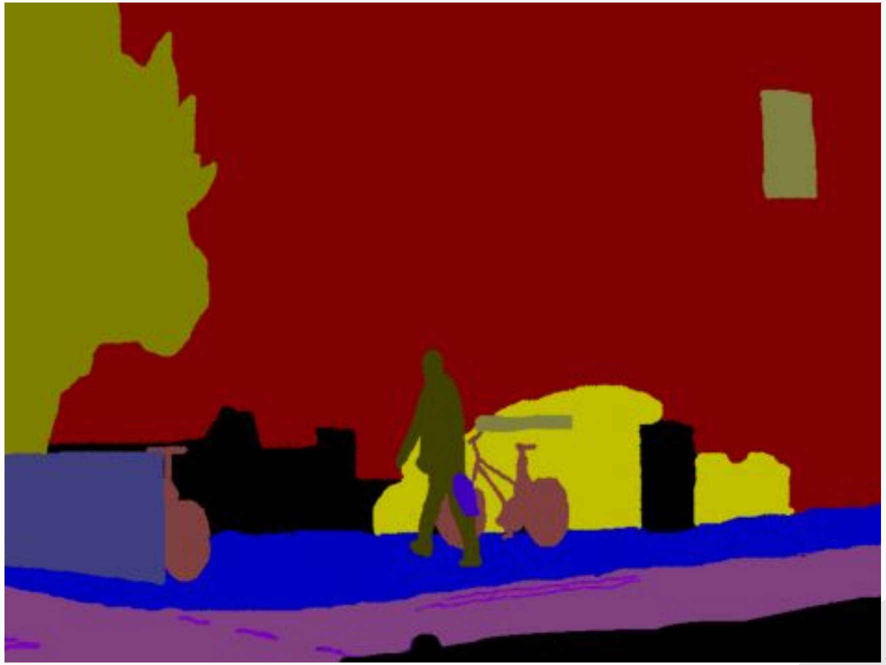
32 Classes



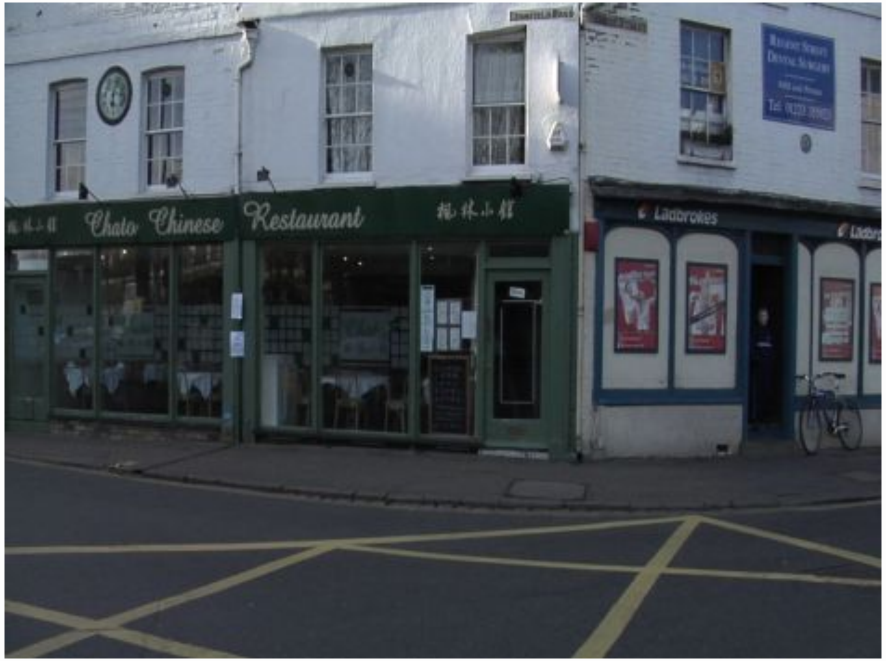
11 Classes (+ void)

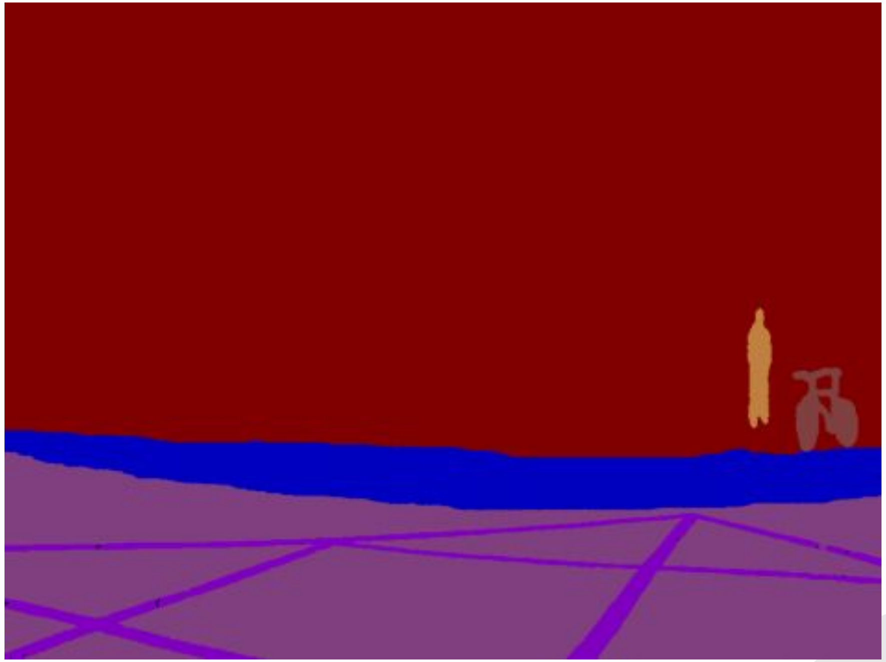


No Sky Observations



No Sky Observations





Data Preprocessing steps

- **Resizing:** Images/masks resized to 256x256 pixels
- **Splitting:**
  - Initial: 80% train, 20% test
  - Train split: 80% train, 20% validation
- **Normalization:** Pixel values scaled to [0, 1]
- **Augmentation:** horizontal flipping (enhance generalization)

Note: The class appearance distribution was maintained consistently across train, validation, and test sets.

Training Setup

Hyperparameters

- **Batch Size**: 16
- **Epochs**: 100
- **Optimizer**: Adam
- **Learning Rate**: 0.001

Loss Function

Categorical Crossentropy

Callbacks

- **EarlyStopping**: patience of 20 on validation mean IoU)
- **ModelCheckpoint**: save the best model based on validation mean IoU
- **CSVLogger** (log metrics)
- **ReduceLROnPlateau**: Reduce learning rate by a factor of 0.1 when validation loss plateaus

Model Evaluation Metrics

Mean IoU (Intersection over Union)

Chosen as the primary metric because it gives a balanced view of performance across all classes, highlighting both common and rare classes. This is particularly important for the CamVid dataset, which is highly imbalanced with many underrepresented classes. For the calculation of mean IoU for the 11 classes, the class "Void" is excluded from the calculation.

IoU for each class

Helps identify and address specific classes where the model may be underperforming, crucial for handling class imbalance in the CamVid dataset

Accuracy

While useful for a general overview, accuracy can be misleading in imbalanced datasets, which is why it is considered secondary to IoU metrics.

Rare Class Problem

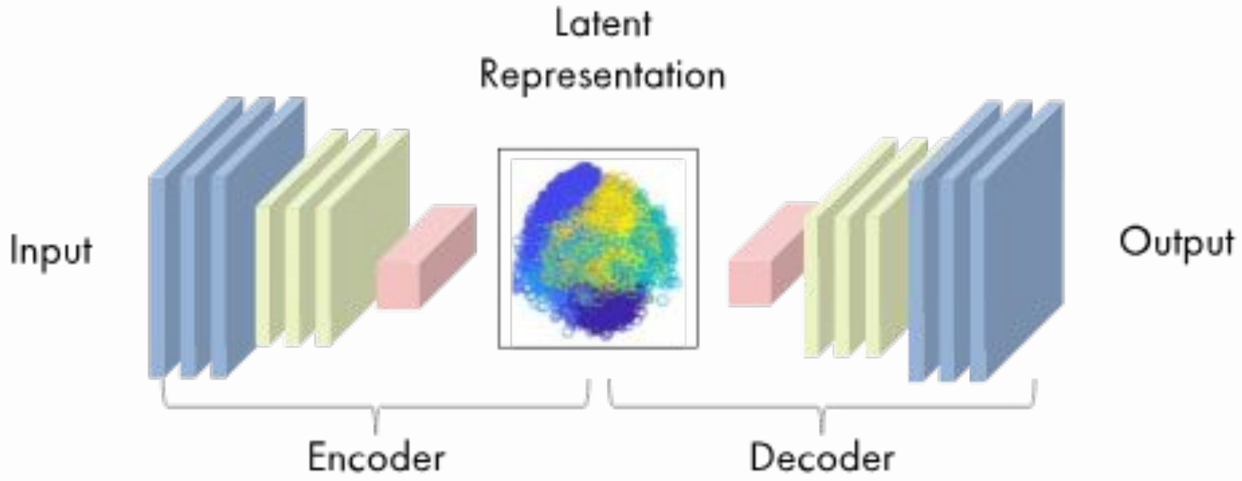
Class: Tunnel

- **Occurrence:** The "Tunnel" class appears only once in the entire dataset.
- **Training Set:** It appears once in the training set.
- **Validation and Test Sets:** It does not appear at all in the validation and test sets.
- **Impact on Mean IoU:** Since it does not appear in the validation and test sets, it is not considered in the mean IoU calculation.

Class: Train

- **Occurrence:** The "Train" class does not appear in any of the observations, despite being defined as a class.
- **Impact on Mean IoU:** It is excluded from the mean IoU calculation due to its complete absence in the dataset.

Model Architecture: Autoencoders



Starting from a simple autoencoder, we iteratively increased model complexity by adding skip connections, more layers, increasing the number of filters, and applying data augmentation techniques.

Iterative Model Improvements (32 classes)

01

Model 1

Filters: Starts at 32, up to 64

Convolution layers per block: 1

No Skip connections

02

Model 2

Filters: Starts at 32, doubles each block, up to 256

Convolution layers per block: 2

03

Model 3

Filters: Starts at 32, doubles each block, up to 512

Convolution layers per block: 2

04

Model 4

Filters: Starts at 32, doubles each block, up to 1024

Convolution layers per block: 2

05

Model 5

Filters: Starts at 64, doubles each block, up to 2048

Convolution layers per block: 2

06

Model 5 (augmented)

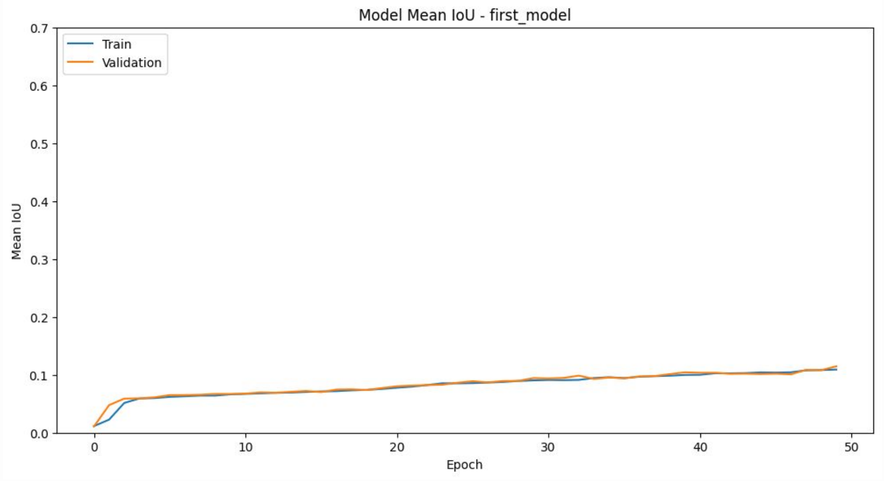
Model 5 with horizontal flip augmentation

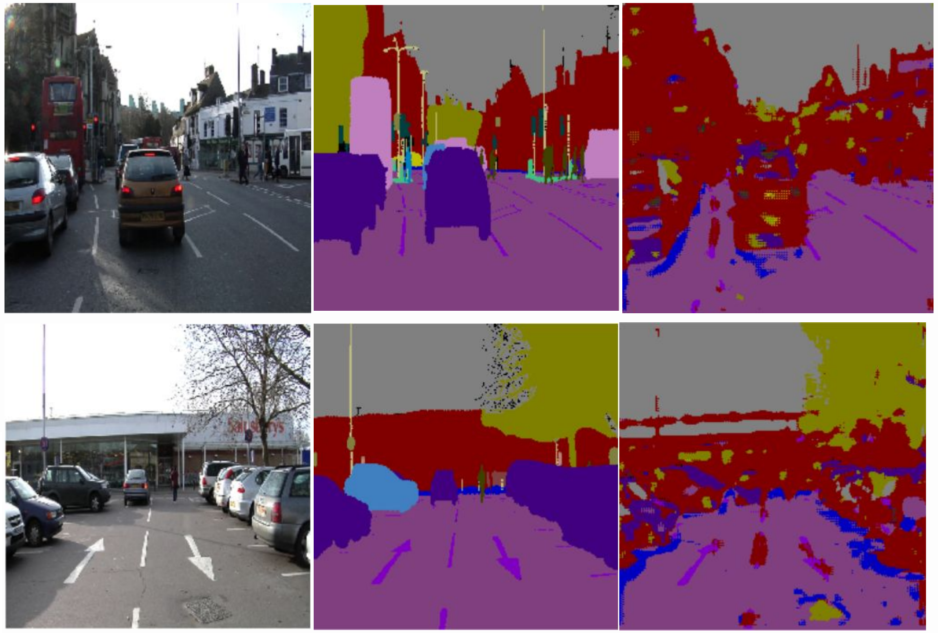
The architecture includes convolutional, max pooling, and Conv2DTranspose layers. Filters double at each block up to the core. Skip connections are added from Model 2 onwards.

Model 1 (Validation)

Mean IoU (%)	**Accuracy(%)**
11,48	68,49

Parameters = 75840





Observations:
- Low Mean Iou
- Model is not complex enough

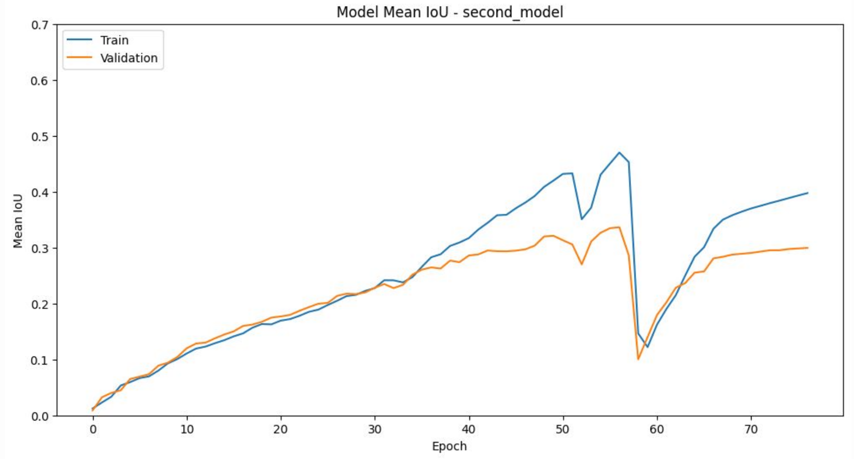
Solution:
- Add 2 encoder and 2 decoder blocks
- Add 1 conv Layer for each block
- Add skip connections

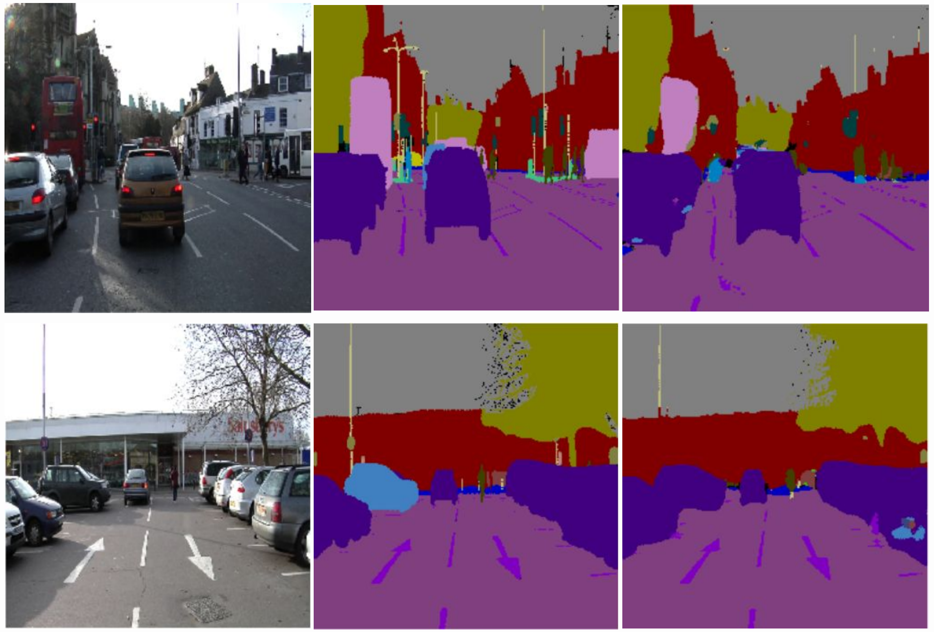
Model 2 (Validation)

Mean IoU (%)	**Accuracy(%)**
33,65 (**+22.17**)	86,80 (**+18.31**)

Parentheses indicate changes from Model 1.

Parameters = 2141664





Observations:

- Training instability towards the end
- Significant improvement in Mean IoU
- Struggles with small details in segmentation

Solution:

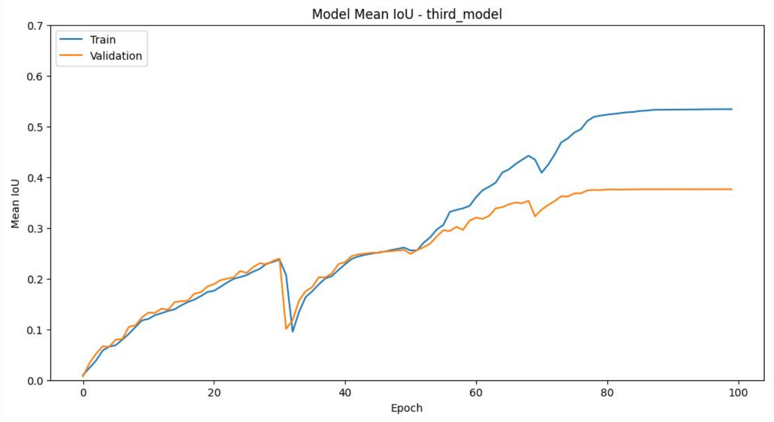
- Increase complexity by Adding 1 encoder and 1 decoder block

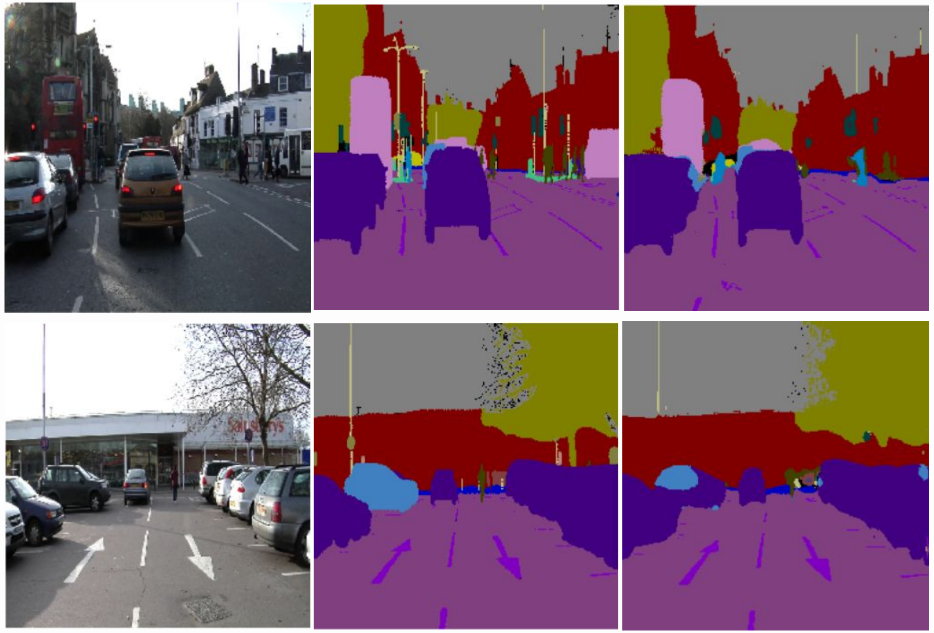
Model 3 (Validation)

Mean IoU (%)	**Accuracy(%)**
37,61 (**+3.96**)	88,39 (**+1.59**)

Parentheses indicate changes from Model 2

Parameters = 8631520





Observations:

- Continued instability during training
- Further improved Mean IoU
- Still struggles with fine details

Solution:

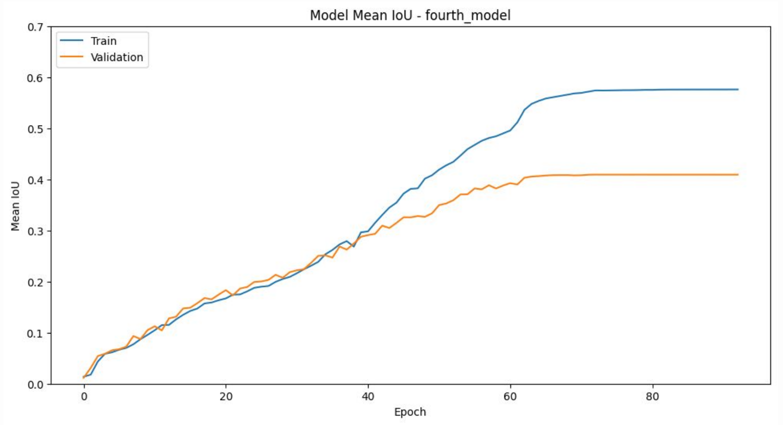
- Add 1 encoder and 1 decoder block

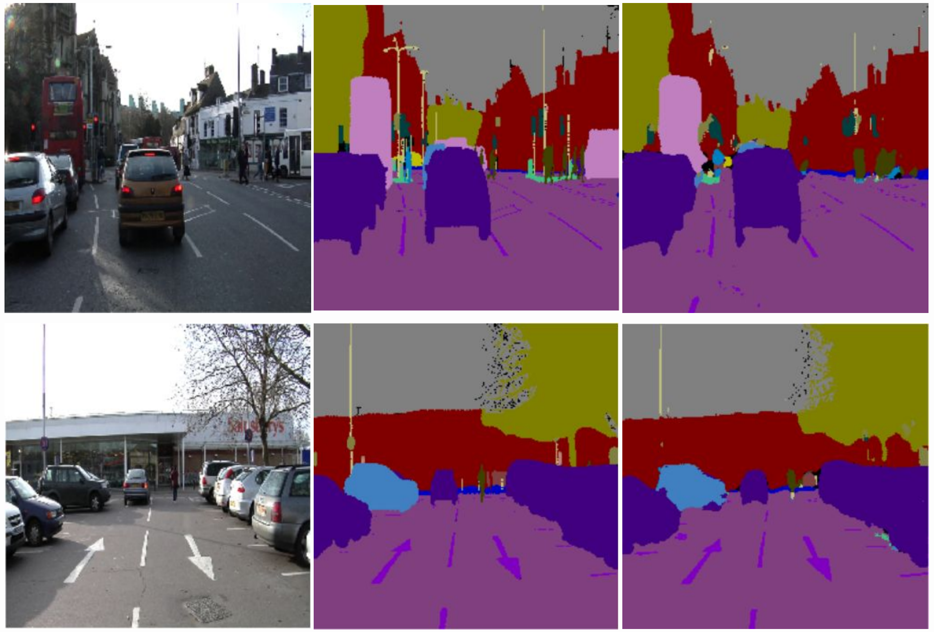
Model 4 (Validation)

Mean IoU (%)	**Accuracy(%)**
40,93 (**+3.32**)	89,43 (**+1.04**)

Parentheses indicate changes from Model 3

Parameters = 34587360





Observations:

- Training is more stable
- Improved Mean IoU
- Model still struggles with fine details and smaller classes

Solution:

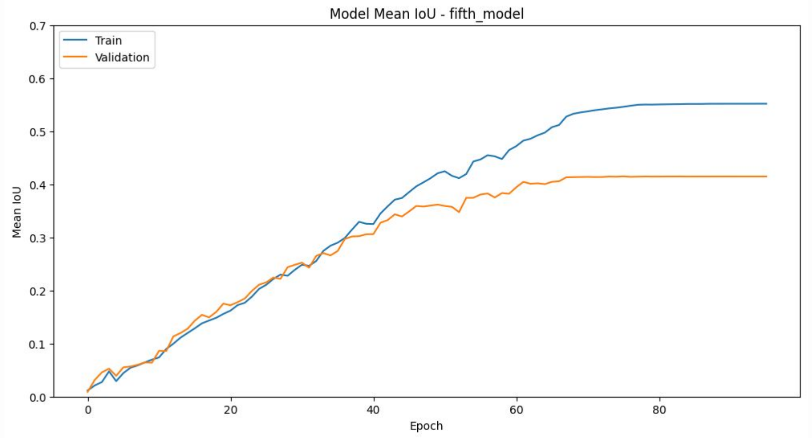
- Increase starting filters (32 to 64)

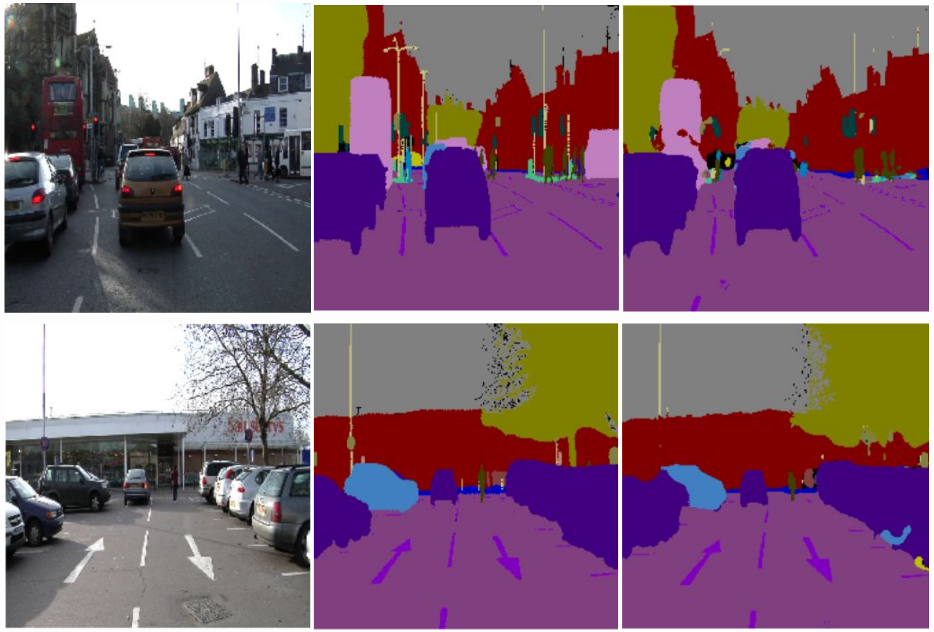
Model 5 (Validation)

Mean IoU (%)	**Accuracy(%)**
41,50 (**+0.57**)	89,74 (**+0.31**)

Parentheses indicate changes from Model 1.

Parameters = 138331552





Observations:

- Stable during training
- Slightly Improved Mean IoU
- Overfitting

Solution:

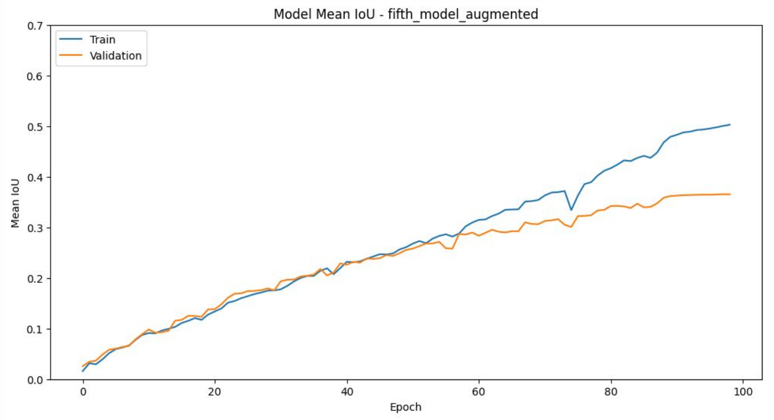
- Data Augmentation (horizontal flip)

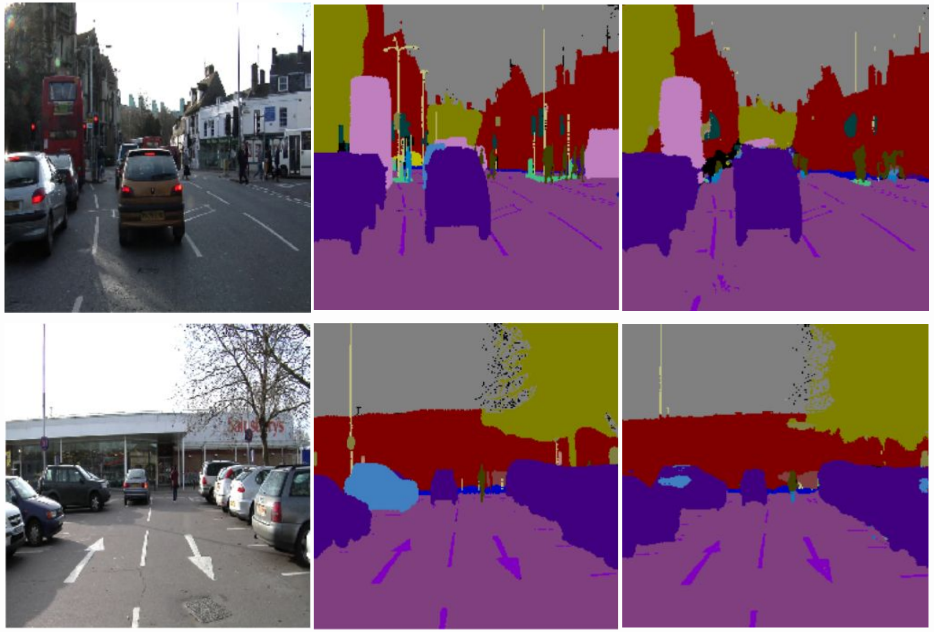
Model 5.2 (Validation)

Mean IoU (%)	Accuracy(%)
36,54 (-4.96)	87,91 (-1.83)

Parentheses indicate changes from Model 5

Parameters = 138331552



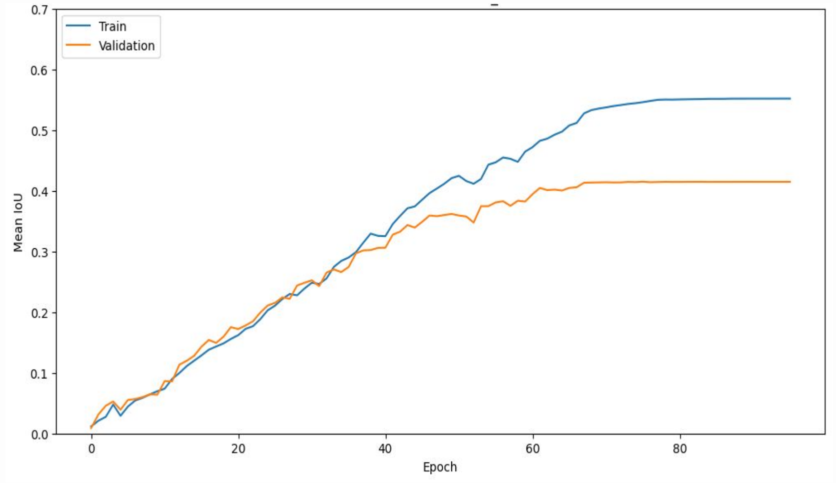


Observations:

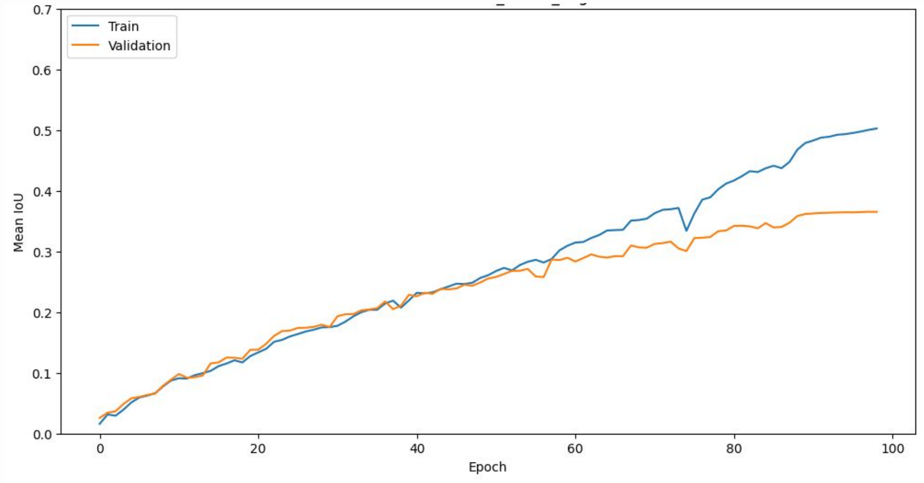
- Stable during training
- Mean IoU decreased, indicating that data augmentation did not improve model performance as expected
- Slight improvement in reducing overfitting, but overall performance dropped

Augmentation Comparison (Model 5)

Model 5



Model 5 + Augmentation



Summary Evaluation (32 Classes)(Validation)

Model	**Main Improvement**	**Description**	**Mean IoU (%)**	**Accuracy (%)**
Model 1	Baseline model	**Filters**: 32 (initial), 64 (core) **Convolution layers per block**: 1	11,48	68,49
Model 2	Deeper architecture	**Filters:** 32 (initial), 256 (core) **Convolution layers per block:** 2	33,65	86,80
Model 3	Increased depth and filters	**Filters**: 32 (initial), 512 (core) **Convolution layers per block**: 2	37,61	88,39
Model 4	Further increased depth	**Filters**: 32 (initial), 1024 (core) **Convolution layers per block**: 2	40,93	89,43
Model 5	Increased number of filters	**Filters:** 64 (initial), 2048 (core) **Convolution layers per block:** 2	41,50	89,74
Model 5.2 augmented	Data augmentation	Same architecture as Model 5. Applied horizontal flip augmentation to improve generalization	36,54	87,91

The architecture includes convolutional, max pooling, and Conv2DTranspose layers. Filters double at each block up to the core. Skip connections are added from Model 2 onwards.

Iterative Model Improvements (11 classes)

01

Model 2

Same as Model 2 (32 classes)

02

Model 3

Same as Model 3 (32 classes)

03

Model 4

Same as Model 4 (32 classes)

04

Model 4.2

Based on Model 4.
Added attention modules in decoder

05

Model 4 .3

Based on model 4.2.
Horizontal Flip Augmentation

06

Model 5

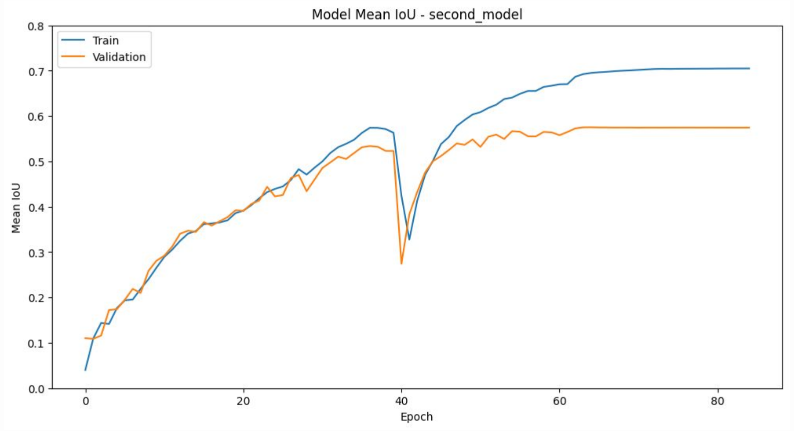
Same as model 5 (32 classes)

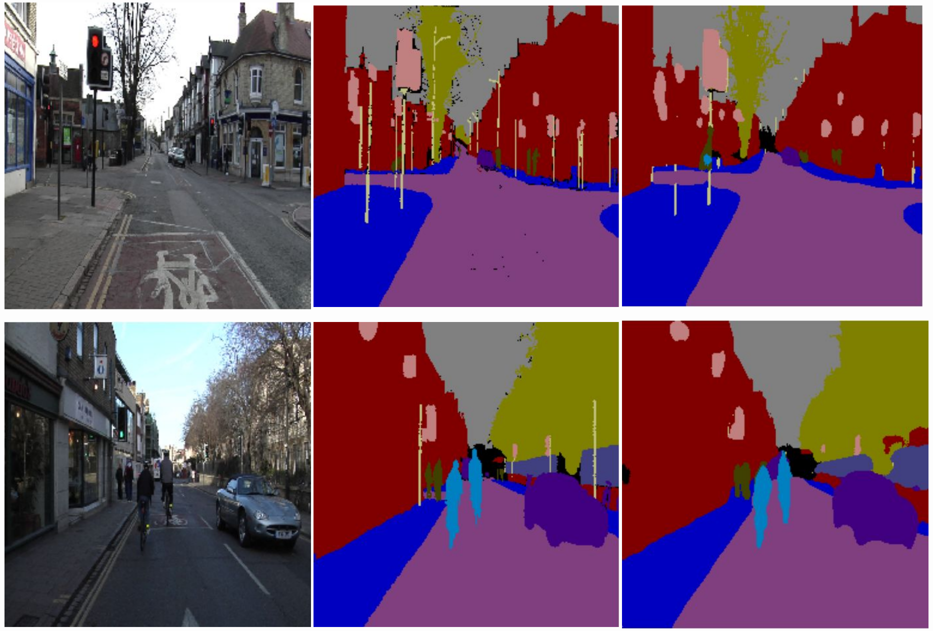
Models 2, 3, 4, and 5 for 11 classes share the same architecture as their counterparts for 32 classes, with only the number of output classes changing.

Model 2 (Validation)

Mean IoU (%)	**Accuracy(%)**
61,71	90,01

Parameters = 2141004





Observations:

- Good initial performance
- instability during training, particularly around epoch 40
- Struggles with fine details in segmentation
-

Solution:

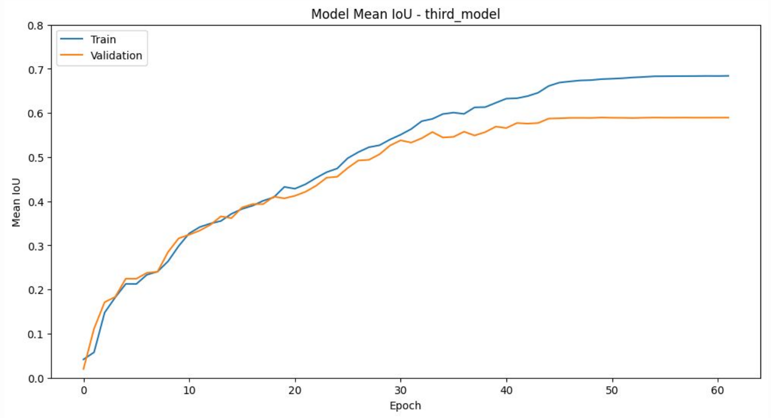
- Add 1 encoder and 1 decoder blocks

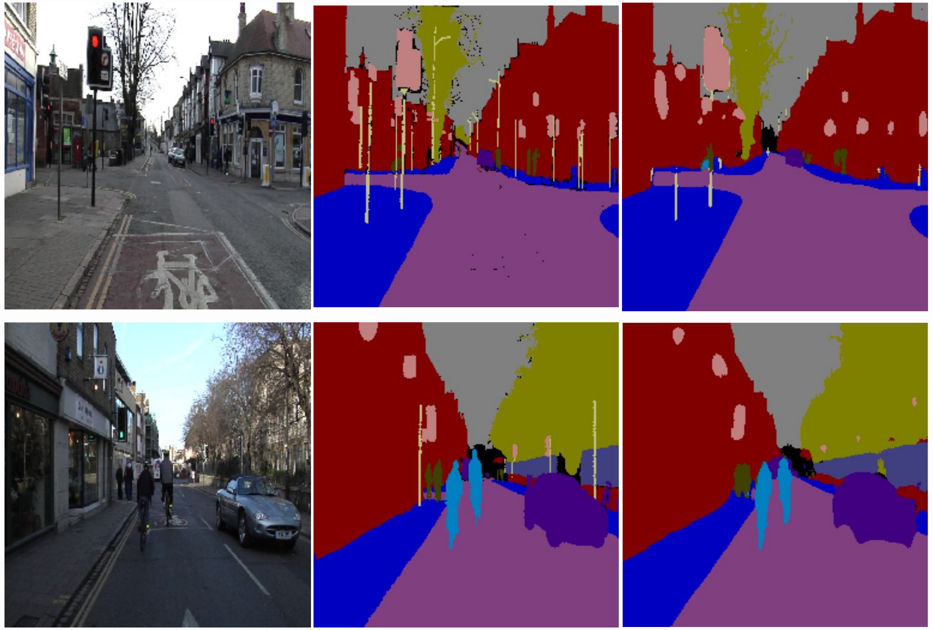
Model 3 (Validation)

Mean IoU (%)	Accuracy(%)
63,17 (**+1.46**)	90,48 (**+0.47**)

Parentheses indicate changes from Model 2.

Parameters = 8630860





Observations:

- More stable training
- Improved Mean IoU
- Training and validation curves flatten towards the end

Solution:

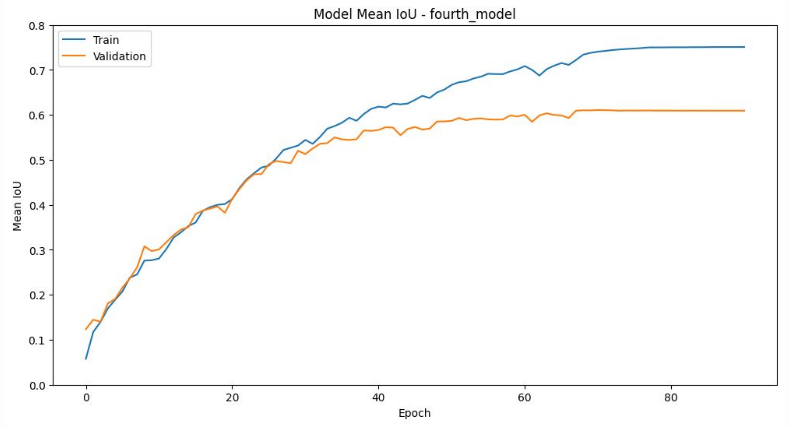
- Add 1 encoder and 1 decoder block

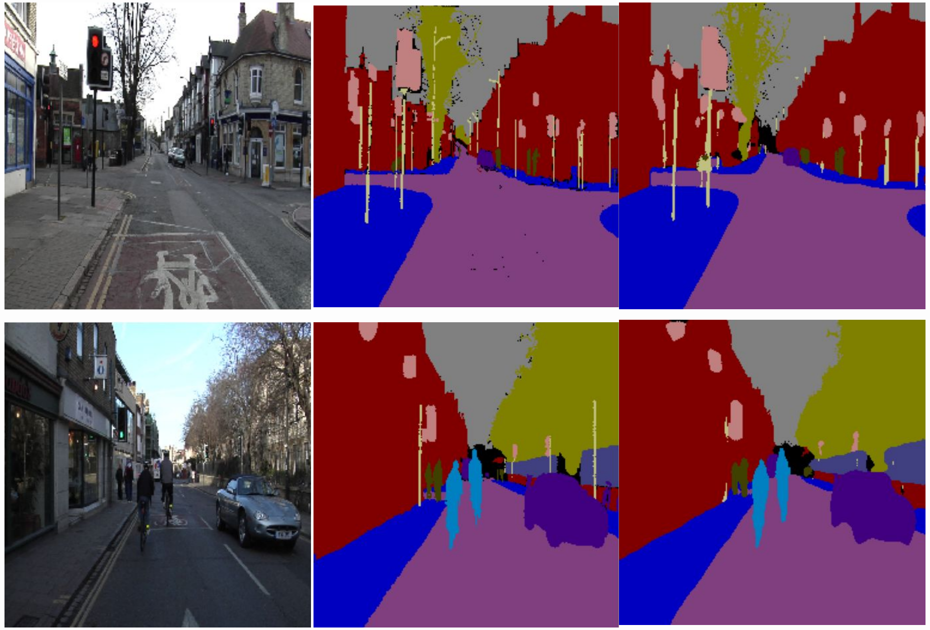
Model 4 (Validation)

Mean IoU (%)	**Accuracy(%)**
65,56 (**+2.39**)	90,77 (**+0.29**)

Parentheses indicate changes from Model 3.

Parameters = 34586700





Observations:

- Slight improvement in Mean IoU
- The training curve flattens towards the end
- Increased gap between training and validation Mean IoU
- Improved detection of poles

Solution:

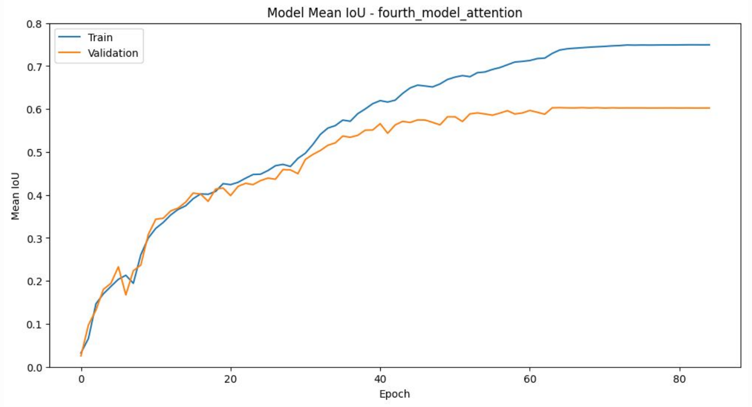
- Add attention modules on decoder

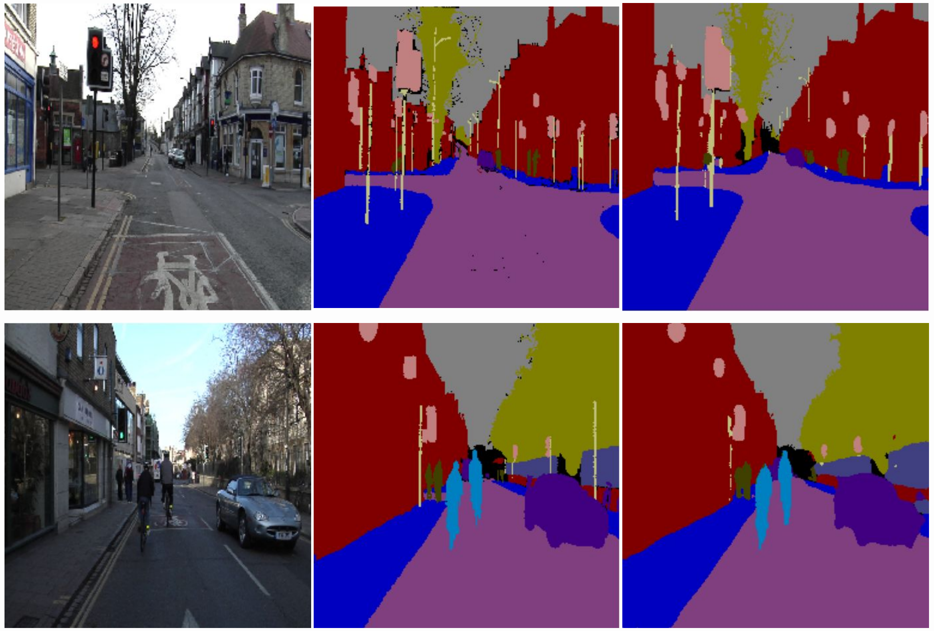
Model 4.2 (Validation)

Mean IoU (%)	Accuracy(%)
64,77 (-0.79)	90,93(+0.16)

Parentheses indicate changes from Model 4.

Parameters = 35288049





Observations:

- Attention modules did not improve performance;
- gap between training and validation didn't improve.
- The training curve flattens towards the end.

Solution:

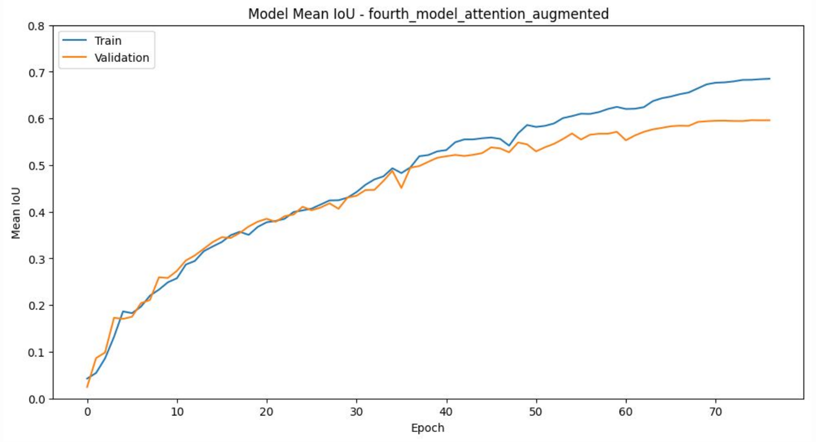
- Augmentation (horizontal flip)

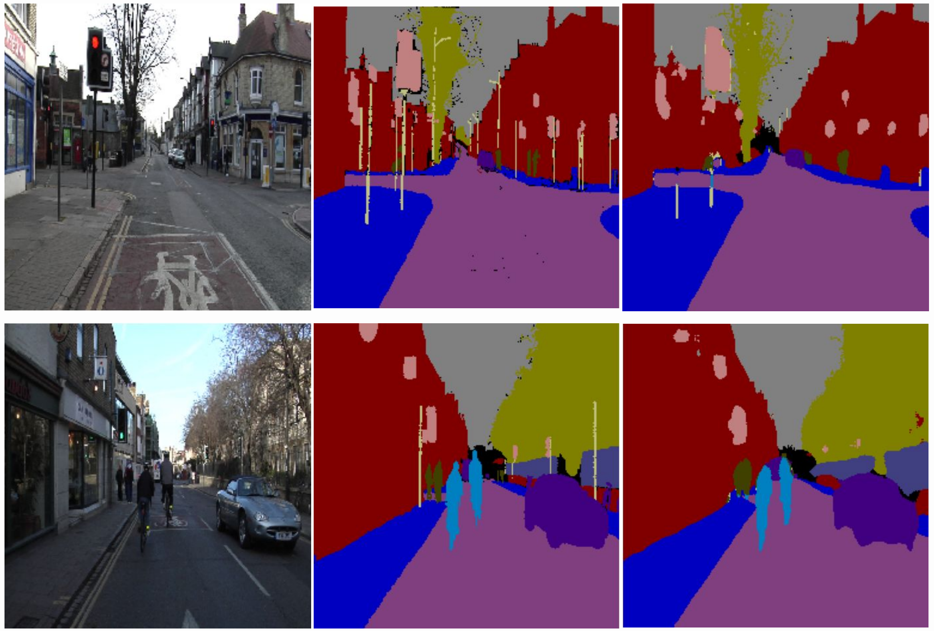
Model 4.3 (Validation)

Mean IoU (%)	Accuracy(%)
63,96 (-0.81)	90,82 (-0.11)

Parentheses indicate changes from Model 4.2

Parameters = 35288049





Observations:

- Worse Performance
- Reduced gap between training and validation
- Does not recognize poles in images

Solution:

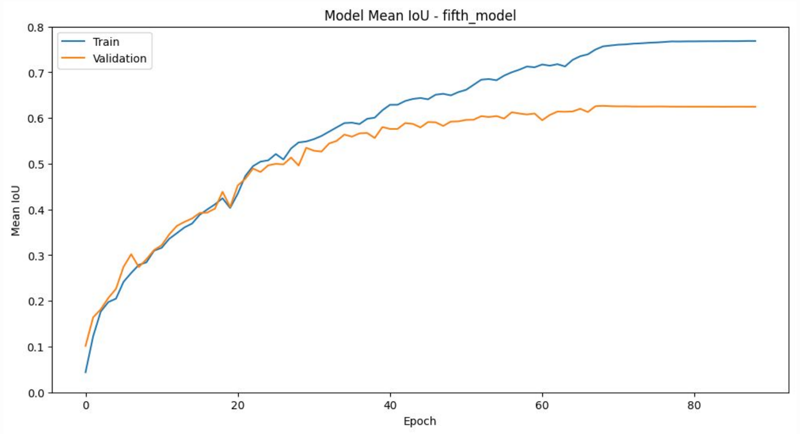
- From model 4 increase number of filters (from 32 initial to 64)

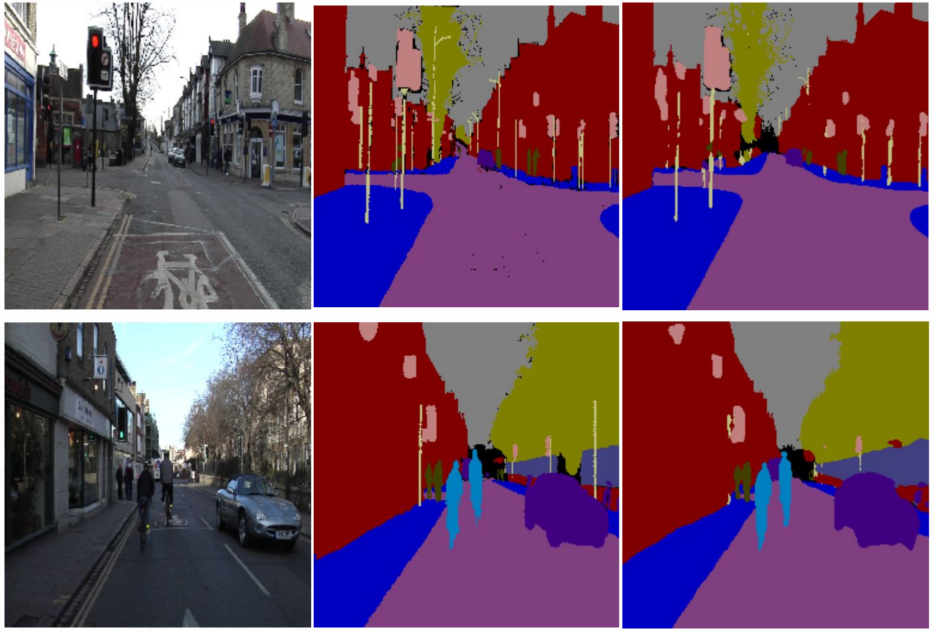
Model 5 (Validation)

Mean IoU (%)	**Accuracy(%)**
67,24 (**+1.68**)	91,35 (**+0.58**)

Parentheses indicate changes from Model 4

Parameters = 138330252





Observations:

- Slight improvement in mean IoU.
- Curves flatten towards the end.
- Doubt if the increased parameters are worth it given the marginal performance gains.

IoU Evaluation (11 classes) (Validation)

	Sky	building	pole	road	side walk	tree	sign	fence	car	pedestrian	bicyclist
Model 2	92.4	81.3	8.6	95.8	80.6	78.0	36.3	47.8	74.8	29.5	53.8
Model 3	92.7	82.9	8.6	85.6	80.5	78.5	36.6	54.1	76.6	32.6	56.3
Model 4	92.5	83.2	16.5	95.8	81.1	78.2	36.7	58.5	78.7	36.5	63.3
Model 4.2	93.0	83.8	15.8	85.8	80.6	79.5	39.7	55.5	77.7	32.6	58.5
Model 4.3	92.8	83.6	9.3	95.7	80.7	78.9	37.5	59.4	78.1	32.9	54.7
Model 5	93.2	84.2	21.4	96.2	82.4	79.6	42.8	57.5	79.2	40.6	62.6

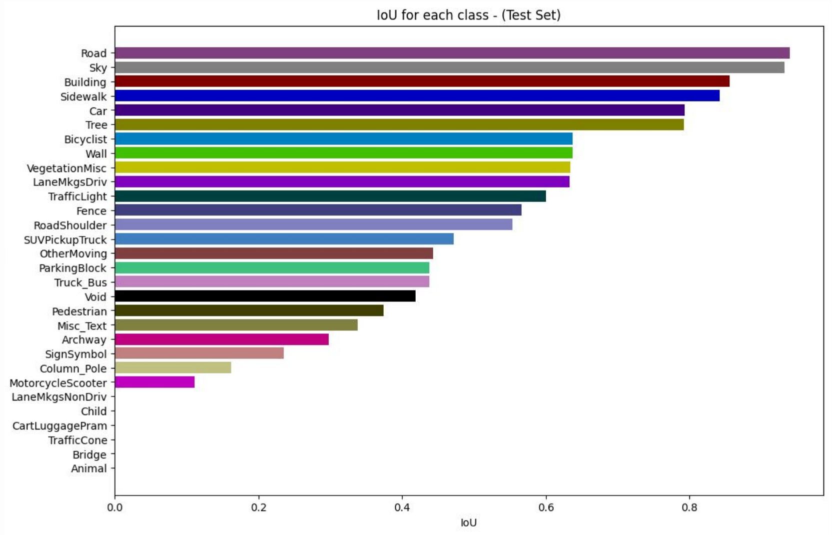
Summary Evaluation (11 Classes) (Validation)

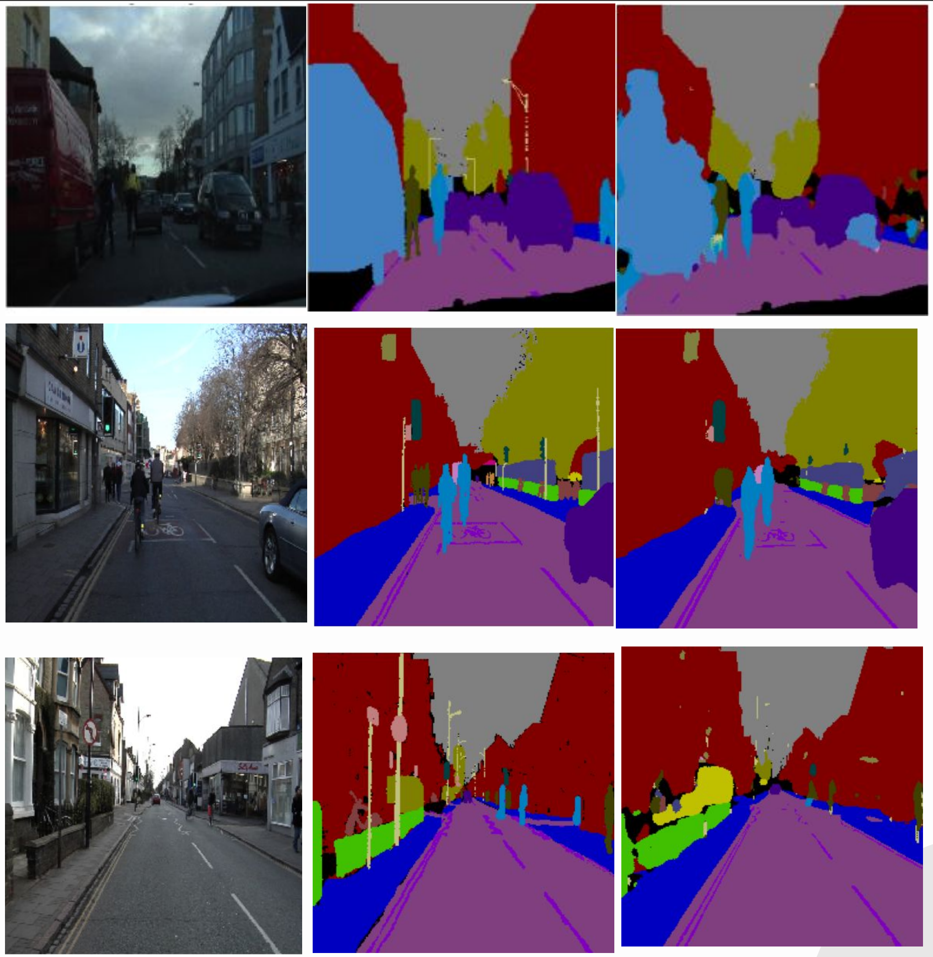
Model	Main Improvement	Description	Mean IoU (%)	Accuracy (%)
Model 2	Baseline model	**Filters**: 32 (encoder), 256(core) **Convolution layers per block**: 2	61,71	90,01
Model 3	Deeper architecture	**Filters:** 32 (initial), 512 (core) **Convolution layers per block:** 2	63,17	90,48
Model 4	Increased Depth	**Filters**: 32 (initial), 1024 (core) **Convolution layers per block**: 2	65,56	90,77
Model 4.2	Added attention mechanisms	**Filters**: 32 (initial), 1024 (core) **Convolution layers per block**: 2 Attention modules in decoder	64,77	90,93
Model 4.3	Data augmentation and attention mechanisms	**Filters:** 32 (initial), 1024 (core) **Convolution layers per block:** 2 Attention modules in decoder + horizontal flip augementation	63,96	90,82
Model 5	Increased number of filters (from model 3)	**Filters**: 64 (initial), 2048 (core) **Convolution layers per block**: 2	67,24	91,35

The architecture includes convolutional, max pooling, and Conv2DTranspose layers. Filters double at each block up to the core. Skip connections are added from Model 2 onwards.

Test Results for 32 Classes - Model 5

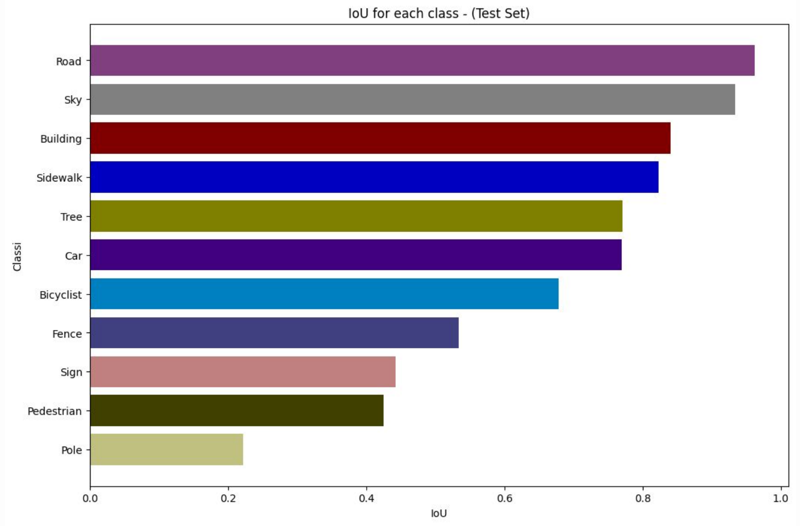
Mean IoU (%)	Accuracy(%)
43,82	89,82

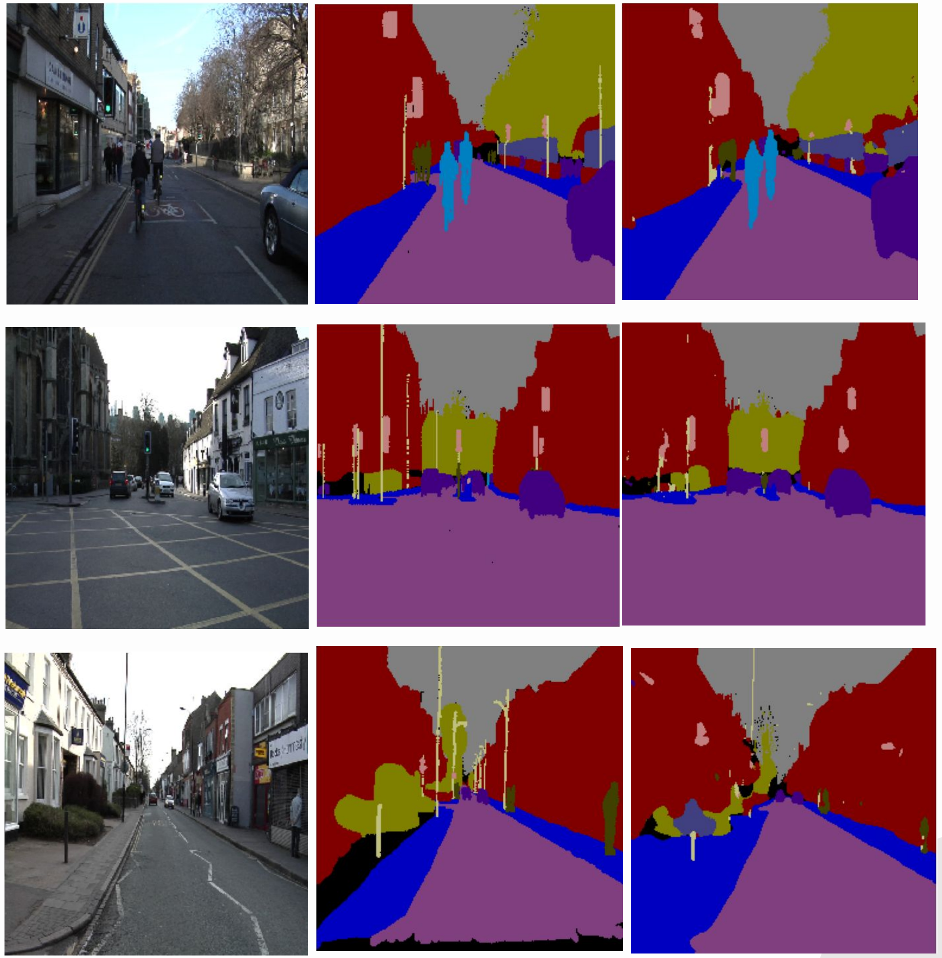




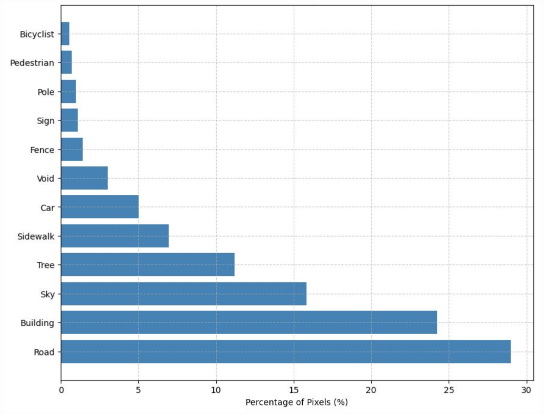
Test Results for 11 Classes - Model 5

Mean IoU (%)	Accuracy(%)
67,31	90,98

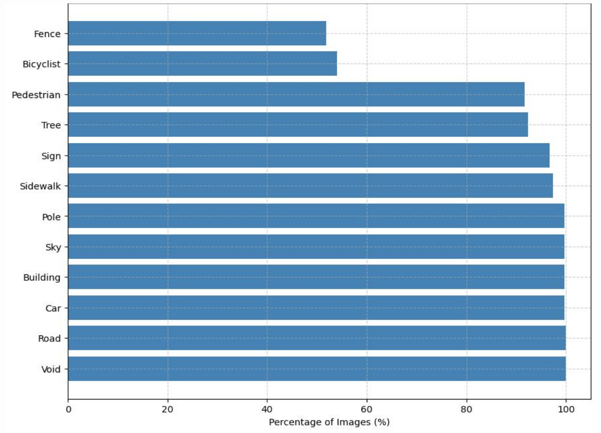




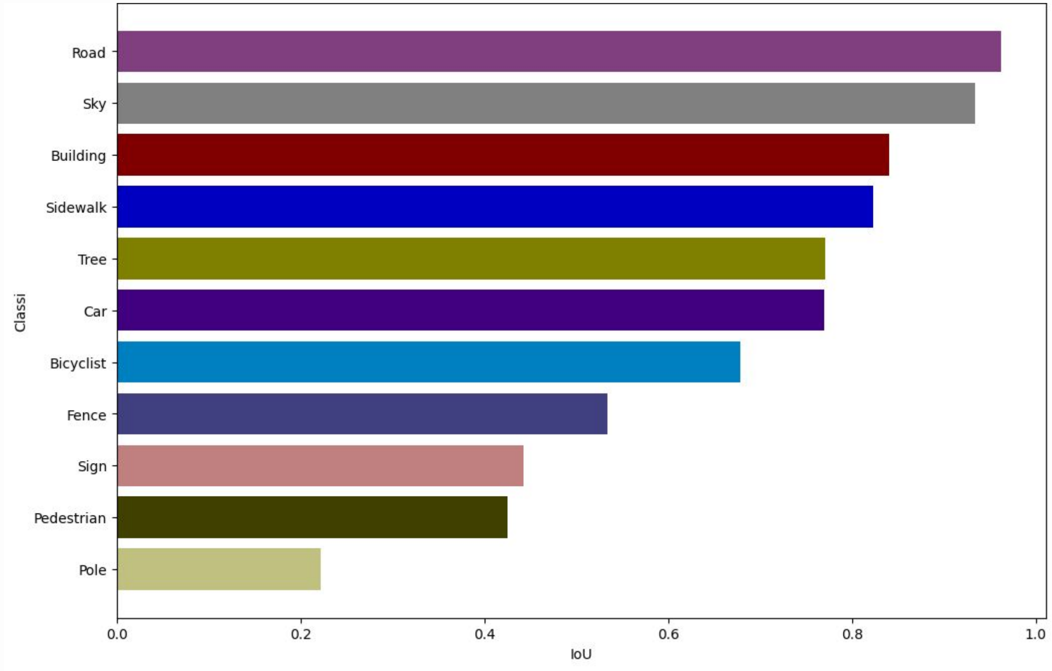
Pixel Class Distribution (11 classes)



Class Appearance Distribution (11 classes)



IoU for each class (11 classes) (Test)



Future Developments

Pretrained Models

Utilizing pretrained models could significantly enhance the model's performance.

Extended Training Periods

Increasing the number of training epochs can potentially improve model accuracy and IoU.

Higher Image Resolution

Using higher quality images (e.g., original resolution instead of 256x256) may lead to better segmentation results.

Thanks!

Julius Maliwat
ID: 864520

Link Models 32 classes
Link Models 11 classes