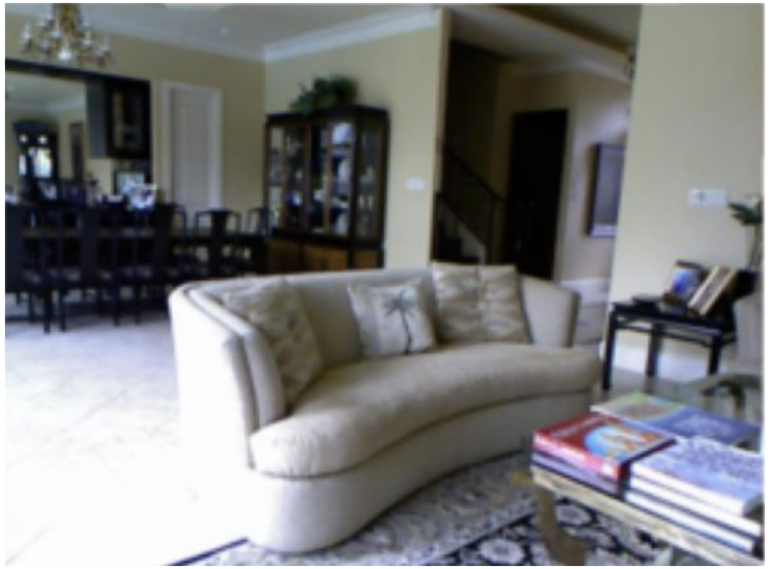
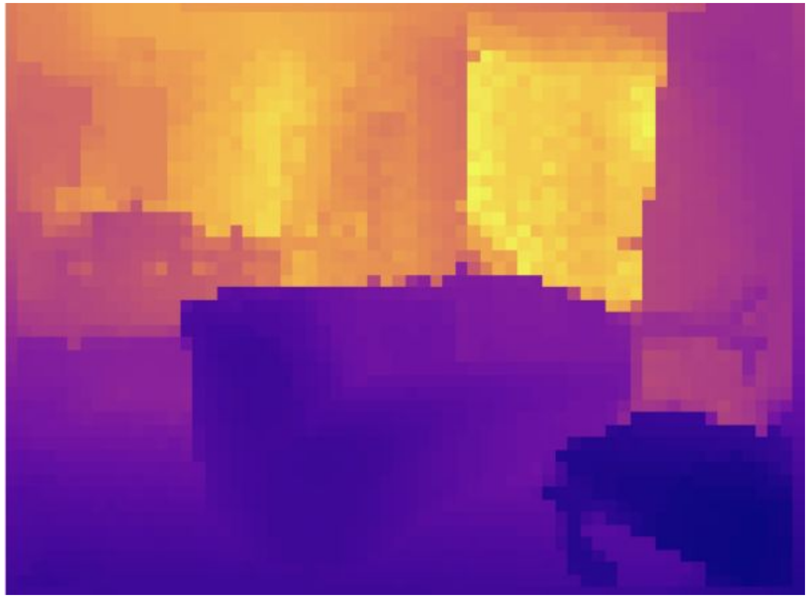
Neuron Selectivity for Efficient Monocular Depth Estimation

Lien Huong Huynh

Evgeniia Rumiantseva

Monocular Depth Estimation





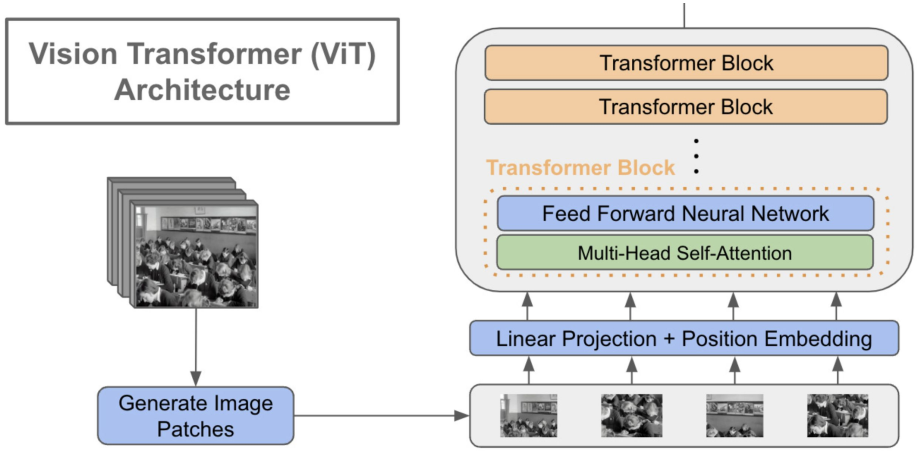
Related Works – CNN-based

- Encoder-Decoder CNNs
- Transfer Learning: Pretrained Encoder + Specific Decoder
- For global extraction capabilities require large computational resources

M. Song, S. Lim, and W. Kim, "Monocular depth estimation using laplacian pyramid-based depth residuals," IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 11, pp. 4381–4393, 2021.

I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," 2018. [Online]. Available: https://arxiv.org/abs/ 1812.11941

Related Works - Visual Transformer

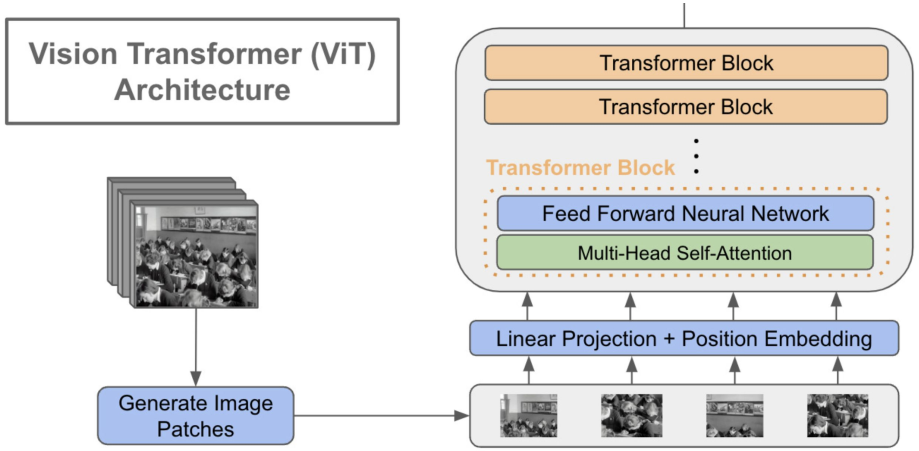


Advantages of ViT: input-adaptive weighting and global processing, which leads to finer-grade predictions with respect to standard CNN

Disadvantages of ViT: too many parameters to run on small devices

Solution: instead of extracting patches straight from the image preprocess & postprocess it with convolutions -> MobileViT -> add skip connections to the decoder -> METER encoder

Related Works - Visual Transformer



<u>Original paper:</u> A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," 2020.

<u>MobileVit:</u> S. Mehta and M. Rastegari, "Mobilevit: Light-weight, general-purpose, and mobile-friendly vision transformer," 2021.

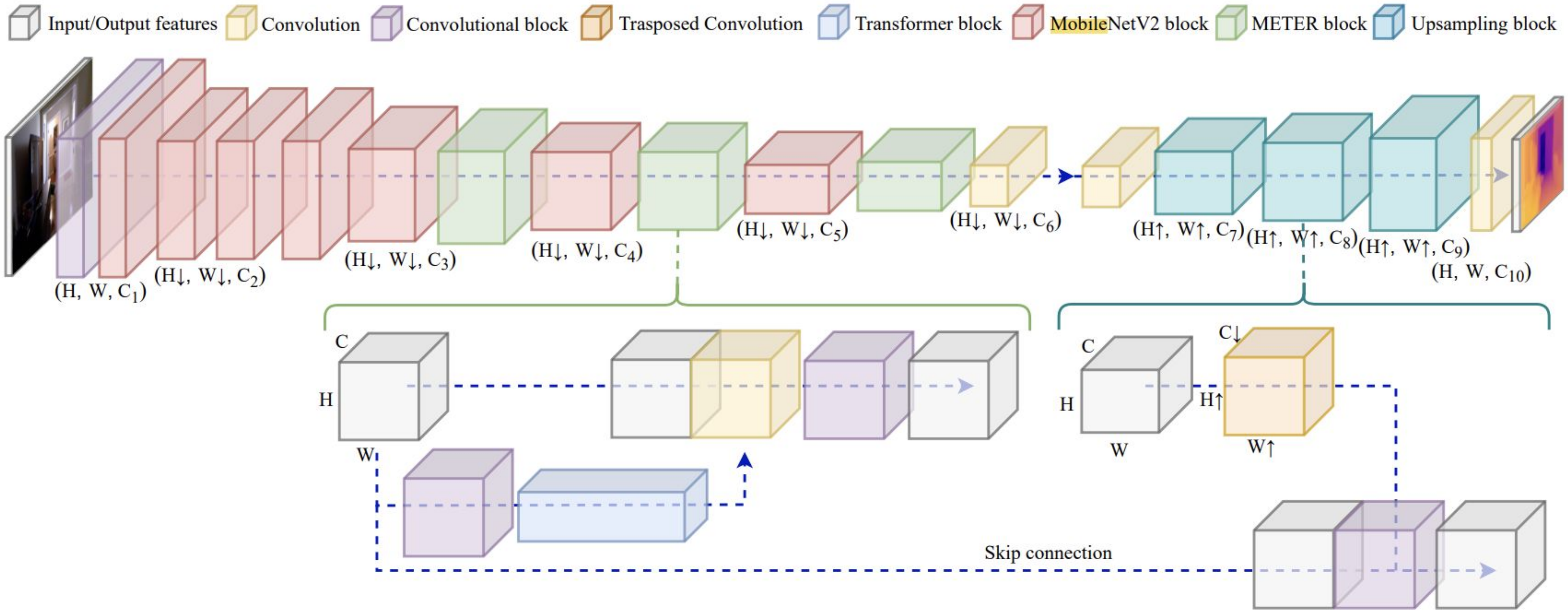
<u>METER:</u> Lorenzo Papa, Paolo Russo and Irene Amerini, "METER: a mobile vision transformer architecture for monocular depth estimation," 2021

Related Works – Interpretability of CV DNNs

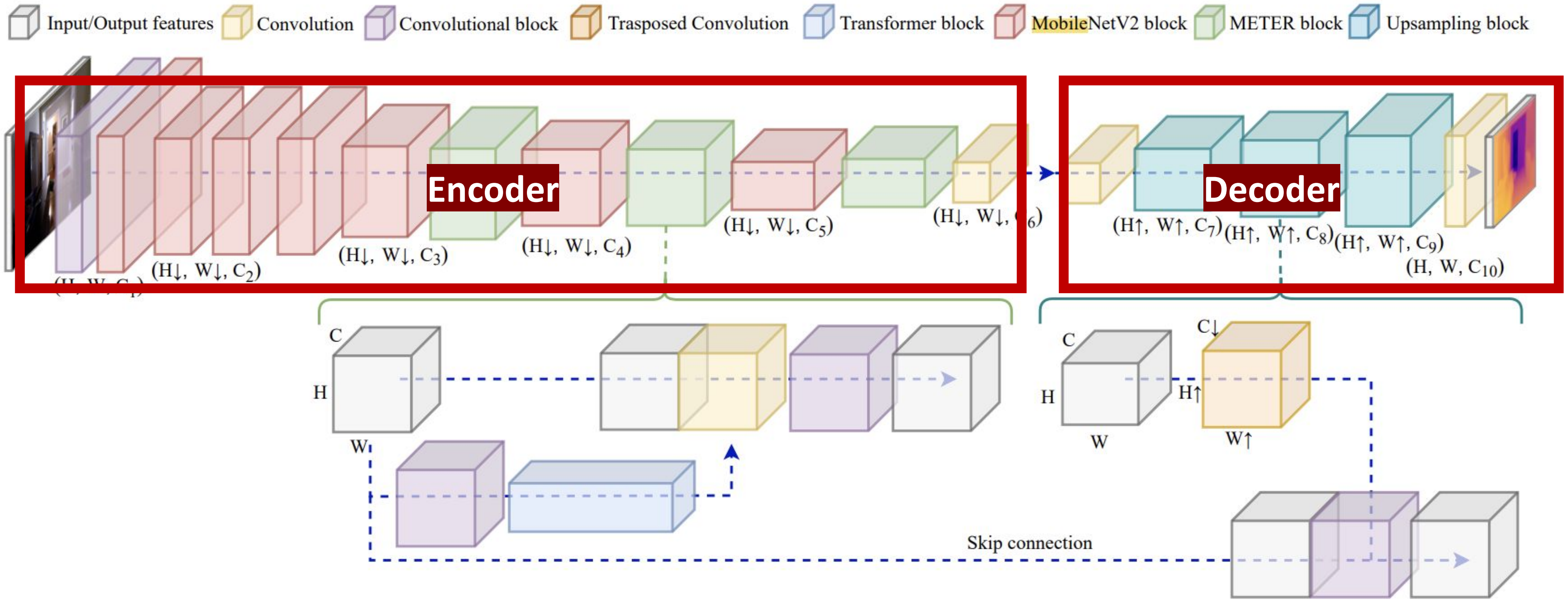
<u>CNNs interpretability (idea is close!):</u> Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018.

<u>Main paper:</u> Zunzhi You, Yi-Hsuan Tsai, Wei-Chen Chiu, Guanbin Li, “Towards Interpretable Deep Networks for Monocular Depth Estimation,” 2021

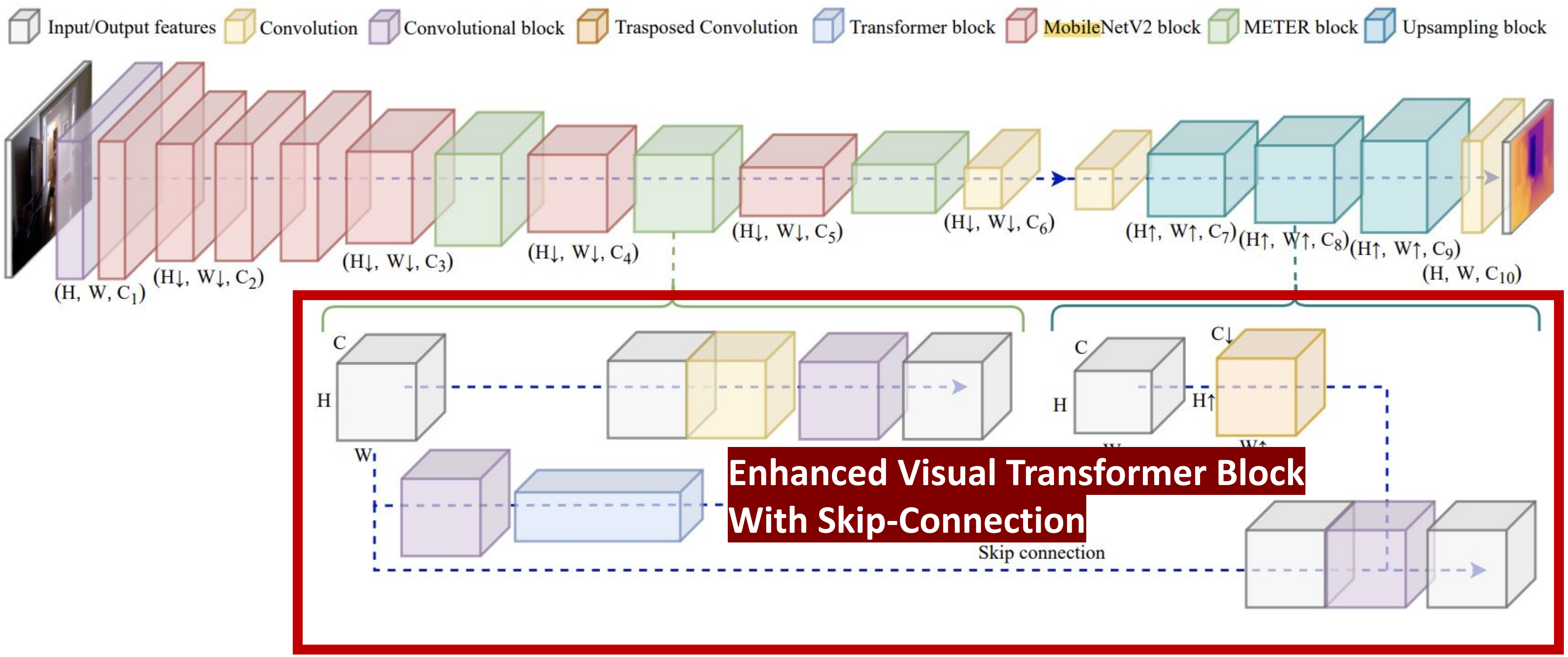
METER Architecture



METER Architecture



METER Architecture



METER

- Loss function: Balanced Loss Function

$$L(y_i, \hat{y}_i) = L_{depth} + \lambda_1 L_{grad} + \lambda_2 L_{norm} + \lambda_3 L_{SSIM}$$

$$L_{SSIM}(y_i, \hat{y}_i) = 1 - SSIM(y_i, \hat{y}_i)$$

Structural similarity

$$L_{depth}(y_i, \hat{y}_i) = \frac{1}{n} \sum_{i=1}^{n} |y_i - \hat{y}_i|$$

$$L_{norm}(y_i, \hat{y}_i) = \frac{1}{n} \sum_{i=1}^{n} \left(1 - \frac{\langle n_{\hat{y}_i}, n_{y_i} \rangle}{\sqrt{\langle n_{\hat{y}_i}, n_{\hat{y}_i} \rangle} \sqrt{\langle n_{y_i}, n_{y_i} \rangle}} \right)$$

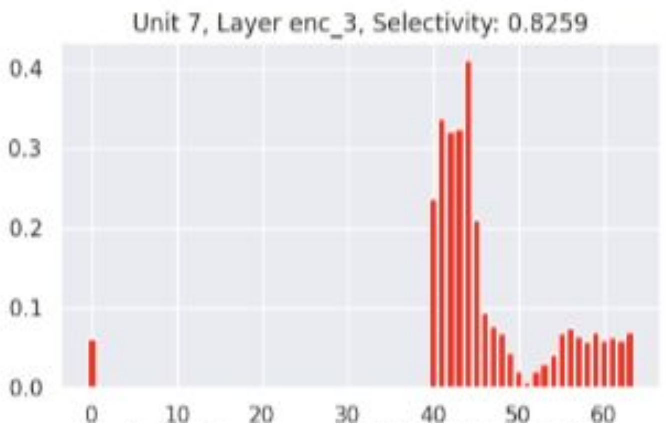
Cosine similarity between depths normals

$$L_{grad}(y_i, \hat{y}_i) = \frac{1}{n} \sum_{i=1}^{n} (\nabla_x(|y_i - \hat{y}_i|) + \nabla_y(|y_i - \hat{y}_i|))$$

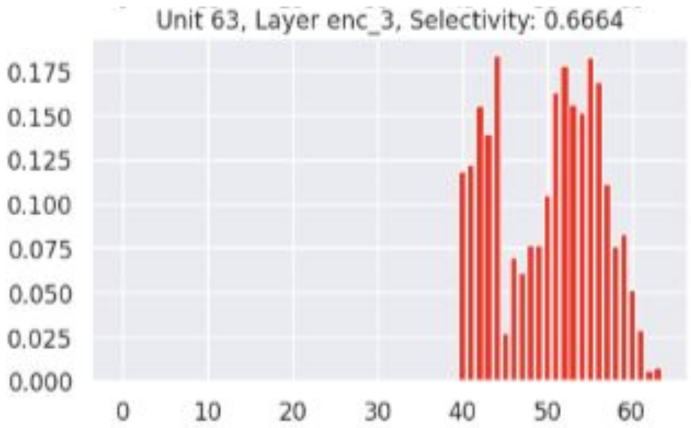
Vertical and horizontal gradient to detect object boundaries

Neuron Selectivity for Interpretability

- Observation: in deep MDE networks, some hidden neurons are selective to specific ranges of depth
- Observation II: ablating neurons with higher selectivity drops quality faster
- Idea: let’s make all the neurons even more depth selective!



Higher selectivity



Lower selectivity

Idea of Depth Selectivity Calculation

1. Computing average response of every separate neuron k in layer l for specific depth range d over the whole dataset: $R_{l,k}^{d}$
2. Compute selectivity index:

$$DS_{l,k} = \frac{|R_{l,k}^{max}| - |\bar{R}_{l,k}^{-max}|}{|R_{l,k}^{max}| + |\bar{R}_{l,k}^{-max}|}$$

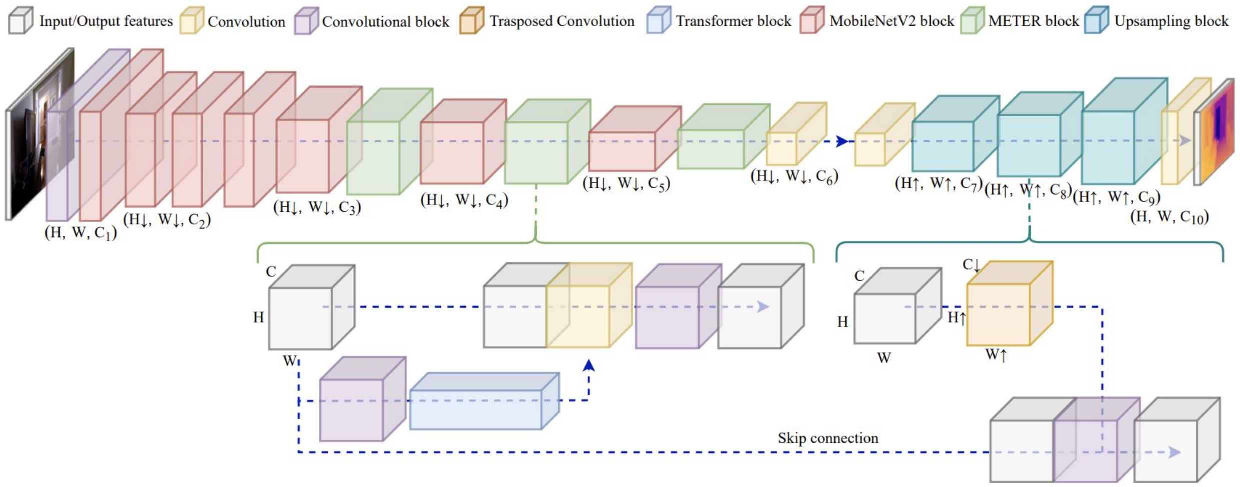
3. Assign each unit a specific depth range & add a corresponding regularizer

Neuron Selectivity Regularization

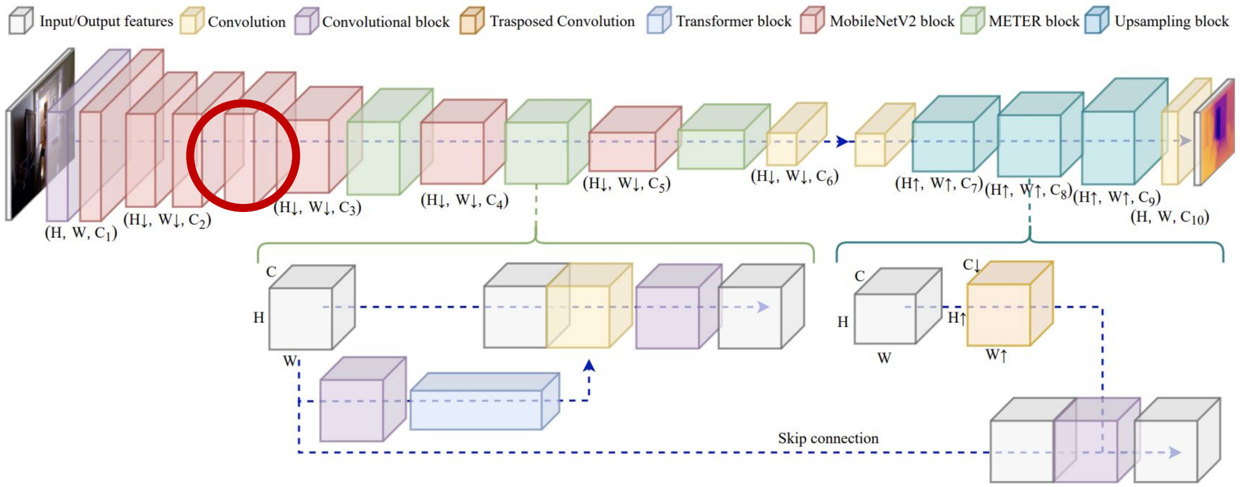
new loss = balanced loss + ɑ·selectivity

$$\mathcal{L}_{assign} = -\lambda \sum_{l \in L} \frac{1}{K_l} \sum_{k} \frac{|R_{l,k}^{d_k}| - |\bar{R}_{l,k}^{-d_k}|}{|R_{l,k}^{d_k}| + |\bar{R}_{l,k}^{-d_k}|}$$

Meter + Neuron Selectivity Regularizer

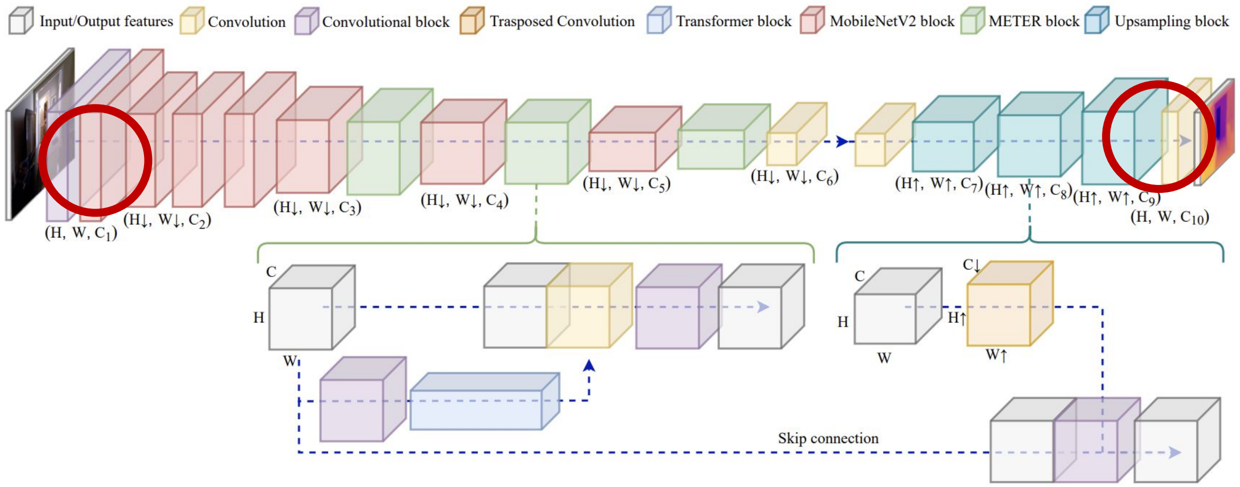


Meter + Neuron Selectivity Regularizer



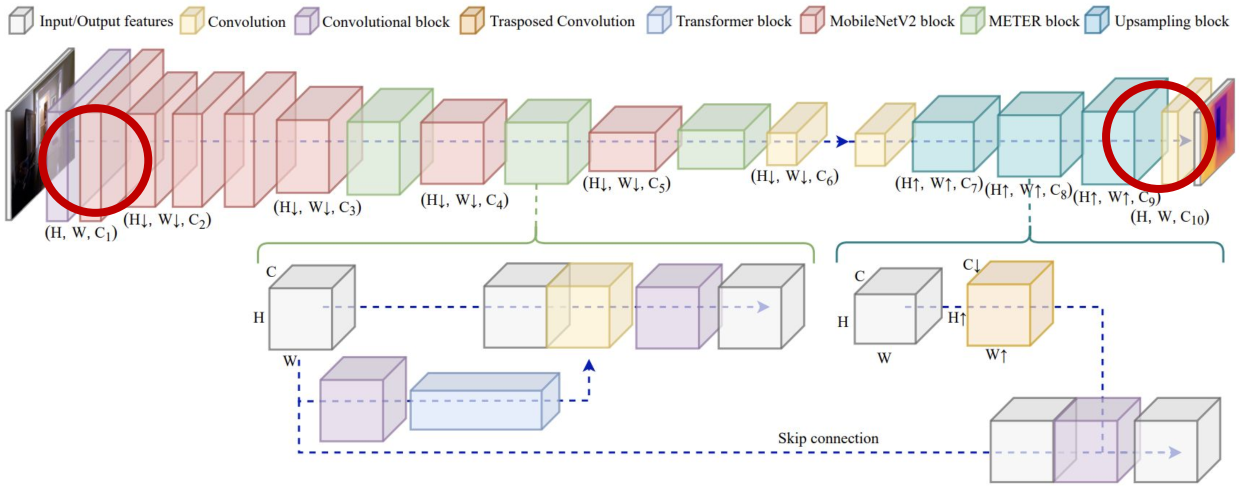
- Setup 1: Selectivity Loss applied to the 3rd encoder skip-connection

Meter + Neuron Selectivity Regularizer



- Setup 1: Selectivity Loss applied to the 3rd encoder skip-connection
- Setup 2: Selectivity Loss applied to 2nd encoder skip-connection + Decoder output

Meter + Neuron Selectivity Regularizer



- Setup 1: Selectivity Loss applied to the 3rd encoder skip-connection
- Setup 2: Selectivity Loss applied to 2nd encoder skip-connection + Decoder output
- Setup 3: Selectivity Loss applied to 2nd encoder skip-connection + Decoder output + adjusted alpha hyperparameter

Data

- NYU Depth v2
  - RGB images and corresponding depth maps in several indoor scenarios
  - Initial resolution is 640×480 pixels
  - For training we use downsampled images to the resolution of 256 x 192

  - Dataset size
    - Train: 40550, Val: 5068, Test: 5070

Evaluation Metrics

- RMSE – for depth estimation quality

$$RMSE = \sqrt{\frac{1}{|n|}\sum_{i\in n}||y_i - \hat{y}_i||^2}$$

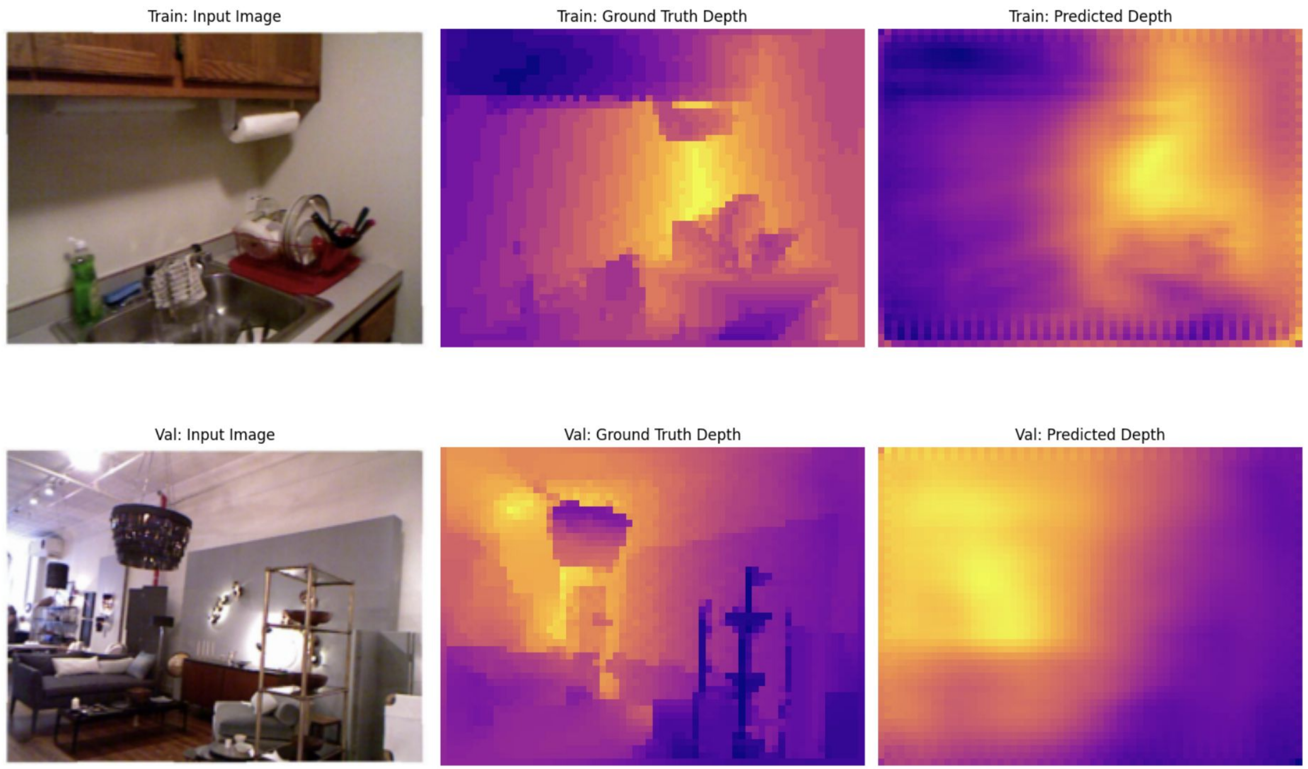
- Average Selectivity for Each Layer

$$\frac{1}{K_l}\sum_{k}\frac{|R_{l,k}^{d_k}| - |\bar{R}_{l,k}^{-d_k}|}{|R_{l,k}^{d_k}| + |\bar{R}_{l,k}^{-d_k}|}$$

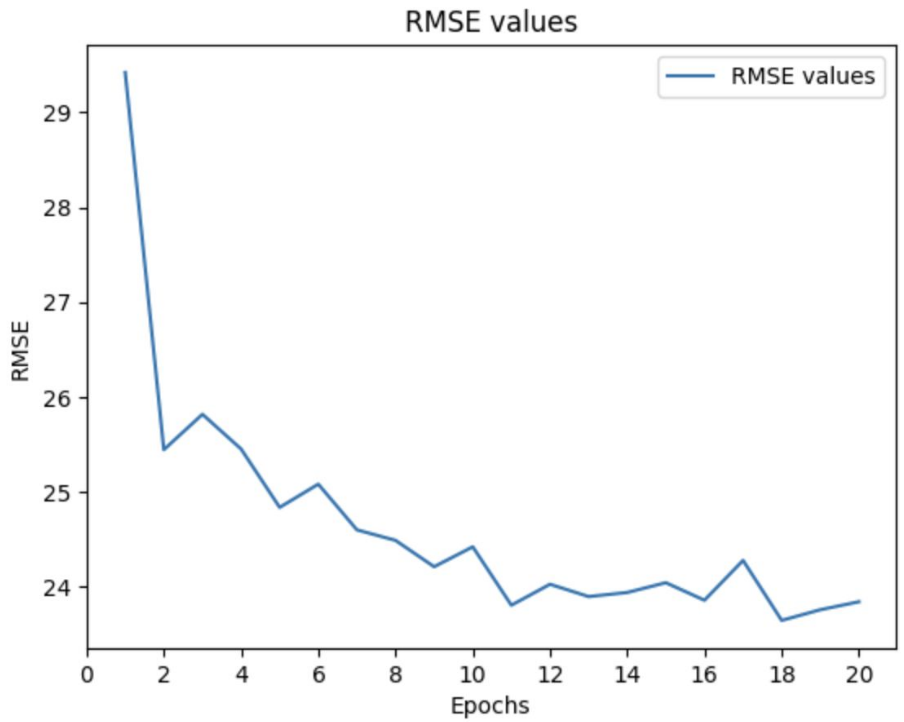
Experimental Setup

- AdamW optimizer: lr = 1e-4, weight_decay = 1e-2
- Number of Epochs: 20
- Batch Size: 64
- Weight for selectivity regularizer:
  - Default: 0.1 (as in the paper)
  - Adjusted: 0.5

Baseline Evaluation

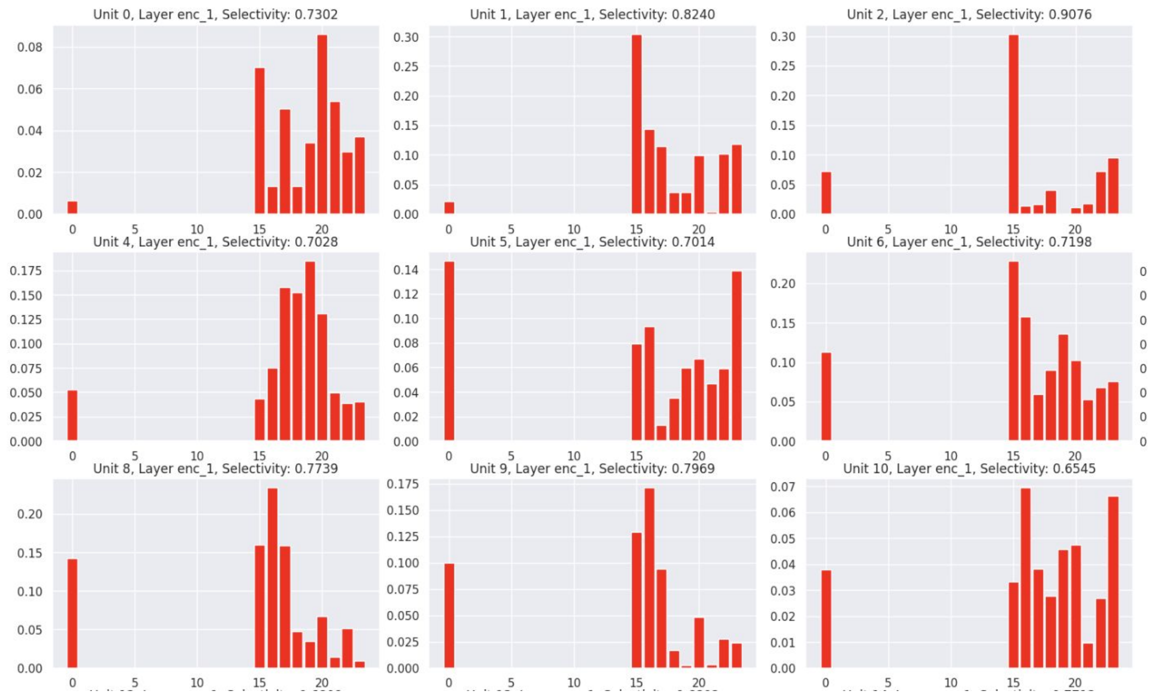


Example predictions on validation dataset after 20 epochs



Best RMSE: 23.64

Baseline Evaluation



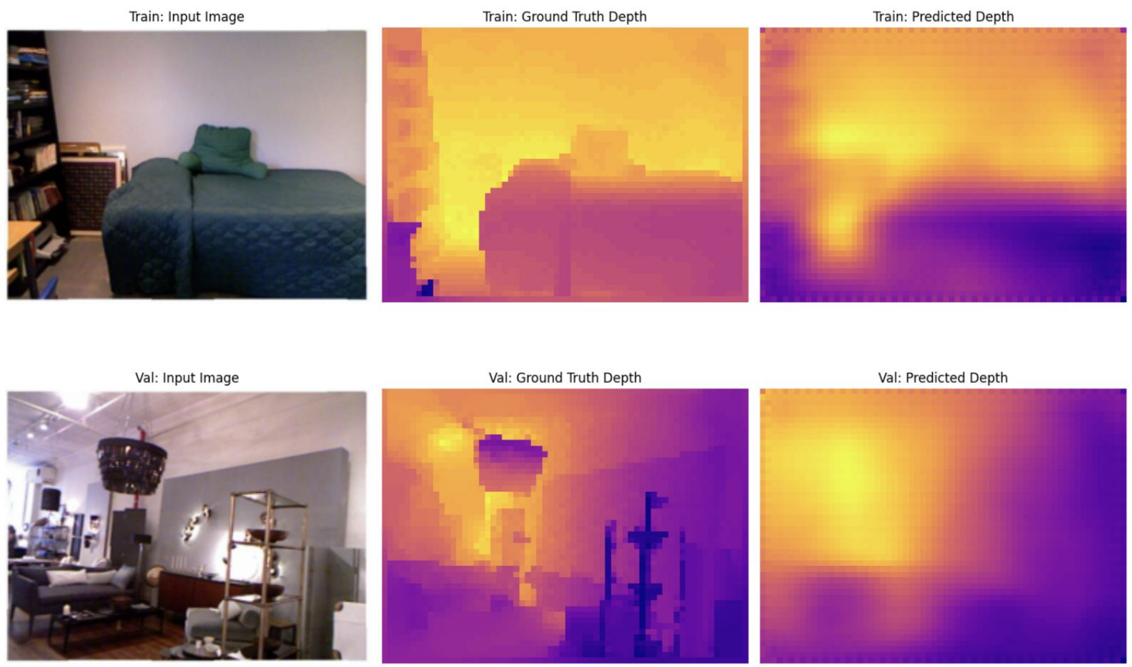
Example neuron depth selectivity distribution for 1st MobileNetV2 block

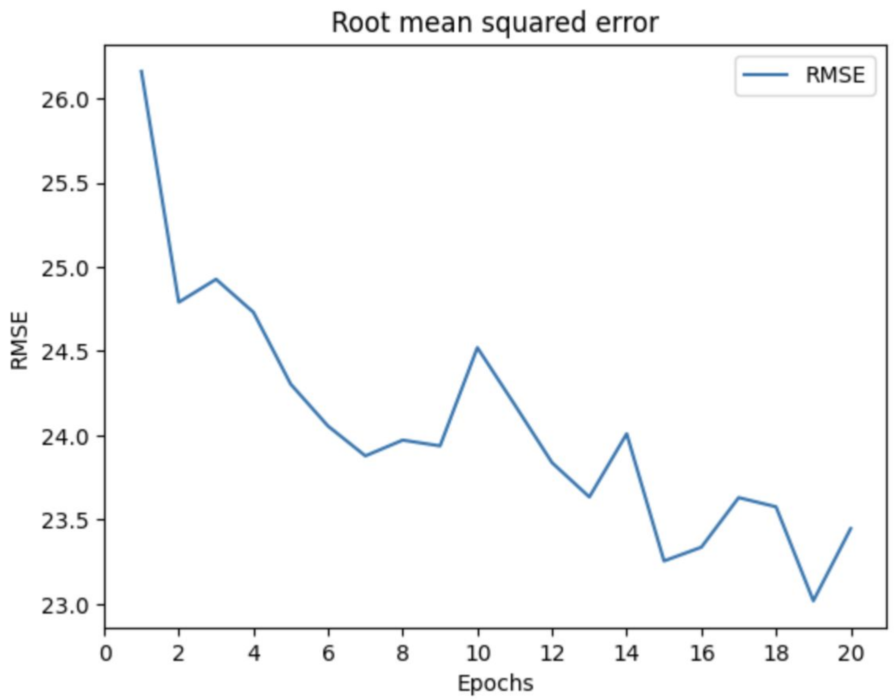
Mean selectivity value for each layer

enc_0 0.533
enc_1 0.555
enc_2 0.746
enc_3 0.734
enc_4 0.735
enc_5 0.745
enc_6 0.579
enc_7 0.784
enc_8 0.590
enc_9 0.758
enc_10 0.672
enc_output 0.667

dec_0 0.635
dec_1 0.606
dec_2 0.583
dec_3 0.436

Evaluation with Neuron Selectivity Loss





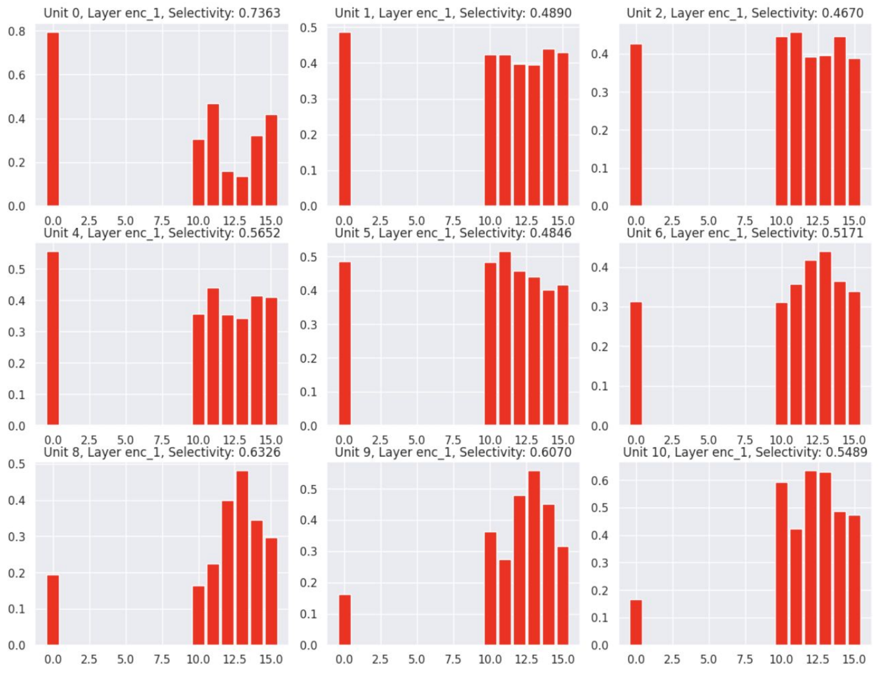
Example predictions on validation dataset after 20 epochs

Best RMSE: 23.02

Evaluation with Neuron Selectivity Loss

Neuron depth selectivity distribution for 1st MobileNetV2 block.

Selectivity enhanced for 3rd encoder MobileNetV2 block and 3rd decoder block.



Mean selectivity value for each layer

enc_0 0.543
enc_1 0.566
enc_2 0.762
enc_3 0.694
enc_4 0.650
enc_5 0.760
enc_6 0.574
enc_7 0.691
enc_8 0.602
enc_9 0.720
enc_10 0.659
enc_output 0.657

dec_0 0.632
dec_1 0.602
dec_2 0.544
dec_3 0.534

Evaluation with Neuron Selectivity Loss

Baseline

enc_0 0.533
enc_1 0.555
enc_2 0.746
enc_3 0.734
enc_4 0.735
enc_5 0.745
enc_6 0.579
enc_7 0.784
enc_8 0.590
enc_9 0.758
enc_10 0.672
enc_output 0.667

dec_0 0.635
dec_1 0.606
dec_2 0.583
dec_3 0.436

AVG selectivity before: 0.647

With Neuron Selectivity Loss

enc_0 0.543
enc_1 0.566
enc_2 0.762
enc_3 0.694
enc_4 0.650
enc_5 0.760
enc_6 0.574
enc_7 0.691
enc_8 0.602
enc_9 0.720
enc_10 0.659
enc_output 0.657

dec_0 0.632
dec_1 0.602
dec_2 0.544
dec_3 0.534

AVG selectivity after: 0.636

Evaluation with Neuron Selectivity Loss + new Alpha

Baseline

enc_0 0.533
enc_1 0.555
enc_2 0.746
enc_3 0.734
enc_4 0.735
enc_5 0.745
enc_6 0.579
enc_7 0.784
enc_8 0.590
enc_9 0.758
enc_10 0.672
enc_output 0.667

dec_0 0.635
dec_1 0.606
dec_2 0.583
dec_3 0.436

AVG selectivity before: 0.647

With Neuron Selectivity Loss

enc_0 0.503
enc_1 0.522
enc_2 0.742
enc_3 0.706
enc_4 0.718
enc_5 0.737
enc_6 0.570
enc_7 0.753
enc_8 0.577
enc_9 0.756
enc_10 0.628
enc_output 0.622

dec_0 0.589
dec_1 0.569
dec_2 0.529
dec_3 0.593

AVG selectivity before: 0.632

Conclusions and Future Work

- Selectivity Regularisation boosts selectivity for some layers, but the overall selectivity does not improve

- We can see slight improvement of RMSE after training with regularisation component

- Improvement of interpretability requires further thorough hyperparameters tuning (alpha / selected layers)

Conclusions and Future Work

Other setups to be explored:

- Setup 3: apply loss to all skip-connections inputs
- Setup 4: apply loss to all skip-connections inputs + encoder output
- Setup 5: apply loss only to encoder output
- Adjust weight of the Selectivity Loss Component