InverseForm: A Loss Function for Structured Boundary-Aware Segmentation

Shubhankar Borse   Ying Wang   Yizhe Zhang   Fatih Porikli

Qualcomm AI Research*

{sborse, yizhez, fporikli}@qti.qualcomm.com, yingwang0022@gmail.com

Abstract

We present a novel boundary-aware loss term for semantic segmentation using an inverse-transformation network, which efficiently learns the degree of parametric transformations between estimated and target boundaries. This plug-in loss term complements the cross-entropy loss in capturing boundary transformations and allows consistent and significant performance improvement on segmentation backbone models without increasing their size and computational complexity. We analyze the quantitative and qualitative effects of our loss function on three indoor and outdoor segmentation benchmarks, including Cityscapes, NYU-Depth-v2, and PASCAL, integrating it into the training phase of several backbone networks in both single-task and multi-task settings. Our extensive experiments show that the proposed method consistently outperforms baselines, and even sets the new state-of-the-art on two datasets.

1. Introduction

Semantic segmentation is a fundamental computer vision task with numerous real-world applications such as autonomous driving, medical image analysis, 3D reconstruction, AR/VR and visual surveillance. It aims to perform pixel-level labeling with a given set of target categories.

There have been notable advancements in semantic segmentation thanks to recent solutions based on deep learning models, such as the end-to-end fully convolutional networks (FCN) [30] that lead to significant performance gains in popular benchmarks. Extensive studies have been conducted to improve semantic segmentation. One prevailing direction is to integrate multi-resolution and hierarchical feature maps [40][38]. Another ambitious objective is to exploit boundary information to enhance segmentation [50][39] further, driven by the observations that segmentation prediction errors are more likely to occur near the boundaries [39] [22]. In parallel, multi-task learning (MTL)
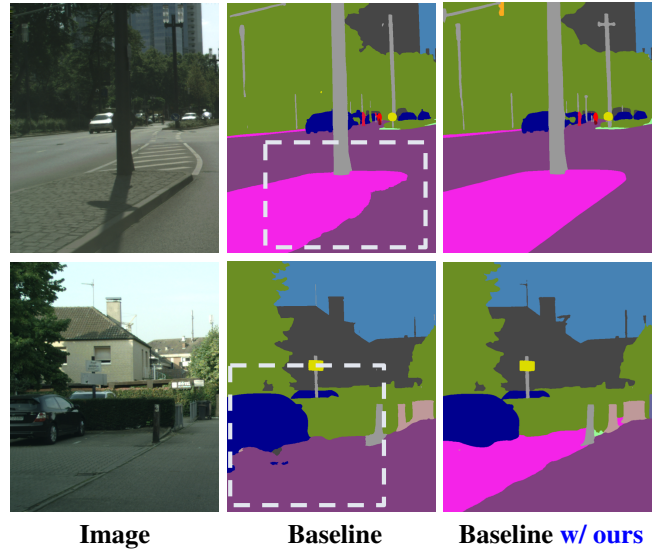
*Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

Figure 1: Left: Images from Cityscapes *val* benchmark [9]. Middle: Segmented prediction for an HRNet-48-OCR [45] [49] baseline. Right: Same backbone trained using our InverseForm boundary loss. Our model achieves clear improvements, e.g. the curbside boundary in the top figure is aligned better with the structure of boundary, and the curbside in the bottom is correctly detected.

frameworks [14][31] explored joint optimization of semantic segmentation and supplemental tasks such as boundary detection [14] [44][31] or depth estimation [48].

Our approach is aligned with making the best use of boundary exploration. One of the main differences is that most of the previous works use a weighted cross-entropy loss as their loss function for boundary detection, which we show in Figure 2, is sub-optimal for measuring the boundary shifts. This is also partially observed in [23]. The cross-entropy loss mainly builds on estimated and ground-truth pixel label changes but ignores the spatial distance of the pixels from the target boundaries. It cannot effectively measure localized spatial variations such as translation, rotation, or scaling between predicted and target boundaries.

Motivated to address this limitation, we introduce a boundary distance-based measure called InverseForm, into the popular segmentation loss functions. We design an inverse transformation network to model the distance between boundary maps, which can efficiently learn the degree of parametric transformations between local spatial regions. This measure allows us to achieve a significant and consistent improvement in semantic segmentation using any backbone model without increasing the inference size and computational complexity of the network.

Specifically, we propose a boundary-aware segmentation scheme during the training phase, by integrating our spatial-distance loss, InverseForm, into the existing pixel-based loss. Our distance-based loss complements the pixel-based loss in capturing boundary transformations. We utilize our inverse transformation network for distance measurement from boundaries and jointly optimize pixel-label accuracy and boundary distance. We can integrate our proposed scheme into any segmentation model; for instance, we adopt the latest HRNet [45] architecture as one of the backbones since it maintains high resolution feature maps.

We also adopt various MTL frameworks [31] [44] to leverage on their boundary detection task towards further segmentation performance improvement, at no added computational and memory cost. In this variant, we show a consistent performance improvement across all tasks as a plus.

We conduct comprehensive experiments on large benchmark datasets including NYU-Depth-v2 [37], Cityscapes [9] and PASCAL-Context [15]. For NYU-Depth-v2, we show that InverseForm based segmentation outperforms the state-of-art in terms of mean intersection-over-union (mIoU). We also show that we outperform state-of-the-art Multi-task learning methods on PASCAL in terms of their multi-task performance. This includes superior performance in predicted edge quality on odsF-score [32] and improvement in other tasks, such as human-parts estimation and saliency estimation, on mIoU. Then, we perform rigorous experiments on Cityscapes to compare our method with contemporary works such as SegFix [50] and Gated-SCNN [39].

The contributions of our work include the following:

- We propose a boundary distance-based measure, InverseForm, to improve semantic segmentation. We show that our specifically tailored measure is significantly more capable of capturing the spatial boundary transforms than cross-entropy based measures, thus resulting in more accurate segmentation results.

- Our scheme is agnostic to the backbone architecture choice and is very flexible to be plugged into any existing segmentation model, with no additional inference cost. It does not impact the main structure of the network due to its plug-and-play property. It is flexible and can fit into multi-task learning frameworks.

- We show through extensive experiments that our boundary-aware-segmentation method consistently outperforms its baselines, and also outperforms the state-of-the-art methods in both single-task (on NYU-Depth-v2) and multi-task settings (on PASCAL).

2. Related Work

Semantic segmentation: The introduction of fully convolutional networks (FCNs) [30] led to significant progress in semantic segmentation. Many works build on FCNs to improve segmentation performance. The work in [26] proposed an approximate inference algorithm for conditional random fields (CRFs) models to improve segmentation efficiency. Various architectures have been proposed to improve inference speed, such as DeepLab [5], PSPNet [54], and HRNet [45]. Numerous recent works utilize HRNet by integrating diverse contextual models [49] [50] [39].

Boundary-aware segmentation: Boundary-aware segmentation has been studied in quite a few recent efforts. There are several ways of modeling this problem. In [1], a global energy model called boundary neural field (BNF) is introduced to predict boundary cues to enhance semantic segmentation. The authors show that the energy decomposes semantic segmentation into a set of binary problems which can be relaxed and solved by global optimization. In [10] a boundary-aware feature propagation module (BFP) is proposed to propagate local features isolated by the boundaries learned in the UAG-structured image. The work in [29] learns semantically-aware affinity through a spatial propagation network to refine image segmentation boundaries. Similarly, [3] adopts a fast domain transform filtering on the output and generates edge maps to refine segmentation output. Gated-SCNN [39] injects the learned boundary into intermediate layers to improve segmentation. This is done using gating to allow interaction between the main segmentation stream and the shape stream. However, these boundaries are learnt using a pixel-based loss. We show in Section 4 how this loss benefits from using our proposed spatial loss. Other related works include [22] [11] [12] [28] [7] [34], and [23].

One of the main differences between our work and all the previous works is that most of the previous works use weighted cross-entropy as their loss function for boundary detection, which is sub-optimal for measuring the boundary changes. The cross-entropy loss mainly considers the pixel label but ignores the distance between the boundaries. It cannot effectively measure the shifts, scales, rotations, and other spatial changes between two image regions. In our work, we introduce a distance-based metric into this loss function. Specifically, we employ an inverse-transformation network to model the spatial distance between boundary
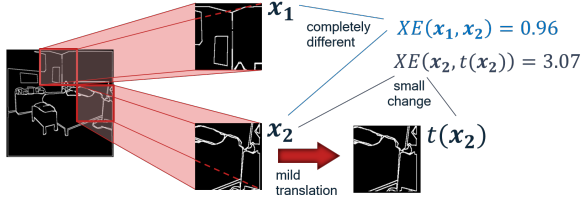
Figure 2: Cross-entropy(XE) based distance fails for spatial transformations of boundaries. In the above example, x_1 and x_2 are slices sampled from a boundary map. Only a mild shift is applied to x_2 to generate a third slice $t(x_2)$.

maps, which is shown to significantly improve the capability in capturing such boundary transforms. An interesting recent work which makes use of spatial relations is Seg-Fix [50]. It learns discrete distance-transformation maps to incorporate spatial relations. Instead of learning these offset maps, our work regresses over homography parameters over which it uses a derived distance computation. We show both qualitatively and quantitatively that this is more effective in Section 4. Another major difference between our work and all previous mentioned works is that our proposed method requires no extra cost during inference.

Multi-task learning: Multi-task learning (MTL) [24] learns shared representations from multi-task supervisory signals under a unified framework. It is effective at exploring interactions among multiple tasks while saving memory and computation. [44] shows the superior performance of multi-task learning for semantic segmentation and depth estimation for NYU-Depth-v2. [43] provides a comprehensive overview of multi-task learning techniques. In [31] the authors address the task interference issue by performing one task at a time using the multi-task framework. They allow the network to adapt its behavior through task attention.

3. Proposed Scheme: InverseForm

Below, we explain our method for boundary-aware segmentation using a spatial distance-based loss function.

3.1. Motivation for distance-based metrics

Boundary-aware segmentation approaches use pixel-wise cross-entropy or balanced cross-entropy losses. Such loss functions take into account the pixel-wise features (intensity, etc.) but not spatial distance between object boundaries and ground-truth boundaries. Hence, they are insufficient for imposing boundary alignment for segmentation.

This drawback is illustrated in Figure 2. We pick an image from NYU-Depth-v2 and convert the ground-truth segmentation into a boundary map. We then sample two regions x_1 and x_2 from the map. We apply a spatial translation by 3 pixels in both dimensions to x_2. The cross-

entropy(XE) between pairs of these images is also shown. Notice that the actual difference between the region x_2 and its transformed version $t(x_2)$ is not high, as the boundary maps are shifted by only a few pixels. However, cross-entropy between this pair of images is much higher than the cross-entropy between x_1 and x_2, which have no correlation. Ideally, if x_2 is the ground truth map; a network should be penalized more when it generates x_1 as compared to when it generates $t(x_2)$. This is clearly not the case. Pixel-based losses are thus, not enough to measure 'distance' between such inputs. To counter the effect of sensitivity to small displacements, boundary detection networks trained with pixel-based losses produce thicker and distorted boundaries. Hence, there is an existing need to measure a distance function between two boundary maps which models possible spatial transformations between them.

Some works use Hausdorff distance [36] to model this measure between boundaries for instance-level segmentation, but this loss cannot be efficiently applied in a semantic segmentation setting. The correlation operation could also be considered between two images, but this would only model translations between image boundaries and cannot model other transformations. Thus, we need a loss function that accurately models the distance between two images of object boundaries and can be computed efficiently.

3.2. Inverse transformation network

To model spatial relations between two boundary maps, we assume that they are related to each other through a homography transformation. Thus, we create a network that inputs two boundary maps and predicts the "homography change" as its output. Theoretically, this network performs the inverse operation of a spatial transformer network (STN) [21]. As shown in Figure 3(a), STN takes an image x as its input, generates a controlled transformation matrix θ, and produces a realistic transformed boundary image $t_\theta(x)$.

Figure 3(b) shows our proposed network. Inputs to our network are two boundary maps. The network regresses on the transformation parameters between these maps as its output. Thus, we call it an *inverse transformation network*. Some recent works [27] [52] attempted to train a similar network to use its encoder for an unsupervised learning task. Our architecture and application are completely different; we use the network to model a spatial distance measure, and our model has only dense layers. We compare our inverse transformation network's performance with [27] in Section 5, where we discuss the translation-invariance of convolutions and why we selected a dense architecture.

The outputs of the inverse-transformation network are the coefficients of the homography matrix. There are numerous methods to formulate a distance metric from these values. Below, we present two distance metric choices.

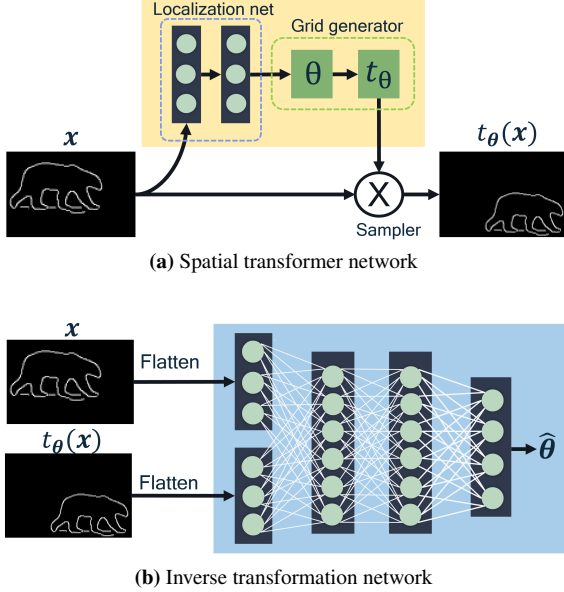We note that one may also attempt to directly regress on

(a) Spatial transformer network

(b) Inverse transformation network

Figure 3: Spatial transformer (a) and our inverse-transformation network (b). Our inverse-transformation network architecture inputs two images $\boldsymbol{x}$ and $t_{\boldsymbol{\theta}}(\boldsymbol{x})$, and predicts the transformation parameters $\hat{\boldsymbol{\theta}}$. This performs the inverse operation as that of the spatial transformer network.

the distance instead of estimating the transformation coefficients. However, such an approach would not allow optimization of the boundary-aware segmentation network.

3.3. Measuring distances from homography

Assuming two boundary maps are related to each other with a homography transformation, our inverse-transformation net inputs two boundary maps and estimates the transformation matrix parameters. Once our inverse-transformation network is trained, we freeze its weights to generate our proposed InverseForm loss. If there is a perfect match between input boundary maps, the network should estimate an identity matrix. Thus, a measure of spatial distance between the boundary maps can be calculated by comparing the network outputs to an identity matrix.

Here are two distance measures we propose to relate the two matrices: Euclidean distance and Geodesic distance.

Euclidean distance: The Euclidean distance between two homography matrices is a measure of spatial distance between the two input boundary maps. We observed that Euclidean distance could model shift and scale relations well. However, it fails to reflect rotations and other perspective transformations. Considering the output of the network as $\hat{\boldsymbol{\theta}}$, the 3×3 identity matrix is $\boldsymbol{I_3}$ and $\|\cdot\|_F$ is the Frobenius norm, the InverseForm distance is calculated as

$$d_{if}(\boldsymbol{x}, t_{\boldsymbol{\theta}}(\boldsymbol{x})) = \left\| \hat{\boldsymbol{\theta}} - \boldsymbol{I_3} \right\|_F. \qquad (1)$$

We train the inverse-transformation network by reducing the Euclidean distance between predicted and ground-truth parameters, and use Equation (1) at inference time.

Geodesic distance: Homography transformations reside on an analytical manifold instead of a flat Euclidean space. The geodesic distance can capture these transformations, as explained in the recent works [27] [42]. Considering the ground-truth parameters as $\boldsymbol{\theta}$, we use the formulation of geodesic distance from both these works, as follows

$$d_{if}(\boldsymbol{x}, t_{\boldsymbol{\theta}}(\boldsymbol{x})) = \left\| \frac{\mathrm{Log}(\boldsymbol{\theta}^{-1}\hat{\boldsymbol{\theta}})}{\mathrm{Log}(\boldsymbol{I_3})} \right\|_F. \qquad (2)$$

To train the network using this formulation of geodesic distance, we are required to calculate gradient errors over the Riemannian logarithm, which does not have a closed-form solution. Thus, we use the method in [27] to project the homography Lie group onto a subgroup **SO(3)** [41] where the calculation of geodesic distance does not need the Riemannian logarithm. The formulation is given by

$$d_{if}(\boldsymbol{x}, t_{\boldsymbol{\theta}}(\boldsymbol{x})) = \arccos\left[\frac{\mathrm{Tr}\,(\boldsymbol{P}) - 1}{2}\right] + \lambda\,\mathrm{Tr}\,(\boldsymbol{R_\pi^T}\boldsymbol{R_\pi}), \qquad (3)$$

where λ is a weighting parameter set at 0.1 and the projection $\boldsymbol{P}$ onto the rotation group **SO(3)** is given as $\boldsymbol{P} = \boldsymbol{U}diag\{1, 1, det(\boldsymbol{UV})^T\}\boldsymbol{V}^T$ and the projection residual $\boldsymbol{R_\pi}$ is calculated as

$$\boldsymbol{R_\pi} = \boldsymbol{\theta}^{-1}\hat{\boldsymbol{\theta}} - \boldsymbol{P}. \qquad (4)$$

This formulation of geodesic distance can be used to train the inverse-transformation network. During inference, we insert $\boldsymbol{\theta} = \boldsymbol{I_3}$ in Equation (4) to compute the distance between the two inputs to the inverse-transformation network.

For using geodesic distance, there are 8 degrees of freedom in the 3×3 homography matrix. The inverse-transformation network predicts 8 values and we set the 9th value to 1. On the other hand, the matrix has 6 degrees of freedom if we assume only 2D affine transformations. Hence, the network must predict 6 values if we use the Euclidean distance measure. We have compared the effect of using these distance measures in Section 5.

3.4. Using InverseForm as a loss function

We first train the inverse-transformation network using boundary maps of images sampled from the target dataset. We apply the STN [21] to generate the transformed versions of boundary images. We could also do this by randomly sampling transformation parameters using the method described in [27], however this leads to lesser realistic transformations. Then, we use these images and their affine transformations as inputs to the inverse-transformation network. Before feeding boundary maps to the network, we
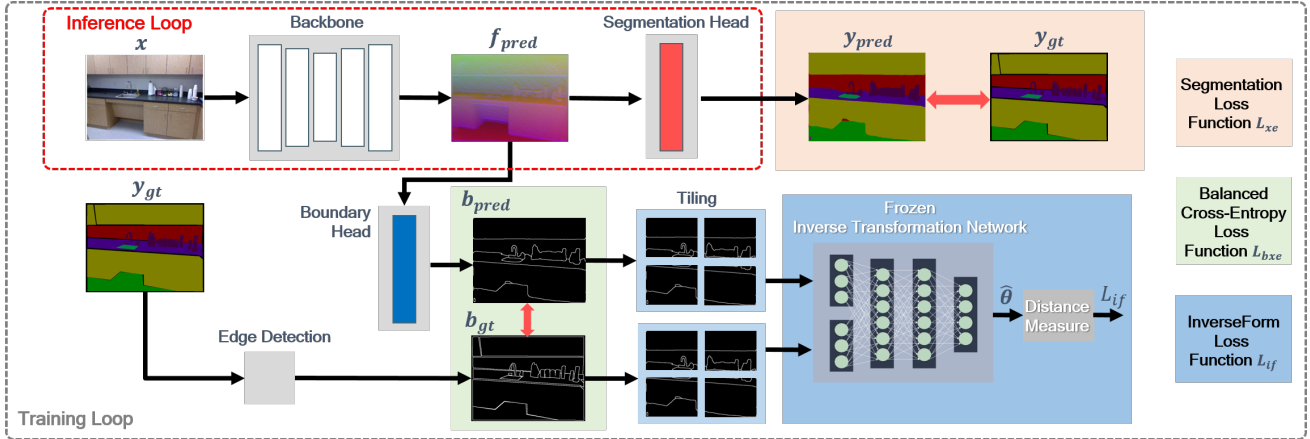
4

Figure 4: Overall framework for our proposed boundary-aware segmentation.

split the images into smaller tiles. Tiling is done in a way that the local context can be preserved along with the global context. Ideally, the best tiling dimension should provide a balance between local and global contexts. We discuss the effect of tiling dimension in the Appendix. Training details and hyperparameters used can be found in the Appendix.

Once we train the inverse-transformation network, we freeze its weights and use it as a loss function in a boundary-aware segmentation setting. Using networks as loss functions is a common practice in generative adversarial network literature [16]. However, the distance computed based on inverse-transformation network outputs doesn't fall in the category of discriminator or adversarial losses, according to the definition given by [13].

To model the loss function, we assume the predicted boundary $\boldsymbol{b_{pred}}$ is a transformed version of the ground truth boundary label $\boldsymbol{b_{gt}}$. i.e. $\boldsymbol{b_{pred}} = t_{\boldsymbol{\theta}}(\boldsymbol{b_{gt}})$. Hence, we can formulate our InverseForm loss function in terms of the spatial distance calculated in Section 3.5. This function first splits the input boundaries $\boldsymbol{b_{pred}}$ and $\boldsymbol{b_{gt}}$ into N smaller tiles $\boldsymbol{b_{pred,j}}$ and $\boldsymbol{b_{gt,j}}$. Next, it passes the inputs through the inverse-transformer network and calculates spatial distance in one of the ways described in Section 3.2. The formulation of our InverseForm loss function L_{if} is given by

$$L_{if}(\boldsymbol{b_{pred}}, \boldsymbol{b_{gt}}) = \sum_{j=1}^{N} d_{if}(\boldsymbol{b_{pred,j}}, \boldsymbol{b_{gt,j}}) \,. \quad (5)$$

3.5. Boundary-aware segmentation setup

We train various architectures using our loss function, with both single-task and multi-task settings in Section 4. For multi-task settings, we directly use the architecture used by MTI-Nets [44] and ASTMT [31] and add our Inverse-Form loss term to the boundary loss.

To train single-task architectures using InverseForm loss, we use a simple boundary-aware segmentation setup, as dis-

played in Figure 4. This setup could be used over any backbone. Consider $\boldsymbol{x}$ is the input image, $\boldsymbol{y_{gt}}$ is the ground-truth segmentation map and $\boldsymbol{y_{pred}}$ is the network prediction. The network produces intermediate features $\boldsymbol{f_{pred}}$, which are fed into a segmentation backbone to produce the output segmentation. We pass the features $\boldsymbol{f_{pred}}$ into an auxiliary head to produce a boundary map $\boldsymbol{b_{pred}}$. Optionally, this boundary map could be concatenated with the features $\boldsymbol{f_{pred}}$ and then fed to the segmentation head to produce the output $\boldsymbol{y_{pred}}$. We don't add this supervision in our experiments to ensure zero computational overhead.

Pixel-wise cross-entropy loss L_{xe} is used on the segmentation output. The ground-truth $\boldsymbol{y_{gt}}$ is passed through a So-bel filter [2] to produce a binary boundary map. We integrate the InverseForm loss and weighted cross-entropy loss L_{bxe} over the predicted boundary and ground-truth maps into the overall loss function. Specifically, we define the overall loss function as

$$\begin{aligned} L_{total} = L_{xe}(\boldsymbol{y_{pred}}, \boldsymbol{y_{gt}}) + \beta L_{bxe}(\boldsymbol{b_{pred}}, \boldsymbol{b_{gt}}) \\ + \gamma L_{if}(\boldsymbol{b_{pred}}, \boldsymbol{b_{gt}}) \quad (6) \end{aligned}$$

where L_{xe} is the cross-entropy loss, L_{bxe} is weighted cross-entropy loss, and L_{if} is the InverseForm loss defined in (5). β, γ are the scaling parameters for the weighted cross-entropy loss and InverseForm loss terms, respectively. They are treated as hyperparameters and we discuss the effects of these terms in the Appendix.

4. Experimental Results

Datasets: We present performance scores for various tasks on three datasets, NYU-Depth-v2 [37], PASCAL [15] and Cityscapes [9]. We use the original 795 training and 654 testing images for NYU-Depth-v2. For PASCAL, we use a split from PASCAL-Context, which has annotations for segmentation, edge-detection, and human parts. We also use la-

Network	Tasks	mIoU	mBA
HRNet-w18	S	33.18	37.46
HRNet-w18 **w/ ours**	S+E	**34.79**	**40.72**
PAD-HRNet18	S+D	32.80	38.10
PAD-HRNet18	S+D+E+N	33.10	42.69
PAD-HRNet18 **w/ ours**	S+D+E+N	**34.70**	**43.24**
MTI-HRNet18	S+D	35.12	42.44
MTI-HRNet18	S+D+E+N	37.49	43.26
MTI-HRNet18 **w/ ours**	S+D+E+N	**38.71**	**45.59**
HRNet-w48	S	45.70	56.01
HRNet-w48 **w/ ours**	S+E	**47.42**	**59.34**

Table 1: Comparing baselines for NYU-Depth-v2 using HRNet-w18 and HRNet-w48 backbones in both single task and multi-task learning using our loss. In multi-task settings, S:Semantic segmentation, D:Depth, E:Edge detection, N:Surface normal estimation. Consistent improvement in segmentation mIoU and boundary mBA is visible.

Network	Backbone	mIoU	Pixel-acc
TD-Net [19]	PSPNet-50	43.5	55.2
PAD-Net [48]	ResNet-50	50.2	75.2
PAP-Net [53]	ResNet-50	50.4	76.2
MTI-Net [44]	HRNet-w48	49.0	-
RSP [25]	ResNet-152	46.5	73.6
ACNet [20]	ResNet-50	48.3	-
Malleable 2.5D [47]	ResNet-101	50.9	76.9
SA-Gate [35]	ResNet-101	52.4	77.9
Ours	ResNet-101	**53.1**	**78.1**

Table 2: Comparison with state-of-the art methods on NYU-Depth-v2. Training with InverseForm consistently improves results, and outperforms state-of-the-art methods.
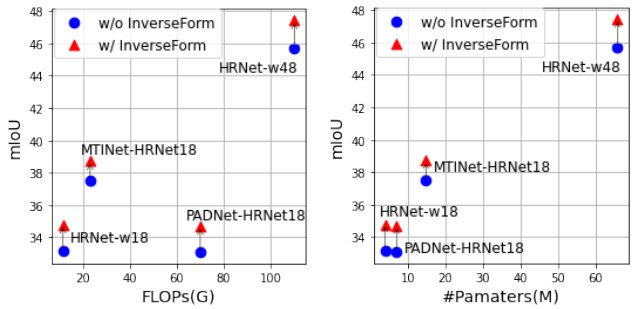

Figure 5: Comparison of mIoU v/s FLOPs and mIou v/s # params for different schemes on NYU-Depth-v2 dataset.

bels for surface normals and saliency from [31]. This split for PASCAL is used by ASTMT [31] and MTI-Nets [44] for a multi-task framework. For Cityscapes, we use their 2975/500/1525 train/val/test splits to report performance. Models reporting on test set are trained using train+val set. **Evaluation metrics:** For single-task learning settings, we measure semantic segmentation performance. This is evaluated using mean intersection-over-union (mIoU) and pixel-accuracy. To show improvement on boundary regions, we use our implementation of multi-class Mean Boundary Accuracy(mBA) from [8]. This is computed over segmentation outputs. For multi-task learning settings on PASCAL, the model is trained for five tasks: semantic segmentation, boundary detection, saliency estimation, human parts and surface normal estimation. We measure mIoU for semantic segmentation, human parts, and saliency estimation. The mean error in predicted angles is calculated for surface normal, and the optimal dataset F-measure is used to evaluate boundary detection. We measure depth and semantic segmentation for NYU-Depth-v2. Depth estimation is measured using root mean square error (rmse). Finally, the multi-task performance of model m, Δ_m [31] is given by the relative increase wrt. single-task performance for all tasks: $\Delta_m = \frac{1}{T} \sum_{i=1}^{T} (-1)^{l_i} \frac{(M_{m,i} - M_{b,i})}{M_{b,i}}$. Here, $M_{m,i}$ and $M_{b,i}$ are metrics for the multi-task model and single-task model for task i. l_i is 1 when the metric is positive v/s 0 when the metric is negative.

4.1. Results on NYU-Depth-v2

For NYU-Depth-v2, we report scores using three backbone architectures, i.e. HRNet-w18, HRNet-w48 (adapted from [46]) and ResNet-101 [17]. For the vanilla HRNet-w18 and HRNet-w48 backbones, we train with Inverse-

Form loss using the method described in Figure 4. We also add the InverseForm loss term to the boundary loss function within available multi-task baselines as implemented by MTI-Net [44] over HRNet-18. We used their implementation of PAD-Net [48] in multi-task settings. These results are given in Table 1. For computational analysis, we plot the no. of parameters v/s mIoU and GFLOPs v/s mIoU curves in Figure 5. We remove the boundary detection head before running inference. Hence, our method increases the mIoU score without any extra computations during inference.

The best scores reported on NYU-Depth-v2 are achieved by the multi-scale inference on RGBD inputs using SA-Gates [35] over ResNet-101 backbone and DeepLab-V3+ [5] as the decoder. We use their implementation and train this model with our InverseForm loss and the boundary-aware framework shown in Figure 4. Our model outperforms the SA-Gates baseline by 0.7% mIoU and other state-of-the-art models by a large margin on NYU-Depth-v2, as shown in Table 2. We provide the hyperparameters for our experiments and the multi-task results for this experiment in the Appendix.

Network	InverseForm	Seg (↑)	Edge (↑)	Saliency (↑)	Parts (↑)	Normals (↓)	$\Delta_m(\%)$ (↑)
SE-ResNet-26+ASTMT		64.61	71.0	64.70	57.25	15.00	-0.11
	✓	**65.13**	**71.4**	**65.29**	**57.93**	15.07	**0.49**
SE-ResNet-50+ASTMT		68.00	72.4	66.13	61.10	14.60	-0.04
	✓	**68.83**	**72.5**	**67.50**	**61.13**	**14.55**	**0.95**
SE-ResNet-101+ASTMT		68.51	73.5	67.72	63.41	**14.37**	-0.6
	✓	**70.14**	**73.7**	**68.70**	**64.76**	14.55	**0.39**
ResNet-18+MTI-Net		65.70	73.9	66.80	61.60	14.60	3.84
	✓	**65.96**	**74.2**	**67.23**	**61.71**	**14.52**	**4.34**
HRNet-w18+MTI-Net		64.30	73.4	68.00	62.10	14.80	2.74
	✓	**65.12**	**73.6**	**68.61**	**62.53**	**14.67**	**3.72**

Table 3: Training state-of-the-art multi-task learning methods on PASCAL by adding InverseForm loss over boundary detection. HRNet-18 and SE-Resnet backbones are used in a multi-task setting and mIoU scores for segmentation, saliency, human parts and surface normal tasks as well as F-scores for boundary detection are compared with the original results. InverseForm loss consistently improves results barring a few cases.

4.2. Results on PASCAL

For the PASCAL dataset, we report scores comparing with the state-of-the-art multi-task learning results from ASTMT [31] and MTI-Net [44]. As implemented by ASTMT, we use a base architecture of Deeplab-V3+ [6] with a ResNet encoder and squeeze-and-excite (SE) [18] blocks. We train this network both with and without the InverseForm loss function over the boundary detection task. We also add our loss term over boundary detection task to train the HRNet-w18 backbone from MTI-Net. On this benchmark, we observe consistent improvement in performance by training with our InverseForm loss. These results can be found in Table 3. We calculate the multi-task metric relative to the single-task scores reported by the respective papers. The hyperparameters used for our training algorithm are shown in the Appendix.

4.3. Results on Cityscapes

We report results on the Cityscapes *val* and *test* splits. As a baseline, we use an HRNet-48+OCR [49] model. We train the model by adding InverseForm both with and without hierarchical multi-scale attention (HMS) [40]. We also use their auto-labeled coarse annotations. We train with our boundary-aware segmentation scheme from Section 3.5. We only use the backbone during inference, as illustrated in Figure 4; the network does not require any extra compute compared to the baseline.

We compare our approach to state-of-the-art boundary-aware segmentation frameworks such as Gated-SCNN [39] and SegFix [50], quantitatively in Table 4 and qualitatively in Figure 6. We outperform SegFix using an HRNet-OCR backbone by 0.3% mIoU, and using a GSCNN model with WideResNet-38 [51] backbone by 1.1% mIoU. For GSCNN, we show that our method is complimentary to their work. We train their network by adding the Inverse-
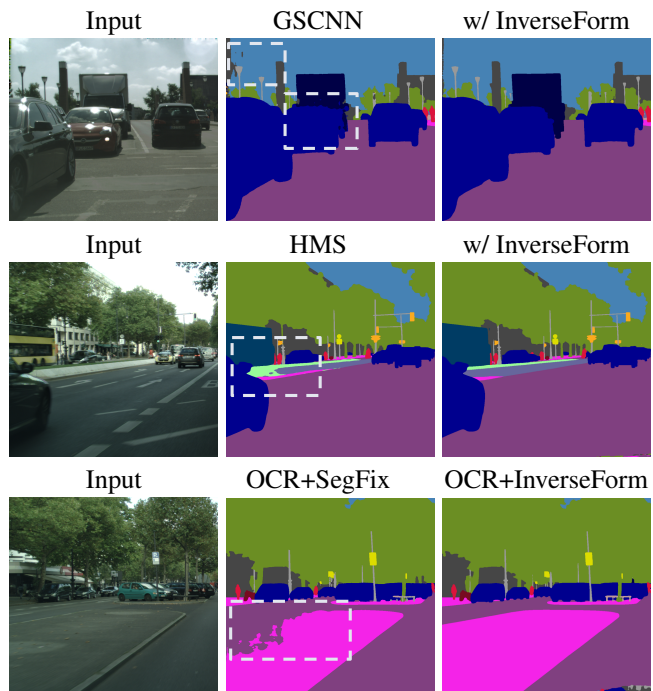

Figure 6: Cityscapes results showing visual effect of training different baselines with InverseForm loss. The structure of predicted outputs is improved in highlighted regions due to boundary-aware segmentation.

Form loss term to their existing pixel-based boundary loss, outperforming their baseline by 1.6% mIoU. In Table 4, all networks are pretrained on Mapiliary Vistas [33]. They use different training sets provided by Cityscapes. These contain coarse and finely-annotated training samples, which are available in the dataset. In Figure 6, we use the images provided by authors of [50] and [39], and use the state-of-the-

Method	Backbone	Split	F	C	mIoU
Naive-student [4]	WRN41	Test	✓	✓	85.2
GSCNN [39]	WRN38	Test	✓		82.8
HRNet-OCR [50]	HRNet48	Test	✓	✓	84.2
OCR+SegFix [50]	HRNet48	Test	✓	✓	84.5
OCR **w/ ours**	HRNet48	Test	✓	✓	84.8
HMS [40]	HRNet48	Test	✓	✓	85.1
HMS **w/ ours**	HRNet48	Test	✓	✓	**85.6**
GSCNN [39]	WRN38	Val	✓		81.0
GSCNN+SegFix	WRN38	Val	✓		81.5
GSCNN **w/ ours**	WRN38	Val	✓		82.6
GSCNN **w/ ours**	WRN38	Val	✓	✓	84.0
HMS	HRNet48	Val	✓	✓	86.7
HMS **w/ ours**	HRNet48	Val	✓	✓	**87.0**

Table 4: Our method compared to various state-of-the-art algorithms on Cityscapes. The models reporting on test split are trained using training+validation data. F:Fine annotations, C:Coarse annotations
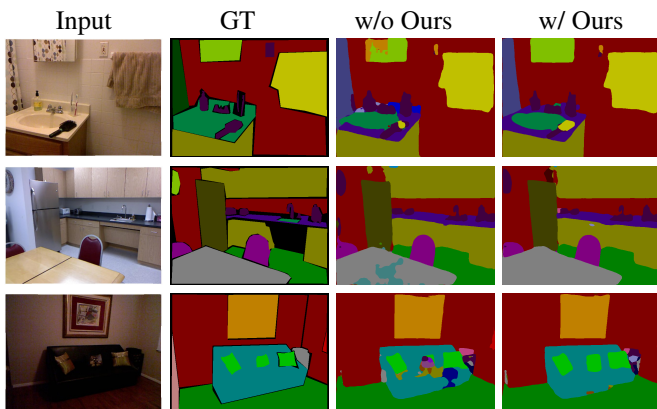

Figure 7: NYU-Depth-v2 results showing visual effect of training SA-Gates [35] baseline with InverseForm loss. Clear improvement is visible in the structure of predicted outputs due to boundary-aware segmentation.

art model provided by [40] to generate baseline predictions. We then use the same baselines trained with InverseForm from Table 4 to generate our predictions. As seen in Table 4 and Figure 6, all models trained using InverseForm loss consistently improve compared to their baselines. We are rank 3 on the public Cityscapes leaderboard at the time of submission.

5. Ablation Studies

Searching for the best inverse-transformation: We analyze how our current dense inverse-transformation architecture performs compared to the convolutional architecture used in AET [52]. To reproduce their result, we use their

pretrained model and fine-tune with tiled NYU-Depth-v2 boundary images, using their training strategy. We also conduct experiments with both formulations of distance given in Section 3.3 on NYU-Depth-v2 using vanilla HRNet-w48 and HRNet-w18 as backbones, and the architecture defined in Figure 4. The loss functions used are Inverse-Form using geodesic distance measure $L_{if,geo}$ and Inverse-Form using the Euclidean measure $L_{if,euc}$. Our experiments show a clear improvement in performance for the inverse-transformer architecture v/s the one used by AET. We hypothesize that the reason for better performance is that convolutions are translation-invariant. While the framework used by AET stacks two convolutional embeddings, the local resolution takes a hit compared to global context.

model	Loss net.	$L_{if,geo}$	$L_{if,euc}$	mIoU
	AET	✓		47.19
	Ours	✓		47.28
HRNet-w48	AET		✓	47.03
	Ours		✓	**47.42**
	AET	✓		33.82
	Ours	✓		**34.84**
HRNet-w18	AET		✓	33.97
	Ours		✓	34.79

Table 5: Searching for the optimal InverseForm loss

Distance function: Our results for using the geodesic distance v/s Euclidean distance however, do not show a clear winner. This is partly due to the reason that the formulation in Section 3.3 for geodesic distance can lead to exploding gradients easily. This severely limits the search-space for hyperparameters. Euclidean distance on the other hand might not model perspective homography best, but using this gives us a wider search-space and hence a more consistent improvement. We recognize the need to explore the benefits of using geodesic distance, and hence continue our research on the optimal InverseForm network architecture.

6. Conclusion

We propose a distance-based boundary-aware segmentation method that consistently improves any semantic segmentation backbone it is plugged in. During training, the boundary detection and segmentation are jointly optimized in a multi-task learning framework. At inference time, it does not require any additional computational load. The distance-based measure outperforms cross-entropy based measures while efficiently capturing the boundary transformation. We show empirically that our scheme achieves superior segmentation accuracy and better structured segmentation outputs. We continue to look for the optimal architecture and distance measure for this method, as we show some room for improvement in our ablation experiments.

References

[1] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Semantic segmentation with boundary neural fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3602–3610, 2016. 2

[2] John Canny. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698, 1986. 5

[3] Liang-Chieh Chen, Jonathan T Barron, George Papandreou, Kevin Murphy, and Alan L Yuille. Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4545–4554, 2016. 2

[4] Liang-Chieh Chen, Raphael Gontijo Lopes, Bowen Cheng, Maxwell D. Collins, Ekin D. Cubuk, Barret Zoph, Hartwig Adam, and Jonathon Shlens. Naive-student: Leveraging semi-supervised learning in video sequences for urban scene segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 8

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 2, 6

[6] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018. 7

[7] Dongcai Cheng, Gaofeng Meng, Shiming Xiang, and Chunhong Pan. Fusionnet: Edge aware deep convolutional networks for semantic segmentation of remote sensing harbor images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(12):5769–5783, 2017. 2

[8] Ho Kei Cheng, Jihoon Chung, Yu-Wing Tai, and Chi-Keung Tang. Cascadepsp: Toward class-agnostic and very high-resolution segmentation via global and local refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8890–8899, 2020. 6

[9] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1, 2, 5

[10] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-aware feature propagation for scene segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6819–6829, 2019. 2

[11] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic correlation promoted shape-variant context for segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8885–8894, 2019. 2

[12] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2393–2402, 2018. 2

[13] Hao-Wen Dong and Yi-Hsuan Yang. Towards a deeper understanding of adversarial losses under a discriminative adversarial network setting, 2020. 5

[14] Nikita Dvornik, Konstantin Shmelkov, Julien Mairal, and Cordelia Schmid. Blitznet: A real-time deep network for scene understanding. In *Proceedings of the IEEE international conference on computer vision*, pages 4154–4162, 2017. 1

[15] Mark Everingham, S. Eslami, Luc Van Gool, Christopher Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111, 01 2014. 2, 5

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 5

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[18] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 7

[19] Ping Hu, Fabian Caba, Oliver Wang, Zhe Lin, Stan Sclaroff, and Federico Perazzi. Temporally distributed networks for fast video semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8818–8827, 2020. 6

[20] Xinxin Hu, Kailun Yang, Lei Fei, and Kaiwei Wang. Acnet: Attention based network to exploit complementary features for rgbd semantic segmentation. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1440–1444. IEEE, 2019. 6

[21] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 2017–2025, 2015. 3, 4

[22] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[23] Iasonas Kokkinos. Pushing the boundaries of boundary detection using deep learning. In *4th International Conference on Learning Representations, ICLR 2016*. International Conference on Learning Representations, ICLR, 2016. 1, 2

[24] Iasonas Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings*

of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6129–6138, 2017. 3

[25] Shu Kong and Charless C Fowlkes. Recurrent scene parsing with perspective understanding in the loop. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 956–965, 2018. 6

[26] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected CRFs with Gaussian edge potentials. In *Advances in neural information processing systems*, pages 109–117, 2011. 2

[27] Feng Lin, Haohang Xu, Houqiang Li, Hongkai Xiong, and Guo-Jun Qi. Aetv2: Autoencoding transformations for self-supervised representation learning by minimizing geodesic distances in lie groups, 2019. 3, 4

[28] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017. 2

[29] Sifei Liu, Shalini De Mello, Jinwei Gu, Guangyu Zhong, Ming-Hsuan Yang, and Jan Kautz. Learning affinity via spatial propagation networks. In *Advances in Neural Information Processing Systems*, pages 1520–1530, 2017. 2

[30] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 1, 2

[31] Kevis-Kokitsi Maninis, Ilija Radosavovic, and Iasonas Kokkinos. Attentive single-tasking of multiple tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1860, 2019. 1, 2, 3, 5, 6, 7

[32] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(5):530–549, 2004. 2

[33] G. Neuhold, T. Ollmann, S. R. Bulò, and P. Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5000–5009, 2017. 7

[34] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2017. 2

[35] Chen Qian, Hongsheng Li, and Gang Zeng. Bi-directional cross-modality feature propagation with separation-and-aggregation gate for rgb-d semantic segmentation. 6, 8

[36] Javier Ribera, David Güera, Yuhao Chen, and Edward J. Delp. Weighted hausdorff distance: A loss function for object localization. *CoRR*, abs/1806.07564, 2018. 3

[37] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012. 2, 5

[38] Ke Sun, Yang Zhao, Borui Jiang, Tianheng Cheng, Bin Xiao, Dong Liu, Yadong Mu, Xinggang Wang, Wenyu Liu, and Jingdong Wang. High-resolution representations for labeling pixels and regions. *arXiv preprint arXiv:1904.04514*, 2019. 1

[39] Towaki Takikawa, David Acuna, Varun Jampani, and Sanja Fidler. Gated-scnn: Gated shape cnns for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5229–5238, 2019. 1, 2, 7, 8

[40] Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020. 1, 7, 8

[41] Camillo J Taylor and David J Kriegman. Minimization on the lie group so (3) and related manifolds. *Yale University*, 16(155):6, 1994. 4

[42] O. Tuzel, F. Porikli, and P. Meer. Learning on lie groups for invariant detection and tracking. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 4

[43] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey, 2020. 3

[44] Simon Vandenhende, Stamatios Georgoulis, and Luc Van Gool. Mti-net: Multi-scale task interaction networks for multi-task learning. In *European Conference on Computer Vision*, pages 527–543. Springer, 2020. 1, 2, 3, 5, 6, 7

[45] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 1, 2

[46] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition, 2020. 6

[47] Yajie Xing, Jingbo Wang, and Gang Zeng. Malleable 2.5d convolution: Learning receptive fields along the depth-axis for rgb-d scene parsing, 2020. 6

[48] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 675–684, 2018. 1, 6

[49] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation, 2020. 1, 2, 7

[50] Yuhui Yuan, Jingyi Xie, Xilin Chen, and Jingdong Wang. Segfix: Model-agnostic boundary refinement for segmentation. In *European Conference on Computer Vision*, pages 489–506. Springer, 2020. 1, 2, 3, 7, 8

[51] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference 2016*. British Machine Vision Association, 2016. 7

[52] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2547–2555, 2019. 3, 8

[53] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4106–4115, 2019. 6

[54] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017. 2