The Supplementary Material for Towards Comprehensive Representation Enhancement in Semantics-guided Self-supervised Monocular Depth Estimation

Jingyuan Ma*, Xiangyu Lei*, Nan Liu, Xian Zhao, and Shiliang Pu**

Hikvision Research Institute
{majingyuan,leixiangyu,liunan6,zhaoxian, pushiliang.hri}@hikvision.com

1 Boundary Triplet Margin Loss with Patch-Based Sampling Strategy in [2]

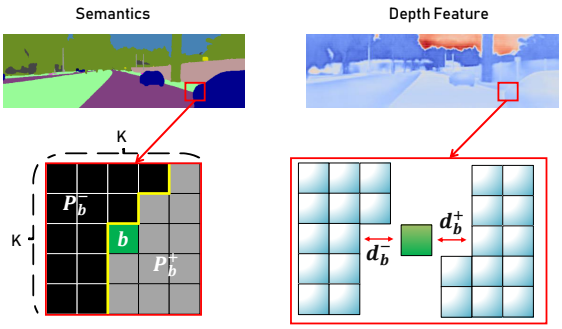

Fig. 1. The visualization of patch-based sampling strategy from [2]. Semantic labels are firstly divided by kernel of size $K \times K$ to formulate triplets for computing semantics-guided triplet loss. Within a patch, green pixel represent the anchor, gray pixels are its positives and black pixels are its negatives. The yellow line represents the semantic boundary within this patch. With the sampled triplets, the positive distance d_b^+ and negative distance d_b^- for this anchor are computed with the normalized depth features.

Semantic labels are firstly divided into patches by a kernel of size $K \times K$ and stride of one. The centers of patches are sampled as anchors, $\mathcal{P}_B$. Within each patch, for its anchor, $b \in \mathcal{P}_B$, positives, $\mathcal{P}_b^+$, are sampled from pixels of the same class as anchor b, while negatives, $\mathcal{P}_b^-$ are sampled from pixels of

* First two authors contribute equally to this work
** Corresponding Author

different classes from anchor b. The visualization of patch-based sampling process is shown in Fig 1. A triplet margin loss is computed with sampled triplets. Given the normalized depth feature F_d, positive distance d_b^+/negative distance d_b^- are defined as the distance between anchor and positives/negatives in feature space:

$$d_b^+ = \frac{1}{|\mathcal{P}_b^+|} \sum_{b^+ \in \mathcal{P}_b^+} (\sqrt{(F_d(b) - F_d(b^+))^2}) \tag{1}$$

$$d_b^- = \frac{1}{|\mathcal{P}_b^-|} \sum_{b^- \in \mathcal{P}_b^-} (\sqrt{(F_d(b) - F_d(b^-))^2}) \tag{2}$$

Then, for each boundary anchor $b \in \mathcal{P}_B$, a triplet marginal loss is defined as:

$$\mathcal{L}(\mathcal{P}_b) = \max(0, m + (d_b^+ - d_b^-)) \tag{3}$$

where $m = 0.3$ is the margin to separate positive distance, d_b^+, and negative distance, d_b^-, which are defined in Eq 1 and Eq 2. Concurrently, a boundary region mask, B, is generated to solve misclassification problem. At anchor b, the value of boundary mask is one if the number of positives and negatives of anchor b are larger than threshold T in a patch: $B(b) = \mathbb{I}[|\mathcal{P}_b^+|, |\mathcal{P}_b^-| > T]$. The final semantics-guided boundary triplet margin loss is applied only on boundary regions, which is defined as:

$$\mathcal{L}_{BT} = \sum_{b \in \mathcal{P}_B} \frac{\mathbb{I}[|\mathcal{P}_b^+|, |\mathcal{P}_b^-| > T] \cdot \mathcal{L}(\mathcal{P}_b)}{\mathbb{I}[|\mathcal{P}_b^+|, |\mathcal{P}_b^-| > T]} \tag{4}$$

where $\mathbb{I}$ is the indicator function, and $T = K - 1$ with $K = 5$ as that in [2]. The generated boundary region mask, B, is employed to assist our proposed minimum-distance based candidate sampling strategy by identifying the non-boundary region of each instance.

2 Cross-Task Similarity Computation Process

The visualization of cross-task similarity computation [2] is shown in Fig 2. Since such cross-task similarity is computed from the the feature of depth and segmentation at the same spatial location, it adequately helps aligning semantic boundaries with depth boundaries. Such alignment between semantic boundaries and depth boundaries can be affected by extra local neighboring information. Thus, applying window partition on the input query-key-value triplet could blur depth boundary, which might further lead to error at the boundaries of semantic segmentation. Therefore, we maintain the computation process as that in [2].

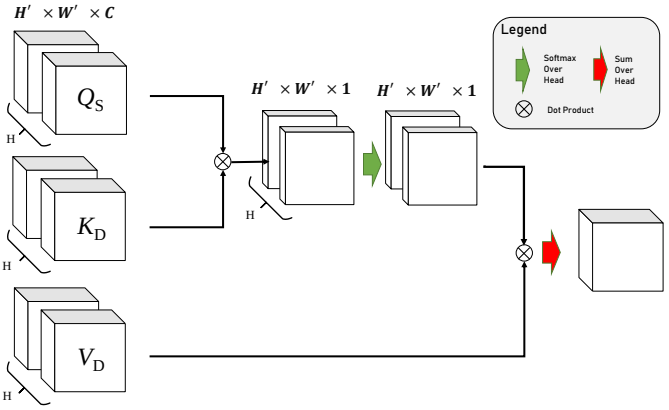

Fig. 2. The visualization of cross-task similarity computation [2]. Here, this computation process is illustrated on depth feature. The inputs are the key-value pair of depth feature and the query of semantic feature.

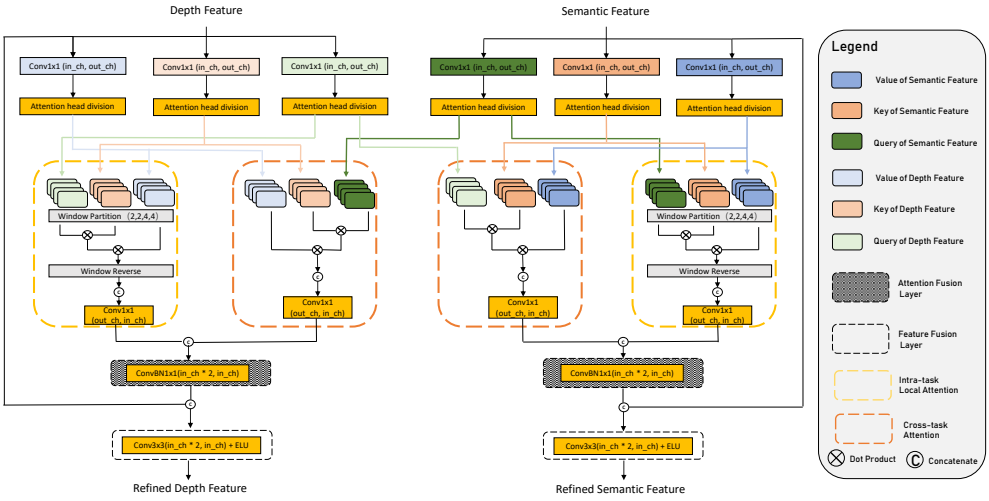

Fig. 3. The Detailed Schematic of IC-MHA block. Output Channel dimension: $out\_ch = in\_ch \times num\_head \times exp\_ratio$, where num_head is the number of attention heads and exp_ratio is the expansion ration of each head.

3 Detailed Schematic of IC-MHA module.

Detailed Schematic of IC-MHA module is shown in Fig 3.

4 Additional Discussions

4.1 Discussion on Proposed IC-MHA Module

The Role of Query-key-value Triplets in Our IC-MHA Module. Inside of our proposed IC-MHA module, intra-task local attention and cross-task attention is computed from projected query-key-value triplet of task-specific features. Only one query-key-value triplet is linearly projected for the feature of each task. The reason, besides computational efficiency, is that we would like the value of task-specific feature, $\{V_t|t \in \{D, S\}\}$, to serve as the private information of each task, which can also be viewed as the representation for task-specific feature uniqueness. Then, the query and key of task-specific feature, $\{(Q_t, K_t)|t \in \{D, S\}\}$ serves as the public information for computing intra-task local similarity and cross-task similarity.

Why Applying Multi-size Window Partition Process for Local Attention Computation. For both tasks, the feature within local neighborhood of the same instance is similar. Thus, both tasks could gain performance boost from local feature similarity within the same window. But, if the feature within a window is from different objects, such local feature similarity could be erroneous. One solution for this issue is to apply smaller windows at cross-instance region and bigger windows within an instance. Therefore, we firstly perform the projection of query-key-value triplet and then apply multi-size window partition on the projected triplets of each head, which is different from that in [3]: applying uniform-size window partition on input features then projecting the query-key-value triplet. To form the final attention map, our local attention, which is computed using different window size, is adaptively incorporated by a projection layer. From the perspective of depth estimation, the local depth variation for different objects is dissimilar: the depth variation for close-to-camera objects within a window is naturally smaller than that for far-away objects with the window of the same size. This implies that partitioning depth feature with smaller windows are more optimal for close-to-camera objects, while for far-away objects, larger windows are more suitable. Therefore, applying multi-size window partition on local attention computation encourages the distribution of depth feature closer to that of the actual depth value.

4.2 Discussion on Proposed Minimum-distance Based Candidate Mining Strategy

The Relationship of Sampled Boundary and Non-boundary Triplets. The core idea of our proposed minimum-distance based candidate mining strat-

egy is to encourage the distribution of depth feature more aligned with that of depth value within each instance. Since the prerequisite for refining depth feature at non-boundary region is a clear feature separation between boundary region of neighboring instances, the boundary triplets [2] are preserved. Then, the positives and negatives for non-boundary anchors are sampled from their closest boundary anchors' positives and negatives. In this way, the feature of non-boundary anchors are more similar to that of boundary pixels without damaging the cross-instance feature separation at boundary regions. The visualization of non-boundary triplet and its corresponding boundary triplet is shown in Fig 4.

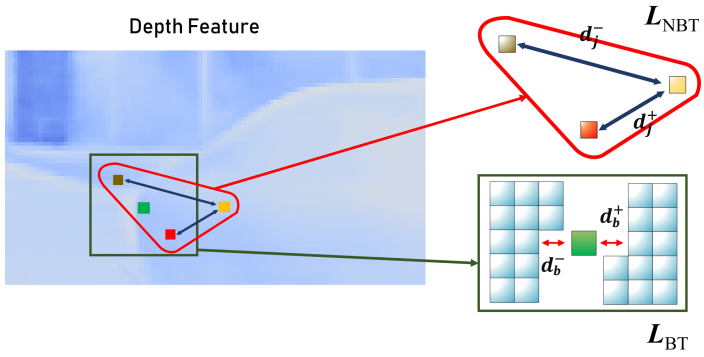

Fig. 4. The visualization of final triplets for boundary triplet loss $\mathcal{L}_{BT}$ [2](dark green box) and proposed hardest non-boundary triplet loss $\mathcal{L}_{NBT}$ (red triangle). Yellow dot represents non-boundary anchor, and green dot represents boundary anchor, which is also the center of the patch. Brown dot and red dot are the hardest negative and positive for non-boundary anchor, not for boundary anchor.

Why Employ Hardest Sample Strategy for Non-Boundary Triplet. The main reason for employing the hardest sample strategy [1] is to decouple non-boundary anchors' features from boundary anchors' features. If the hardest sample strategy is not adopted, the positives and negatives of non-boundary anchors will overlap with that of their corresponding boundary anchors. This could lead to the uniformity of all features within an instance, which is against the core idea of proposing minimum-distance based candidate mining strategy. Employing hardest sample strategy on non-boundary anchors can mitigate this problem, because such strategy forces an non-boundary anchor only focusing on its most dissimilar positive and most similar negative. At the same time, the corresponding boundary anchor remains utilizing all positives and negatives. Thus, there is a separation between each non-boundary anchor and its corresponding

boundary anchor in feature space. This prevents the uniformity between boundary features and non-boundary features.

5 Additional Ablation Study on Hyperparameters Settings

Table 1 shows the ablations on the number of head implemented in IC-MHA module. Thus, four heads is implemented in our final IC-MHA module. To demonstrate the excellent performance of transformer-style IC-MHA, we perform an ablation study on transformer based IC-MHA against skip-connection based IC-MHA. Both structure is trained with final loss $\mathcal{L}$. The result is shown in Table 2. Skip./Trans. represent skip-connection/transformer based IC-MHA module. Table 3 contains the ablation results on different window size used for intra-task local attention in IC-MHA module. The results show that applying $w_h = [2, 2, 4, 4]$ yields the best performance, which proves the effectiveness of applying windows of different size on different heads when computing the intra-task local attention. Table 4 shows the results of different levels that our proposed IC-MHA module is inserted. In general, inserting IC-MHA module on levels $s = 3, 2, 1, 0$ yields better performance.

Table 1. Ablations on number of head implemented in proposed IC-MHA module

H	AbsRel	SqRel	RMSE	RMSElog
2	0.106	0.759	4.580	0.182
4	**0.105**	0.734	**4.516**	**0.180**
6	0.106	**0.732**	4.519	**0.180**

Table 2. Ablations on IC-MHA module implementation base structure

s	AbsRel	SqRel	RMSE	RMSElog
Skip.	0.109	0.806	4.633	0.186
Tran.	**0.104**	**0.690**	**4.473**	**0.179**

Table 3. Ablations on window size(w_h) adopted in our proposed IC-MHA module

w_h	AbsRel	SqRel	RMSE	RMSElog
2,2,2,2	0.106	**0.733**	4.536	0.181
2,2,4,4	**0.105**	0.734	**4.516**	**0.180**
4,4,4,4	**0.105**	0.736	4.547	0.182

Table 4. Ablations on levels(s) that our proposed IC-MHA module is located

s	AbsRel	SqRel	RMSE	RMSElog
2,1,0	0.106	0.746	4.559	0.182
3,2,1	**0.105**	0.745	4.558	0.181
3,2,1,0	**0.105**	**0.734**	**4.516**	**0.180**

In Table 5, we compare the levels that $\mathcal{L}_{BT}$ and $\mathcal{L}_{NBT}$ apply on, *i.e.*, $\mathcal{S}_{BT}$ and $\mathcal{S}_{NBT}$ in final loss term $\mathcal{L}$. Applying non-boundary triplet loss on low-level feature degrades its performance, mainly because low-level features has much lower spatial resolution. Thus, the number of non-boundary pixels is too few leading to overly similarity between sampled non-boundary anchors' features in each instance. In contrast to [2], whose performance would downgrade when applying semantics-guided triplet and CMA module on level $s = 0$, our proposed

non-boundary triplet loss and IC-MHA module benefit significantly from feature at level 0. Feature at level 0 has the largest spatial resolution, which means low percentage of boundary pixels. Thus, solely emphasizing cross-task consistency [2] is not sufficient. In contrast, our proposed method focuses on thorough representational enhancement by addressing task-specific representational uniqueness and imposing extra refinement on non-boundary regions, which is the reason that our proposed method gains performance boost from feature at level $s = 0$.

In Table 6, different number of non-boundary anchors sampled at level $0(N_0)$ is compared. It demonstrates that adequate sampling of non-boundary anchors is necessary to avoid overly similarity of feature within the same object. So, we choose to sample 8000 non-boundary anchors at level 0.

Table 5. Ablations on the different levels that $\mathcal{L}_{NBT}$ and $\mathcal{L}_{BT}$ apply on

$\mathcal{S}_{NBT}$	$\mathcal{S}_{BT}$	AbsRel	SqRel	RMSE	RMSElog
3,2,1	3,2,1	0.106	0.728	4.530	0.181
3,2,1,0	3,2,1,0	0.107	0.730	4.531	0.181
2,1,0	3,2,1,0	0.106	0.711	4.530	0.180
1,0	3,2,1,0	**0.104**	**0.690**	**4.473**	**0.179**

Table 6. Ablations on the number of non-boundary anchors sampled(N_0) at level 0

N_0	AbsRel	SqRel	RMSE	RMSElog
4k	0.106	0.715	4.513	0.180
8k	**0.104**	**0.690**	**4.473**	**0.179**
16k	0.107	0.705	4.528	0.181

6 Additional Qualitative Examples

We include additional qualitative examples of our proposed method and our baseline [2], which is shown in Fig 5.

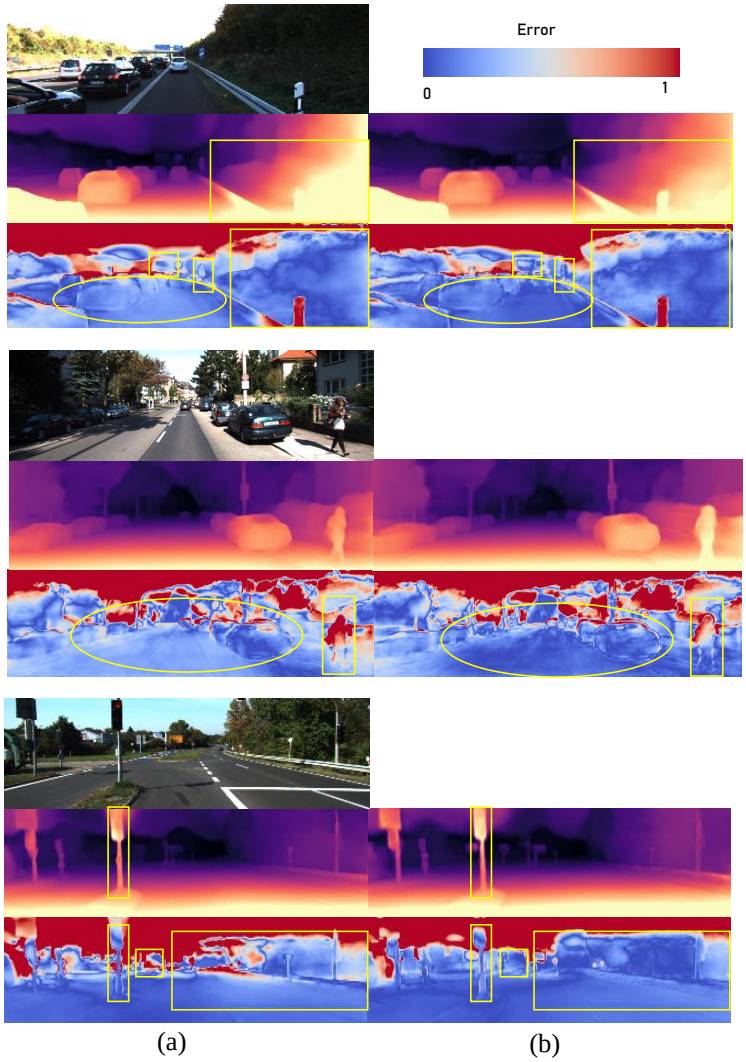

Fig. 5. Extra qualitative examples. (a)Depth output of [2](left) and ours(right). (b)Error map of [2](left) and ours(right).

References

1. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8958–8966 (2019)
2. Jung, H., Park, E., Yoo, S.: Fine-grained semantics-aware representation enhancement for self-supervised monocular depth estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 12642–12652 (October 2021)
3. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows (2021)