We thank all reviewers, and will incorporate all comments and suggestions in the final paper.

[Q1] More analysis of random sampling (R1,R4): The performance of the method is throttled by random sampling when the number of keypoints is small. However, the proposed modules (e.g., random dilation cluster, attentive points aggregation, etc.) can weaken the negative effect of random sampling and therefore, the performance of smaller number of keypoints can be improved by sampling more candidate points. We performed experiments with smaller number of keypoints and different dilation ratios on KITTI dataset to illustrate the effect of random sampling and the receptive field on performance. The results are displayed in Table 1. The distance thresholds for repeatability and precision are set to 0.5 m and 1.0 m, respectively. Note that we select keypoints based on the predicted saliency uncertainty. Denoting the number of keypoints as N_k, the number of sampled points as N_s and dilation ratio as α. In our current implementation, the number of sampled points is twice the number of selected keypoints (e.g., $N_s = 128$ if $N_k = 64$). According to Table 1, the performance significantly drops as N_k and N_s drop. Enlarging α can improve the performance due to the enlargement of the coverage of the whole network. However, when N_s is too small (e.g., $N_s = 64$), simply enlarging the receptive field is hard to cover the whole point cloud and the performance is greatly limited. Even with limitations, the proposed method achieves better performance than state-of-the-art. The given N_k (e.g, $N_k = 128$) is a reasonable number compared to the large scale point cloud. We also provide an alternative strategy for better performance with smaller N_k. The high efficiency of our method permits us to sample more candidate points for a smaller N_k, which does not cause a significant increasing on runtime. For example, if we need $N_k = 32$, we can set $N_s = 256$ and only select 32 keypoints from them. As shown in the bottom three lines of Table 1, the performance is significantly improved for small N_k if we sample more candidate points. The results indicate that sampling method is not a primary constraint on performance when the coverage of the network is large enough.

N_k	N_s	α	Repeatability	Precision
32	64	2	0.301	0.385
32	64	4	0.355	0.451
32	64	6	0.360	0.457
64	128	2	0.479	0.528
64	128	4	0.537	0.587
64	128	6	0.538	0.587
128	256	2	0.665	0.658
128	256	4	0.697	0.688
128	256	6	0.696	0.688
32	128	2	0.452	0.509
32	256	2	0.627	0.628
32	512	2	0.707	0.691

Table 1: The performance of different number of keypoints N_k, number of sampled points N_s and dilation ratio α.

[Q2] Metrics of keypoint detection (R1): We agree that recall should be considered as an evaluation metric to evaluate the keypoint detector. However, due to the lack of ground truth for keypoint detector, it is intractable to define the recall. Nonetheless, precision measures the performance of keypoint detector comprehensively, it relates to the repeatability, informativeness of generated keypoints and the effectiveness of the descriptor. We think the metrics provided in the paper are sufficient to evaluate the performance of the proposed keypoint detector and descriptor.

[Q3] More ablations (R1): We performed ablation studies on the weight and temperature t in the proposed matching loss. As shown in Table 2, the introduction of weight in matching loss improves the performance of the descriptor. Precision with different t is shown in Table 3. The soft assignment can not represent nearest neighbor search well if t is too large (e.g., $t = 0.5$). In our implementation, $t = 0.1$ is a proper choice and the performance will not change significantly when $t < 0.1$. Based on the suggestions of the reviewer, we will add more ablations in the final paper.

Number of keypoints	128	256	512
With weight	0.658	0.742	0.791
Without weight	0.634	0.721	0.769

Table 2: Precision with and without weight in matching loss.

[Q4] Reported runtime of USIP (R2): The runtime reported in paper of USIP does not include the time of farthest point sampling (FPS) and the calculation of descriptor. The time-consuming FPS is implemented in the dataloader according to the released code of USIP and they only reported the processing time of the detector network itself. Thus, we re-calculate the runtime including FPS and descriptor generation using the released code on our own platform.

[Q5] Saliency estimation (R2): We do not estimate saliency for all points in the point cloud. Instead, we only randomly sample several candidate points and use the proposed random dilation cluster as well as an attention mechanism to aggregate neighbor points and estimate the saliency. Thus, saliency estimation is only performed on sampled candidate points rather than all points. There exists minor writing typos of the denotations in line 90: we generate keypoints $\mathbf{X} \in \mathbb{R}^{M \times 3}$ and saliency uncertainties $\mathbf{\Sigma} \in \mathbb{R}^{M}$ rather than $\mathbf{X} \in \mathbb{R}^{N \times 3}$ and $\mathbf{\Sigma} \in \mathbb{R}^{N}$, where M is much smaller than N. We will correct it in the final paper.

t	128	256	512
$t = 0.1$	0.658	0.742	0.791
$t = 0.5$	0.625	0.704	0.755
$t = 0.01$	0.656	0.742	0.792

Table 3: Precision with different temperature t in matching loss.

[Q6] Concerns of too different sampled points (R4): The performance of the proposed method is stable when the coverage of the network is large enough (see **[Q1]** in this rebuttal document). The attentive mechanism tends to generate informative points in its receptive field and the network gives high weights to stable and informative keypoints. With a large coverage, the detection results cover most of the informative points, which are stable and consistent in different point clouds. Thus, the network can generate consistent keypoints even the sampled points in two point clouds are very different. As shown in the bottom three lines of Table 1, even with only 32 keypoints, our network achieves a high repeatability with a large N_s, which indicates the stability of the detected keypoints in different point clouds.