In the supplementary material, we first illustrate some technical details about the weights in Eq. (3), the position information $P$, and some experiment issues. Then, we proof the Eq. (6) in the paper. Third, experiments on different resolutions of latent feature, pure model capacity, and refining transformer-based segmentation results help understand CRM. Finally, additional visualizations show CRM’s refinement performance.

1. Technical details

Owing to space limitations, we do not explain some details on paper. Here, we will illustrate the weights in Eq. (3) and the position information $P$ in detail.

For the weights $w_k$, $k \in \{1, 2, 3, 4\}$ in Eq. (3), we present its details in Fig. 1. After finding the red supporting points $z_k$, $k \in \{1, 2, 3, 4\}$ of blue queried point $x$, we calculate the area value $a_k$ between $z_k$ and $x$. Then swap the symmetric area values $a_k$ about the $x$ point to be weights $w_k$. The final prediction is the weighted average of the predictions of the supporting points.

For position information $P$, in the paper, it consists of the refinement target position $C_t$, the relative target coordinate offset $C_r$, and the ratio $r$ between feature and target [3]. We also normalize the feature map coordinate $C_f$ and the refinement target coordinate $C_t$ to align. In Fig. 2, all components are drawn from left to right, top to down in a simple example. The upper row is the normalized $C_f$, normalized $C_t$, and the relative offset $C_r$ between them. Each item of $C_f$ is the offset vector of blue point on $C_t$ from corresponding red supporting point on $C_f$. The offset is showed in “details of $C_f$” in the lower row. And the way to find the red points of blue point is illustrated in Sec. 3.3. What’s more, the rightest part of lower row is a more complex but common example of $C_f$.

The coarse masks, from FCN [11], Deeplabv3+ [2], RefineNet [10], and PSPNet [15], are all trained on Pascal VOC [8]. And we utilize MaskFormer [4] and SegFormer [12]’s pretrained weights on ADE20K from their Github repositories.

In quantitative comparison, we train the SegFix [14] on the same mask perturbing dataset and keep other settings consistent. We use MGMatting pretrained on RWP [13]. To erase the performance degradation caused by the mask-insensitive matting setting, we update cases that have at least 0.80 IoU with coarse inputs after inference.

2. Proof of Eq. (6)

For convenience, we suppose that each entry of the matrix $A \in \mathbb{R}^{2 \times m}$ is sampled from $\mathcal{N}(0, 2/m)$. We now define the fixed feature space $\mathbb{F} \subset \mathbb{R}^m$:

$$-1 \leq f_i^T f_j \leq 1, \quad \|f_i\|_2 = \|f_j\|_2 = 1, \quad \forall f_i, f_j \in \mathbb{F}.$$
Show that
\[ \dim(\text{conv}(\phi(Af))) \geq \dim(\phi(Af)), \quad f \in \mathbb{F}, \]
where \( \phi : \mathbb{R} \to \mathbb{R} \) is the ReLU activation, \( \text{conv}(\cdot) \) is the convex hull of a bounded set, and \( \dim \) is the covering number of a compact set\(^1\).

**Proof.** Before proving the main results, we first present two useful lemmas.

**Lemma 1.** Let \( x_1, x_2 \in \mathbb{R}^m \), \( \|x_1\|_2 = \|x_2\|_2 = 1 \), and \( x_1 \perp x_2 = z \). We have
\[
\mathbb{E}_w(\phi(w^\top x_1)\phi(w^\top x_2)) = \frac{\sqrt{1 - z^2}}{\pi} + z(\pi - \arccos z),
\]
where \( w \sim \mathcal{N}(0, 2) \) and \( \phi(\cdot) \) is the ReLU function.

This lemma is a direct corollary of the results in \cite{som} (see Table 1 therein). We then present the norm preserving property \cite{som}.

**Lemma 2.** If \( A_{i,j} \sim \mathcal{N}(0, 2/m) \) is the random Gaussian matrix and \( \phi(\cdot) \) is the ReLU function, then for fixed feature \( x \in \mathbb{R}^m \):
\[
\mathbb{P}_A(\|Ax\|_2 \in (1 \pm \epsilon)\|x\|_2) \geq 1 - \exp\{-\epsilon^2 m/100\}.
\]

First of all, for any \( f, f_1, f_2 \in \mathbb{F} \), it is easy to show that the random variable \( (\phi(w^\top f_1)\phi(w^\top f_2)) \) is sub-Exponential since \( (\phi(w^\top f)) \) is sub-Gaussian. Hence, with probability at least \( (1 - 2\exp\{-\Omega(m)\}) \), we have:
\[
|\phi(Af_1)^\top \phi(Af_2) - \mathbb{E}_w(\phi(w^\top f_1)\phi(w^\top x))| \leq \epsilon.
\]

Note that the function \( \sqrt{1 - z^2} + z(\pi - \arccos z) : \mathbb{R}^2 \to [0, 1] \) is bijective and maps the values from \([0, 1] \to [0, 1] \). Namely, with probability at least \( (1 - 2\exp\{-\Omega(m)\}) \):
\[
0 \leq \phi(Af_1)^\top \phi(Af_2) \leq 1 + \epsilon.
\]

Combine with Lemma 2, we have:
\[
1 - \epsilon \leq \|\phi(Af)\|_2 \leq 1 + \epsilon.
\]

Therefore, with probability \( (1 - 2\exp\{-\Omega(\epsilon^2 m)\}) \), \( \phi(Af), \forall f \in \mathbb{F} \) belongs to the banded area of a positive half axis semicircle in \( \mathbb{R}^2 \) with the width being \( \epsilon \). Then the \( \epsilon \)-covering number\(^2\) of \( \phi(Af) \) is:
\[
\dim(\phi(Af)) = \Theta\left(\frac{1}{\epsilon}\right), \quad f \in \mathbb{F}.
\]

\(^1\)Covering number can be seen as a finer measure of the dimensions of a compact set.

\(^2\)The smallest possible cardinality of an \( \epsilon \)-net of a given set.
In contrast, in the worst case, it is impossible to cover the set $\phi(Af)$ with only $\Theta(\frac{1}{\epsilon})$ $\epsilon$-balls and we have:

$$\dim(\text{conv}(\phi(Af))) = \Theta(\frac{1}{\epsilon^2}), \quad f \in F.$$ 

We now finish the proof. 

**Discussion** In general, the representation ability of a ball and a sphere is almost the same in high dimensional space. However, in low dimensional space, it is a completely different scene. This is also the reason that why some common methods which work well for image data are needed to be re-designed for 3D point cloud. It is more meaningful to expand the representation ability of the model in low dimensional space than to only increase the parameters of the model.

**3. Quantitative results on different resolutions of latent feature**

We additionally conduct the experiment on the different resolutions of feature fusion. In detail, using ResNet-50 [9] without conv5_x as the backbone, we choose to fuse the feature from conv2_x, conv3_x, and conv4_x. The fused feature is the concatenation of the resized feature from these three layers. The resolution of fused feature is an important parameter, which influences $F_{latent}$ and the $r$ in position feature $P$. From the Tab. 1, we find larger resolution of the fused feature is better for performance. It may be because segmenting ultra high-resolution images needs finer feature, corresponding to higher resolution.

<table>
<thead>
<tr>
<th>Res. of $F_{latent}$</th>
<th>IoU</th>
<th>mBA</th>
</tr>
</thead>
<tbody>
<tr>
<td>conv2_x</td>
<td>94.18</td>
<td>76.09</td>
</tr>
<tr>
<td>conv3_x</td>
<td>94.02</td>
<td>75.84</td>
</tr>
<tr>
<td>conv4_x</td>
<td>93.50</td>
<td>72.96</td>
</tr>
</tbody>
</table>

Table 1. The influence of different resolutions of latent feature. Res. denotes resolution.

<table>
<thead>
<tr>
<th>IoU/mBA</th>
<th>w/o CRM</th>
<th>w CRM</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaskFormer [4]</td>
<td>82.38/62.52</td>
<td>85.24/76.17</td>
</tr>
<tr>
<td>SegFormer [12]</td>
<td>74.87/56.63</td>
<td>80.25/74.52</td>
</tr>
</tbody>
</table>

Table 2. Refine transformer-based segmentation results.

**4. Results of pure model with single forward**

To verify performance of the pure model, we compare our method and CascadePSP [5] in single forward for different resolution inputs. In detail, CRM directly inferences on different resolution inputs, and CascadePSP [5] refines the inputs patch by patch. The inputs are from PSPNet [15] on the BIG dataset. Fig. 3 shows the performance changing with the image resolution through a single forward of model. CRM is not good at very low resolution. However, when the resolution increases, the performance of CRM is better.

**5. Refine transformer-based segmentation**

Since more and more transformer-based segmentation methods emerge, we also apply our CRM on their segmentation results. From the Tab. 2, we find CRM can also increase the performance of transformer-based methods on the BIG dataset.

In comparison, [4, 12] only release the pretrained weights on Cityscape [6] and ADE20K [16], but the BIG [5]’s annotation follows Pascal VOC’s guideline. We choose the union of ADE20K and Pascal VOC’s categories to evaluate.

**6. Additional visualization**

We also provide additional visualization results of CRM in Figs. 4, 5, 6 and 7. Masked areas are put on a green background for easy distinguishing. Although the images are resized, their original resolution is very large (2K~6K), where the details are more obvious. These visualizations are better viewed on the screen.
Figure 4. Visualization of the refinement on FCN [11]’s output. Better viewed on the screen.
Figure 5. Visualization of the refinement on DeepLabV3+ [2]'s output. Better viewed on the screen.
Figure 6. Visualization of the refinement on RefineNet [10]'s output. **Better viewed on the screen.**
Figure 7. Visualization of the refinement on PSPNet [15]’s output. Better viewed on the screen.
References