Supplementary Material
Human De-occlusion: Invisible Perception and Recovery for Humans

Anonymous CVPR submission

Paper ID 1550

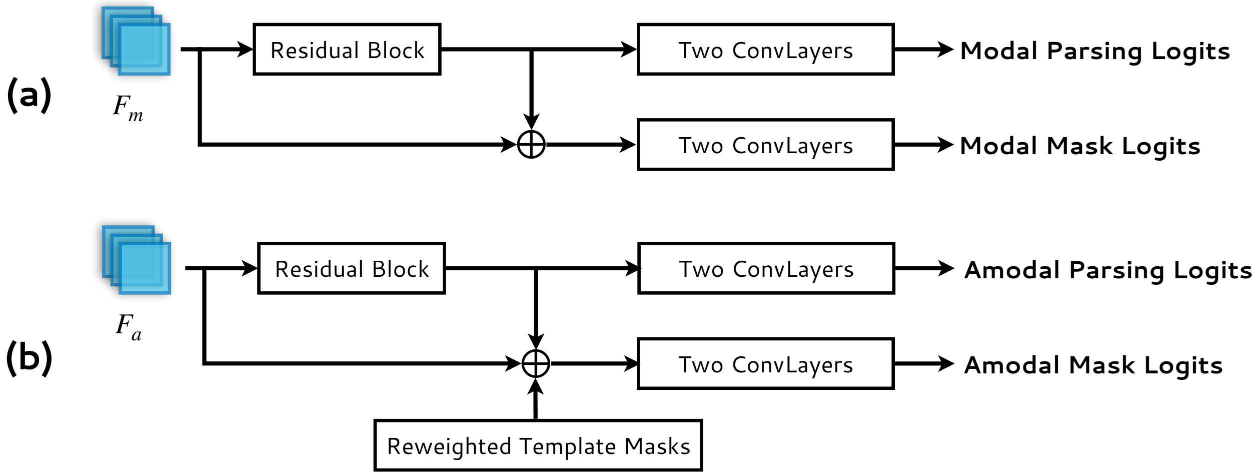

Figure 1. The detailed structures of the modal and amodal branches in the refined mask completion network. F_m and F_a denotes the output feature from the first hourglass module N_m^{hg} and N_a^{hg} respectively.

1. Network Details

We propose a two-stage framework to tackle the human de-occlusion task,which contains mask completion and content recovery. Here we provide more detailed structures of the two networks.

Refined Mask Completion Network We demonstrate the structures of the modal and amodal branches in Fig 1. The two branches share a similar pipeline. Take the modal branch for example, it uses a single residual block [1] and two convolution layers with a ReLU in the middle to extract model parsing logits. To generate the modal mask logits, we concatenate the feature of F_m and the auxiliary feature from the residual block as the input of another two convolution layers to get the final modal mask logits. To obtain the amodal mask logits, we additionally concatenate the reweighted template masks and adjust the channel number of the subsequent layer. We also measure the inference time of our network and the two branches cost extra 2.5ms compared to our baseline. (baseline 13.37ms *vs.* ours 15.83ms)

Parsing Guided Content Recovery Network The pseudo code of the PGA module is shown in Listing 1. In the first attention stream of body parts, we distribute input feature to different body parts and use Squeeze-and-Excitation [2] to extract useful part information. For the spatial attention, we add the one-hot parsing logits before obtaining key and query features, and a mutual spatial attention is applied to extract the relationship between the visible and the invisible regions. Because the differences of the key and the query mainly lie in the invisible points so that the network is able to find and recover them. The inference time increases by 5.82ms compared to the baseline. (baseline 8.02ms *vs.* ours 13.84ms) The total inference time costs about 21.65ms ($\sim$ 46 FPS) so that speed is not a problem in real applications.

2. More Result

We show more comparison results of mask completion and content recovery in Fig 2 and Fig 3 respectively. And some results on real occluded humans are shown in Fig 4. Black borders are padded to fit the input size to 256×256.

```
# x: input feature -> NxCxHxW
# m_oh / a_oh: modal / amodal parsing one-hot logits -> NxCpxHxW (Cp=19)
# ---------------------------------
# out: output feature -> NxCxHxW

def PGA(x, m_oh, a_oh)
    n, c, h, w = x.shape
    m_oh = interpolate(m_oh, (h, w)); a_oh = interpolate(m_oh, (h, w))

    # Attention.B
    m_x = conv1(x) * m_oh; a_x = conv2(x) * a_oh    # 3x3 kernel
    comp_x = concat([x, m_x, a_x], dim=1)
    out_x = conv3(comp_x)                           # 1x1 kernel
    out_a = SELayer(out_x)

    # Attention.T
    q_x = conv_q(concat([x, m_oh], dim=1)).view(n, -1, w*h).permute(0, 2, 1)
    k_x = conv_k(concat([x, a_oh], dim=1)).view(n, -1, w*h)
    v_x = conv_v(x).view(n, -1, w*h)
    energy = bmm(q_x, k_x)                          # batch matrix multiplication
    attention = softmax(energy, dim=-1)             # NxHWxHW
    attention_t = softmax(energy.permute(0, 2, 1), dim=-1)
    out_b = bmm(v_x, attention.permute(0, 2, 1)).view(n, c, h, w)
    out_c = bmm(v_x, attention_t.permute(0, 2, 1)).view(n, c, h, w)
    out = concat([out_a, out_b, out_c], dim=1)      # 1x1 kernel

    out = conv4(out)
    return out
```

Listing 1. The pseudo code of the proposed PGA module.

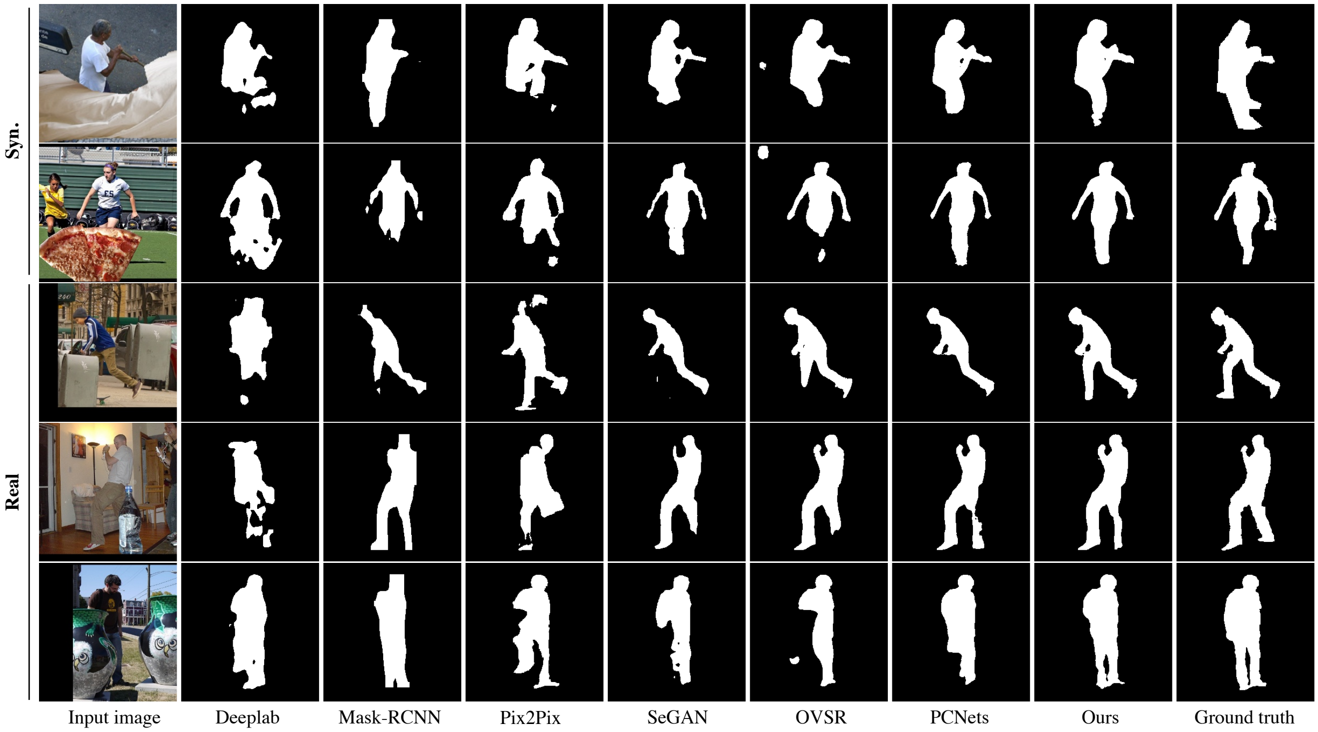

Figure 2. Some comparison results of mask completion results on synthesized and real images.

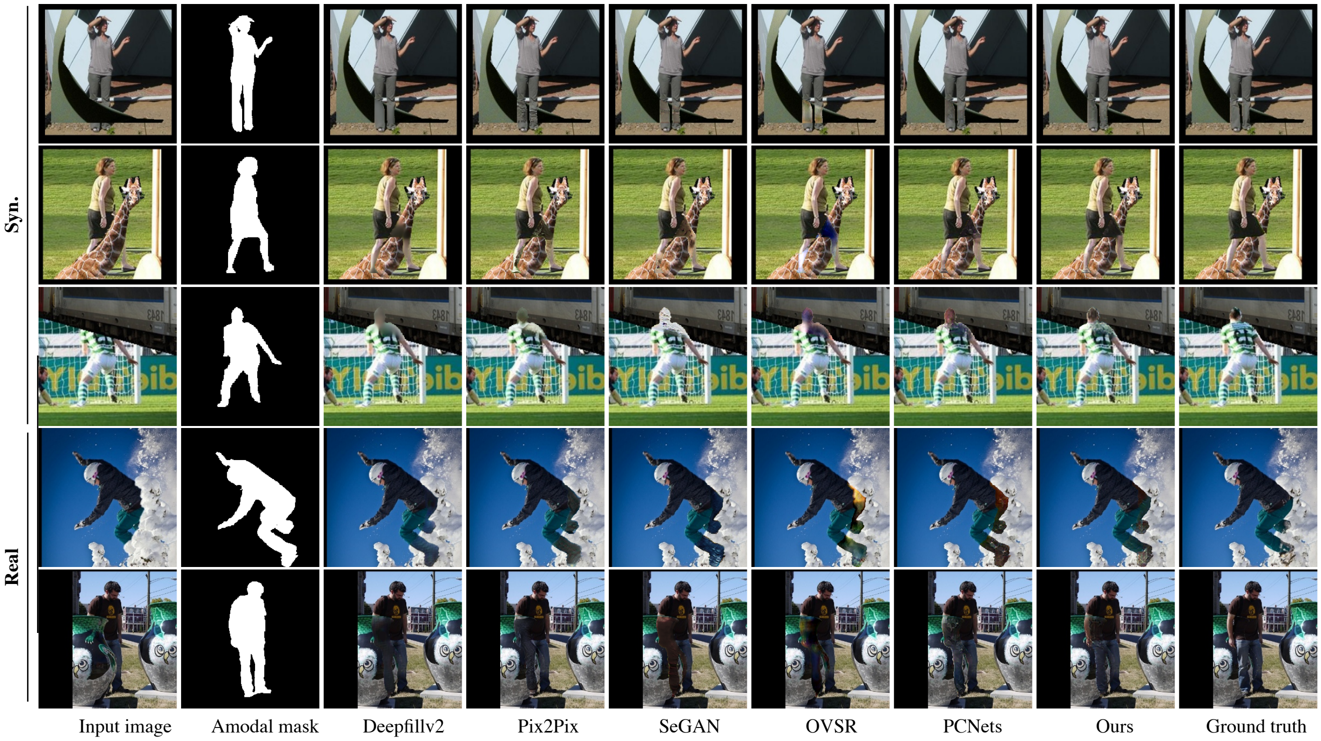

Figure 3. Some comparison results of content recovery results on synthesized and real images.

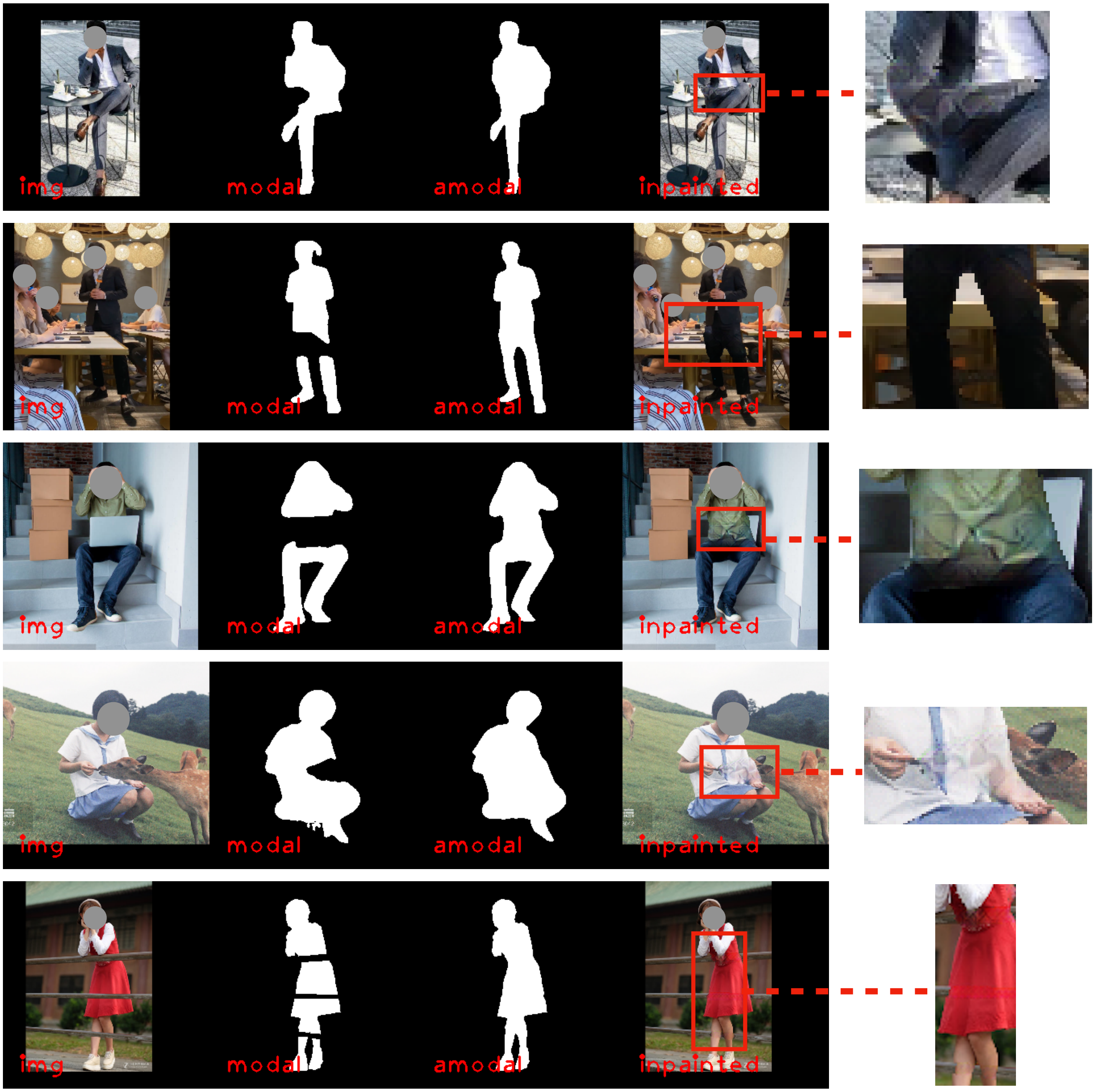

Figure 4. Some results in real human occlusion cases from Internet. We mosaic the faces due to privacy issues.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1

[2] 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. 1