MODNet: Real-Time Trimap-free Portrait Matting via Objective Decomposition

Supplementary Material
Paper 4946

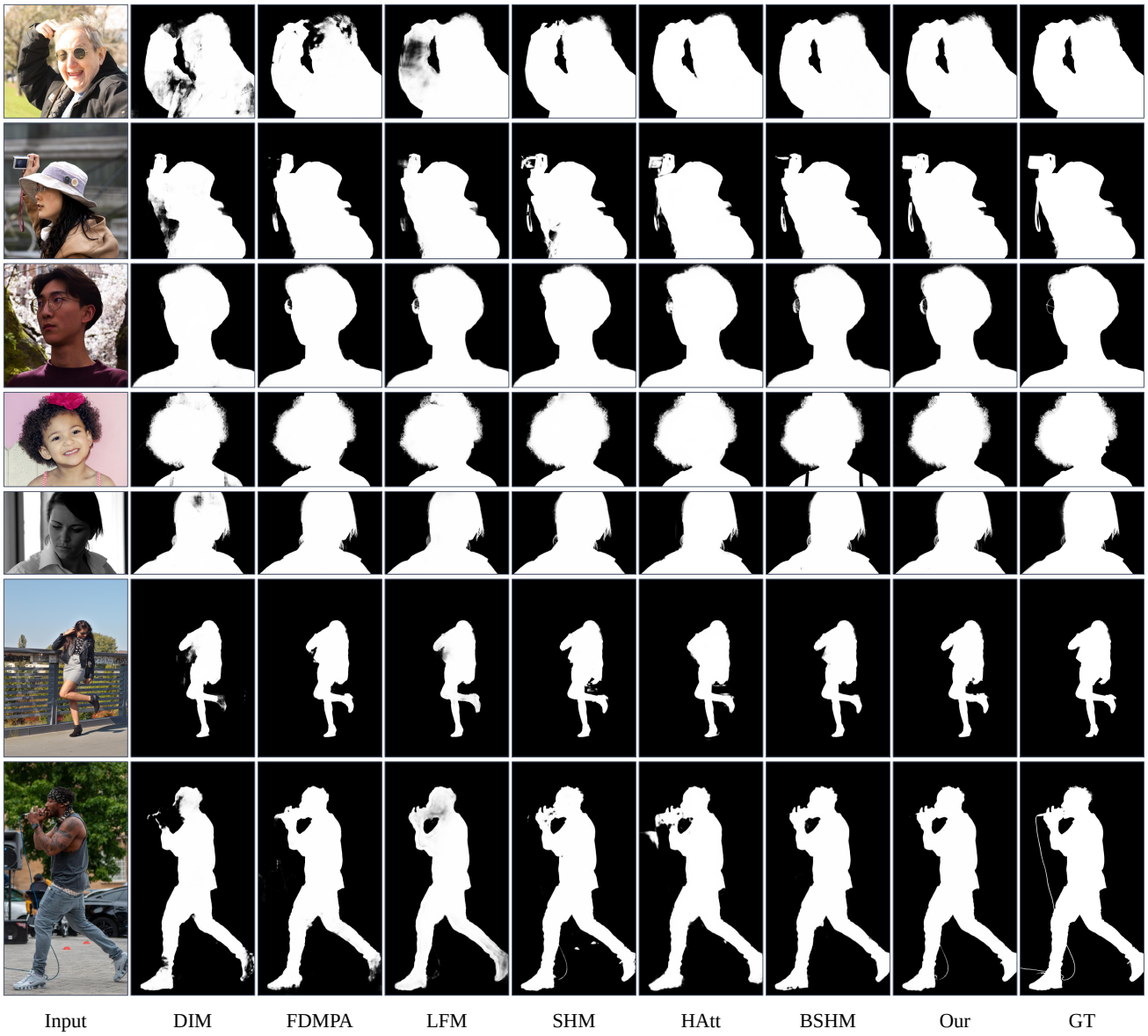

Figure 1: **More Visual Comparisons of Trimap-free Methods on PHM-100.** We compare our MODNet with DIM (Xu et al. 2017), FDMPA (Zhu et al. 2017), LFM (Zhang et al. 2019), SHM (Chen et al. 2018), HAtt (Qiao et al. 2020), and BSHM (Liu et al. 2020). Note that DIM here does not take trimaps as the input but is pre-trained on the SPS (supervise.ly 2018) dataset. Zoom in for the best visualization.

Appendix A: Analysis of e-ASPP

Here we compare the proposed Efficient ASPP (e-ASPP) with the standard ASPP in terms of the number of parameters and computational overhead. For a convolutional layer, the number of its parameters $\mathcal{P}$ can be calculated by:

$$\mathcal{P} = C_{out} \times C_{in} \times K \times K, \tag{1}$$

where C_{out} is the number of output channels, C_{in} is the number of input channels, and K is the kernel size. We can use *FLOPs* to measure the computational overhead $\mathcal{O}$ of a convolutional layer as:

$$\mathcal{O} = C_{in} \times 2 \times K \times K \times H_{out} \times W_{out} \times C_{out}, \tag{2}$$

where H_{out} and W_{out} are the height and the width of output feature maps, respectively.

Following, we represent the size of the input feature maps by (c, h, w), where c is the number of channels, h is the height of the input feature maps, and w is the width of the input feature maps. We represent the number of atrous convolutional layers (with a kernel size of k) in both ASPP and e-ASPP by m.

Standard ASPP (ASPP). In ASPP, (1) all atrous convolutional layers are independently applied to the input features maps to extract multi-scale features. These multi-scale features are then (2) concatenated and processed by a pointwise convolutional layer (with a kernel size of 1). We have:

$$\begin{aligned}\mathcal{P}_{ASPP} =& m \times (c \times c \times k \times k) \\ &+ c \times (m \times c) \times 1 \times 1 \\ =& m \times c^2 \times (k^2 + 1),\end{aligned} \tag{3}$$

$$\begin{aligned}\mathcal{O}_{ASPP} =& m \times (c \times 2 \times k \times k \times h \times w \times c) \\ &+ (m \times c) \times 2 \times 1 \times 1 \times h \times w \times c \\ =& ((2 \times k^2 + 2) \times m \times c) \times (h \times w \times c).\end{aligned} \tag{4}$$

Efficient ASPP (e-ASPP). As shown in Fig. 3 (in the paper), e-ASPP consists of four operations, including (1) Channel Reduction, (2) Multi-Scale Feature Extraction, (3) Multi-Scale Feature Fusion, and (4) Inter-Channel Feature Fusion. The total number of parameters and the total *FLOPs* are the sum of these four operations. We have:

$$\begin{aligned}\mathcal{P}_{e-ASPP} =& \frac{c}{4} \times c \times 1 \times 1 \\ &+ \frac{c}{4} \times m \times (1 \times 1 \times k \times k) \\ &+ \frac{c}{4} \times (1 \times m \times 1 \times 1) \\ &+ c \times \frac{c}{4} \times 1 \times 1 \\ =& \frac{2 \times c^2 + (k^2 + 1) \times m \times c}{4},\end{aligned} \tag{5}$$

Figure 2: **MODNet versus BM with a fixed camera position.** MODNet outperforms BM (Sengupta et al. 2020) when a car is entering the background (red region).

$$\begin{aligned}\mathcal{O}_{e-ASPP} =& c \times 2 \times 1 \times 1 \times h \times w \times \frac{c}{4} \\ &+ \frac{c}{4} \times m \times (1 \times 2 \times k \times k \times h \times w \times 1) \\ &+ \frac{c}{4} \times (m \times 2 \times 1 \times 1 \times h \times w \times 1) \\ &+ \frac{c}{4} \times 2 \times 1 \times 1 \times h \times w \times c \\ =& (c + \frac{(k^2 + 1) \times m}{2}) \times (h \times w \times c).\end{aligned} \tag{6}$$

Following the standard ASPP, we set $k = 3$ and $m = 5$. Usually, $c \geq 256$ is applied in most networks. Therefore, we have:

$$\frac{\mathcal{P}_{e-ASPP}}{\mathcal{P}_{ASPP}} \approx 0.01, \tag{7}$$

$$\frac{\mathcal{O}_{e-ASPP}}{\mathcal{O}_{ASPP}} \approx 0.01. \tag{8}$$

It means that compared to the standard ASPP, our proposed e-ASPP has only 1% of the parameters and 1% of the computational overhead. In MODNet, our experiments show that e-ASPP can achieve performance comparable to ASPP. Note that when the Channel Reduction operation in e-ASPP is disabled, e-ASPP still has only 2% of the parameters and 2% of the computational overhead compared to ASPP.

Appendix B: More Results on PHM-100

Fig. 1 provides more visual comparisons of MODNet and the existing trimap-free methods on PHM-100.

Appendix C: Comparison with BM

We compare MODNet against the background matting (BM) proposed by (Sengupta et al. 2020). Since BM does not support dynamic backgrounds, we conduct validations in the fixed-camera scenes from (Sengupta et al. 2020). BM relies on a static background image, which implicitly assumes that all pixels whose value changes across frames belong to the foreground. As shown in Fig. 2, when a moving object suddenly appears in the background, the result of BM will be affected, but MODNet is robust to such disturbances.

References

Chen, Q.; Ge, T.; Xu, Y.; Zhang, Z.; Yang, X.; and Gai, K. 2018. Semantic human matting. In *ACMMM*.

Liu, J.; Yao, Y.; Hou, W.; Cui, M.; Xie, X.; Zhang, C.; and Hua, X.-S. 2020. Boosting Semantic Human Matting With Coarse Annotations. In *CVPR*.

Qiao, Y.; Liu, Y.; Yang, X.; Zhou, D.; Xu, M.; Zhang, Q.; and Wei1, X. 2020. Attention-Guided Hierarchical Structure Aggregation for Image Matting. In *CVPR*.

Sengupta, S.; Jayaram, V.; Curless, B.; Seitz, S.; and Kemelmacher-Shlizerman, I. 2020. Background Matting: The World is Your Green Screen. In *CVPR*.

supervise.ly. 2018. Supervisely Person Dataset. *supervise.ly*.

Xu, N.; Price, B.; Cohen, S.; and Huang, T. 2017. Deep Image Matting. In *CVPR*.

Zhang, Y.; Gong, L.; Fan, L.; Ren, P.; Huang, Q.; Bao, H.; and Xu, W. 2019. A late fusion cnn for digital matting. In *CVPR*.

Zhu, B.; Chen, Y.; Wang, J.; Liu, S.; Zhang, B.; and Tang, M. 2017. Fast Deep Matting for Portrait Animation on Mobile Phone. In *ACMMM*.