Supplementary Materials for Segmentation-guided Layer-wise Image Vectorization with Gradient Fills

Hengyu Zhou, Hui Zhang✉, and Bin Wang✉

School of Software, Tsinghua University, P. R. China
zhouhy22@mails.tsinghua.edu.cn, {huizhang, wangbins}@tsinghua.edu.cn

1 Qualitative Comparisons

Some qualitative comparisons are presented in this section. More comparisions can be found at the end of this document.

1.1 Noto Emoji

We evaluated on this collection of 256 icons randomly selected from Noto Emoji [1]. Our selection is out of a more recent version than LIVE [2], as shown in Fig. 1, where every two rows are the same emojis from different versions. Our method performs equally well or better with the old design and shows a significant improvement with the new design.

1.2 Fluent Emoji

Most icons from Fluent Emoji [3] come with gradients. Fig. 2 shows that our method keeps more details when a small number of paths are used.

1.3 Iconfont

Illustrations in this dataset enjoy great details achieved via large numbers of paths. When few paths are added, our method creates large Bézigons with gradients, for example, the Sun in Fig. 3a, despite that they are made up of many small solid-filled paths. Participants in the user study show preference for our results.

2 Details of the User Study

We designed a questionnaire consisting of 2,560 questions, from which 20 were selected for each participant. Each question is presented with three images: one

✉ Corresponding authors.

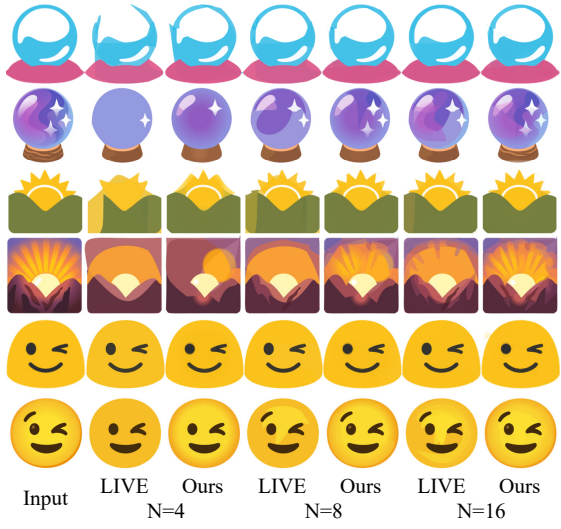


Fig. 1: Comparisons on Noto Emoji. Odd rows are from the design released with Android 7. Even rows are from the current design since Android 11. Input images are resized to 256×256 for vectorization.

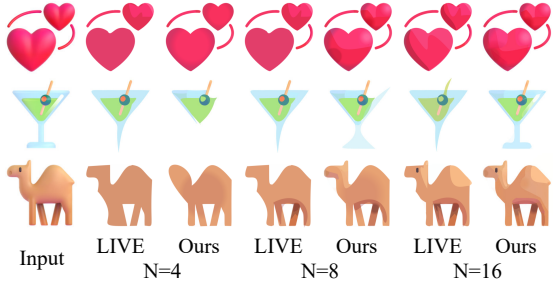
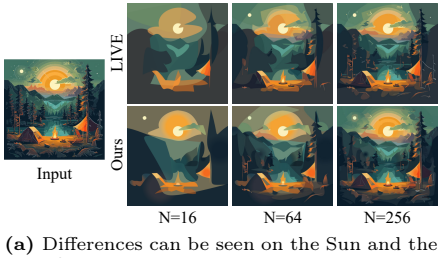


Fig. 2: Comparisons on Fluent Emoji. Our approach preserves the highlights better. Input images are resized to 256×256 for vectorization.



(a) Differences can be seen on the Sun and the campfire, where our method uses radial gradients to approximate the colors.

(b) More comparisons. Our results are generally more colorful and closer to the input, especially when the number of paths is small.

Fig. 3: Comparisons on Iconfont. Input images are resized to 512×512 for vectorization.

is the raster input, and the other two are the vectorized results by LIVE and by our method.

For each of the 640 images from the three datasets (256 from Noto Emoji, 256 from Fluent Emoji, and 128 from Iconfont), 4 numbers of paths are chosen. For images in Noto Emoji and Fluent Emoji, N = 8, 16, 32, 64 are used. For Iconfont, N = 32, 64, 128, 256 are used. These numbers are chosen as we notice that with fewer paths both methods yield unsatisfying results, from which a better one is hard to choose; and with more paths, outputs are almost undistinguishable. Outputs with unused numbers of paths can be found at the end of this document.

References

1. Noto emoji, https://github.com/googlefonts/noto-emoji Accessed: 2023-09-19 1
2. Ma, X., Zhou, Y., Xu, X., Sun, B., Filev, V., Orlov, N., Fu, Y., Shi, H.: Towards layer-wise image vectorization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16314–16323 (2022) 1
3. Fluent emoji, https://github.com/microsoft/fluentui-emoji Accessed: 2023-10-27 1

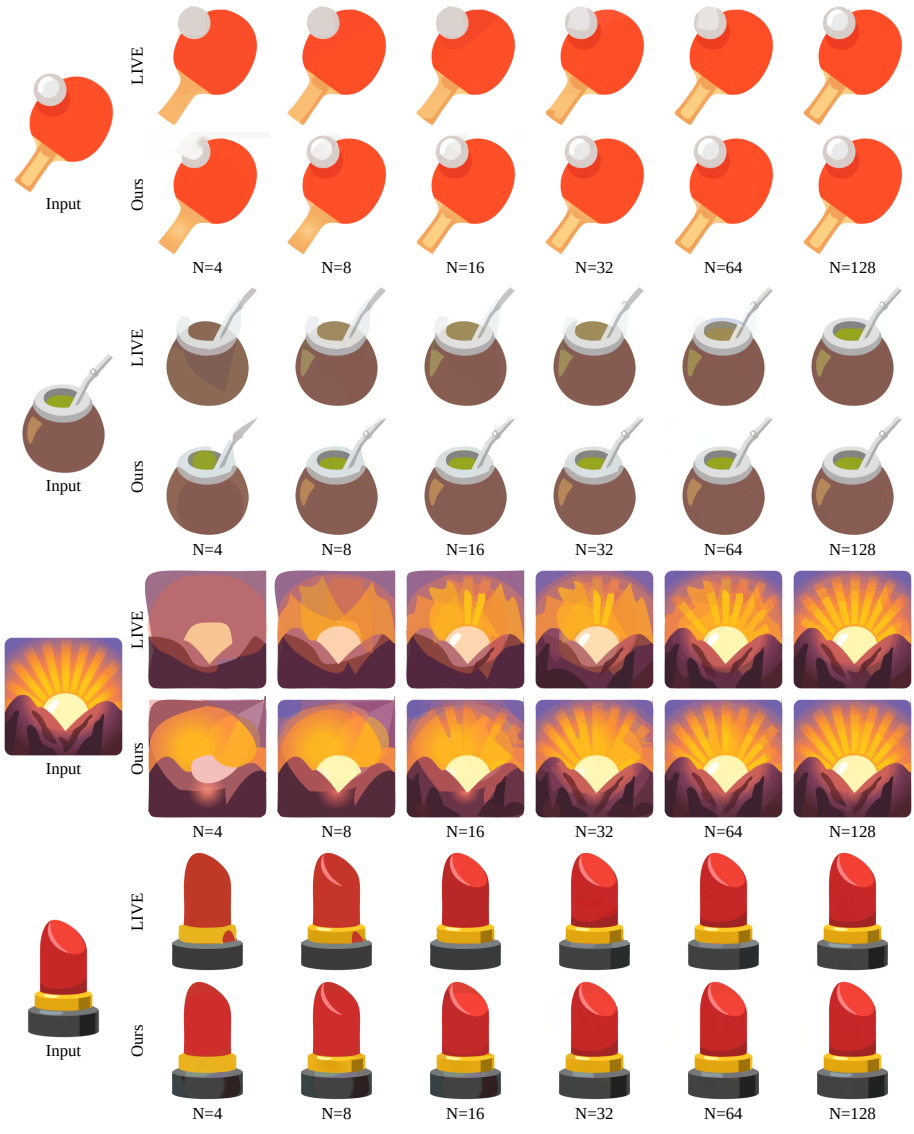


Fig. 4: Comparisons on Noto Emoji

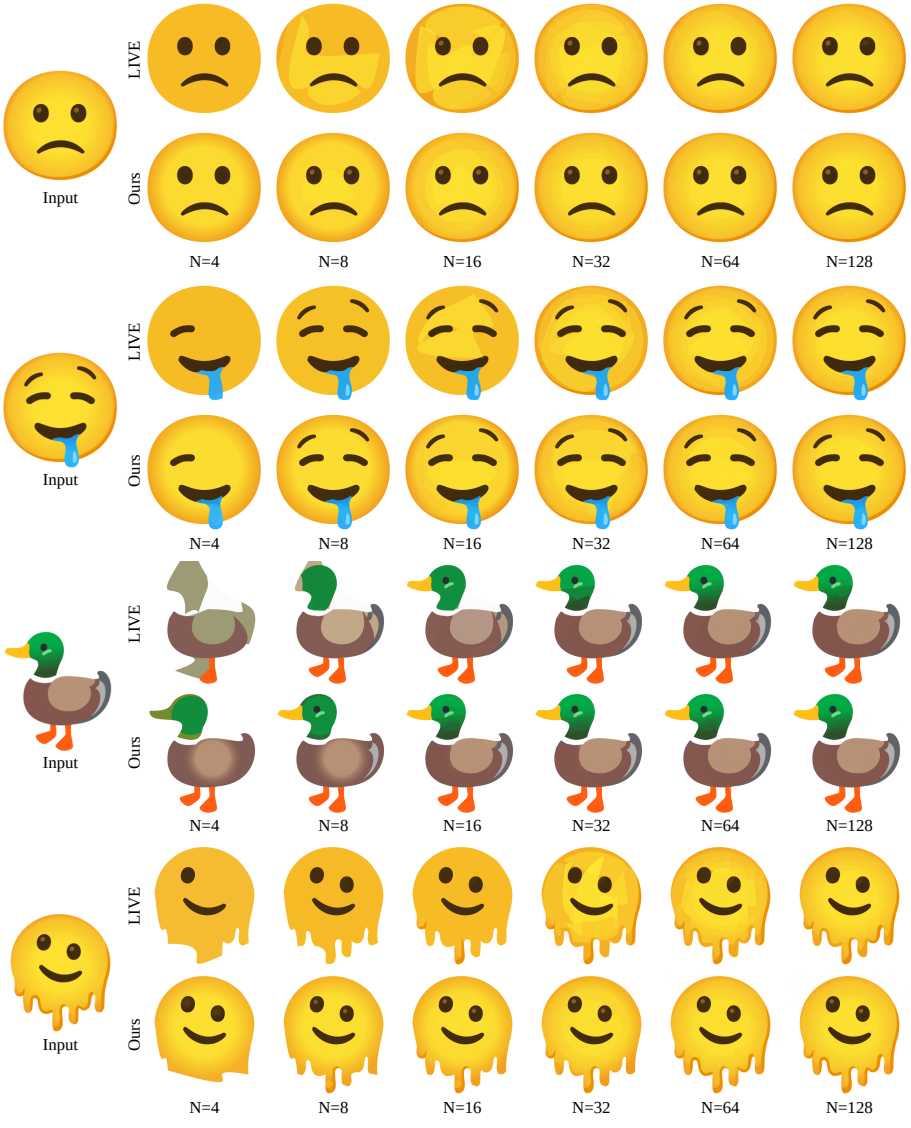


Fig. 5: More comparisons on Noto Emoji

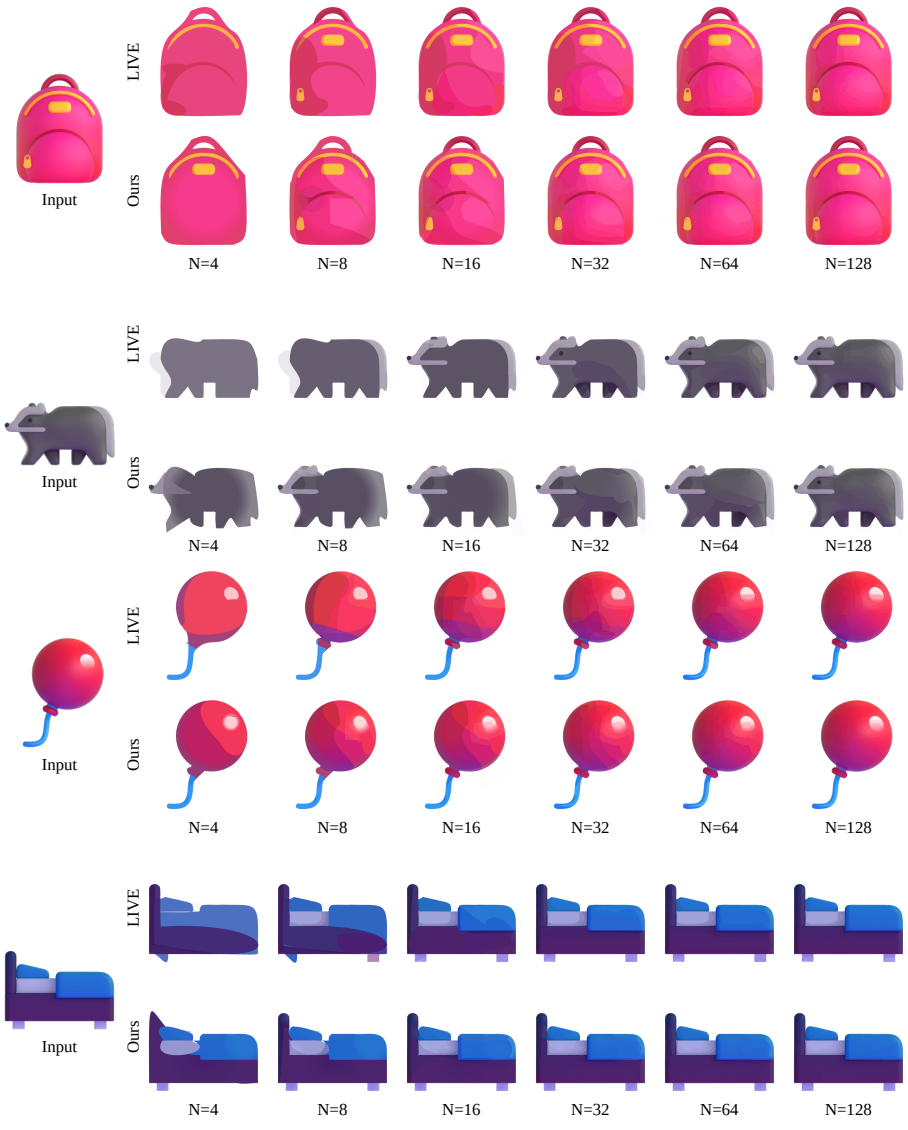


Fig. 6: Comparisons on Fluent Emoji

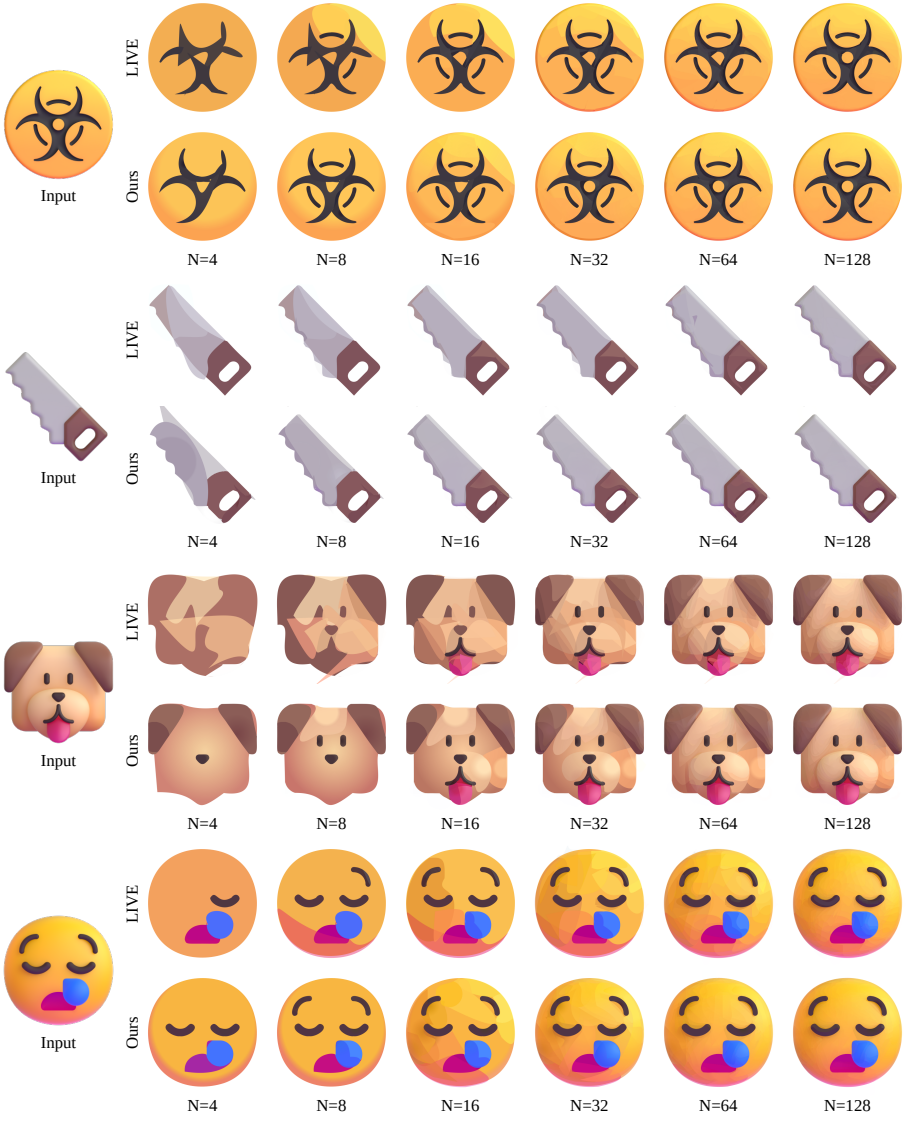


Fig. 7: More comparisons on Fluent Emoji

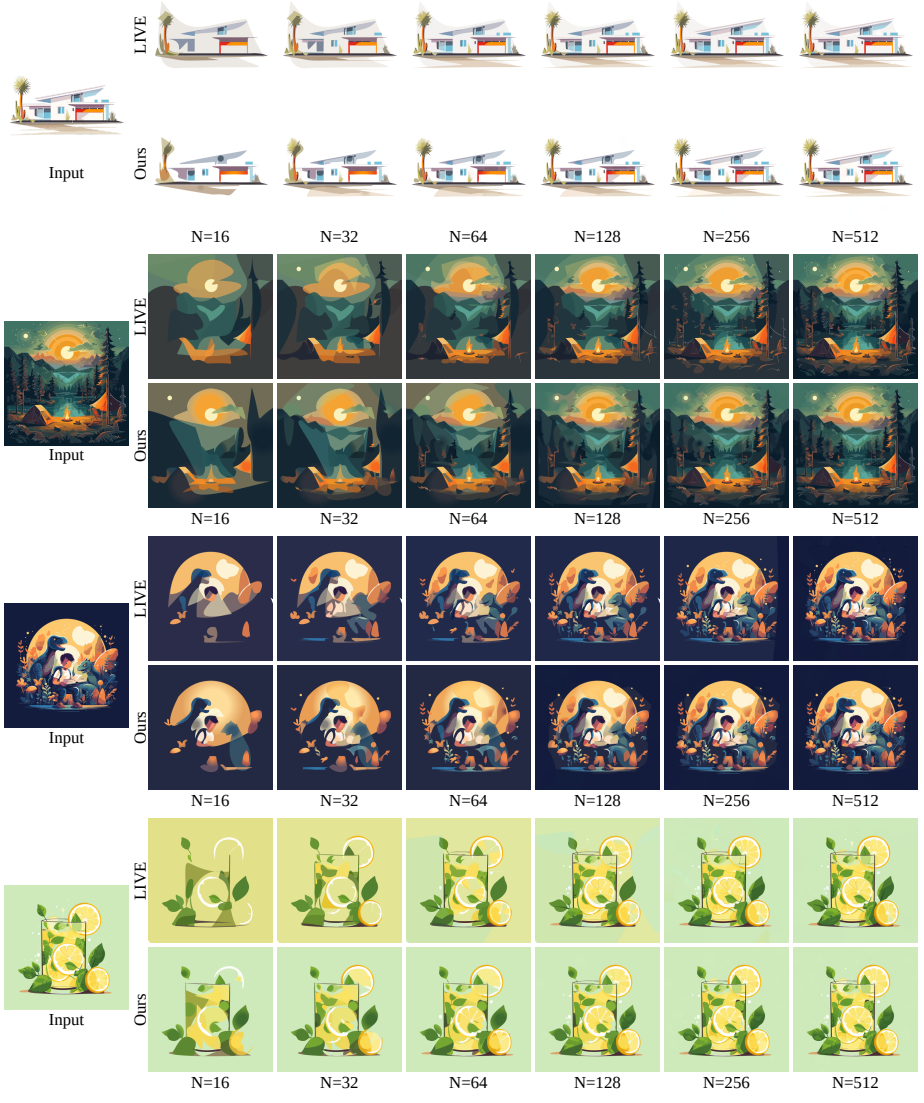


Fig. 8: Comparisons on Iconfont

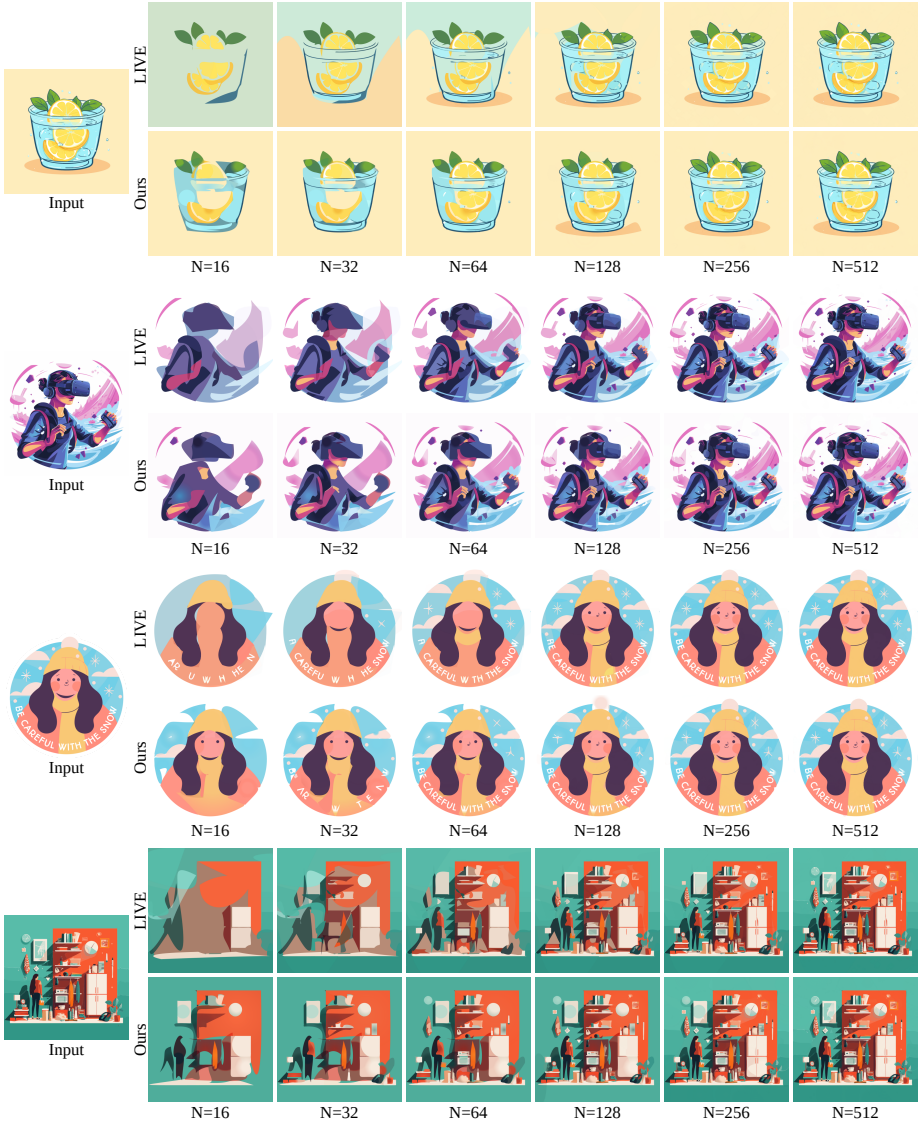


Fig. 9: More comparisons on Iconfont