Stereoscopic Neural Style Transfer

Anonymous CVPR submission

Paper ID 1392

Figure 1. An example simple VR glass to visualize the 3D stylization effects.

1. Overview

Our supplementary material consists of three parts:

- One video to show our main method and some visualization comparison results.
- One folder containing 3D results for reviewers who have simple virtual reality glasses like Figure 1 (You can send these results to mobile phone or some other display devices, then use the VR glasses to visualize the 3D effects).
- One pdf (this one) to describe some remaining details which are not given in the paper.

2. Details about *DispOccNet*

Network structure The detailed network structure of *DispOccNet* is shown in Table 1. Note that *convN, convNa,ConvNb, upconvN* are followed by a *LeakyReLU* layer, whose negative slope value is 0.1. *occN* is followed by a *Sigmoid* layer.

When integrating *DispOccNet* and *StyleNet*, only the final bidirectional disparity maps *disp1* and occlusion masks *occ1* are used, and then bilinearly resized to the same resolution of the feature map of the encoder of *StyleNet*.

Some visualization results. In Figure 2, we show some predicted bidirectional disparity maps and occlusion masks of *DispOccNet*. Compared to *DispNet* [1], which can only generate the single directional disparity, our *DispOccNet* can obtain bidirectional disparity maps and occlusion masks with a single feed-forward pass. Our predicted disparity maps have comparable or even slightly better quality in non-occluded regions. In the last two rows, we compare our predicted occlusions with that generated by post consistency check, which contains more boundary false alarms and noises.

References

[1] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016. 1, 2, 3

Name	Kernel	Str.	Ch I/O	InpRes	OutRes	Input
conv1	7×7	2	6/64	768×384	384×192	Images
conv2	5×5	2	64/128	384×192	192×96	conv1
conv3a	5×5	2	128/256	192×96	96×48	conv2
conv3b	3×3	1	256/256	96×48	96×48	conv3a
conv4a	3×3	2	256/512	96×48	48×24	conv3b
conv4b	3×3	1	512/512	48×24	48×24	conv4a
conv5a	3×3	2	512/512	48×24	24×12	conv4b
conv5b	3×3	1	512/512	24×12	24×12	conv5a
conv6a	3×3	2	512/1024	24×12	12×6	conv5b
conv6b	3×3	1	1024/1024	12×6	12×6	conv6a
disp6+disp_loss6	3×3	2	1024/2	12×6	12×6	conv6b
occ6+occ_loss6	3×3	2	1024/2	12×6	12×6	conv6b
upconv5	4×4	2	1024/512	12×6	24×12	conv6b
updisp6	4×4	2	2/2	12×6	24×12	disp6
upocc6	4×4	2	2/2	12×6	24×12	occ6
iconv5	3×3	1	1028/512	24×12	24×12	upconv5+updisp6+upocc6+conv5b
disp5+disp_loss5	3×3	1	512/2	24×12	24×12	iconv5
occ5+occ_loss5	3×3	1	512/2	24×12	24×12	iconv5
upconv4	4×4	2	512/256	24×12	48×24	iconv5
updisp5	4×4	2	2/2	24×12	48×24	disp5
upocc5	4×4	2	2/2	24×12	48×24	occ5
iconv4	3×3	1	772/256	48×24	48×24	upconv4+updisp5+upocc5+conv4b
disp4+disp_loss4	3×3	1	256/2	48×24	48×24	iconv4
occ4+occ_loss4	3×3	1	256/2	48×24	48×24	iconv4
upconv3	4×4	2	256/128	48×24	96×48	iconv4
updisp4	4×4	2	2/2	48×24	96×48	disp4
upocc4	4×4	2	2/2	48×24	96×48	occ4
iconv3	3×3	1	388/128	96×48	96×48	upconv3+updisp4+upocc4+conv3b
disp3+disp_loss3	3×3	1	128/2	96×48	96×48	iconv3
occ3+occ_loss3	3×3	1	128/2	96×48	96×48	iconv3
upconv2	4×4	2	128/64	96×48	192×96	iconv3
updisp3	4×4	2	2/2	96×48	192×96	disp3
upocc3	4×4	2	2/2	96×48	192×96	occ3
iconv2	3×3	1	196/64	192×96	192×96	upconv2+updisp3+upocc3+conv2
disp2+disp_loss2	3×3	1	64/2	192×96	192×96	iconv2
occ2+occ_loss2	3×3	1	64/2	192×96	192×96	iconv2
upconv1	4×4	2	64/32	192×96	384×192	iconv2
updisp2	4×4	2	2/2	192×96	384×192	disp2
upocc2	4×4	2	2/2	192×96	384×192	occ2
iconv1	3×3	1	100/32	384×192	384×192	upconv1+updisp2+upocc2+conv1
disp1+disp_loss1	3×3	1	32/2	384×192	384×192	iconv1
occ1+occ_loss1	3×3	1	32/2	384×192	384×192	iconv1

Table 1. The detailed network structure of *DispOccNet*, which follows the basic architecture of [1]. The contracting part consists of convolutions *conv1* to *conv6b*. In the expanding part, upconvolutions (*upconvN*,*updispN*,*upoccN*), convolutions (*iconvN*, *dispN*, *occN*) and loss layers are alternating. Features from earlier layers are concatenated with higher layer features, then are fed into *iconvN*. The two channels of *dispN* represent the bidirectional disparity (left and right) respectively, while *occN* denotes the corresponding bidirectional occlusion masks. *disp1* and *occ1* are the final predicted bidirectional disparity maps and occlusion masks.

CVPR
#1392

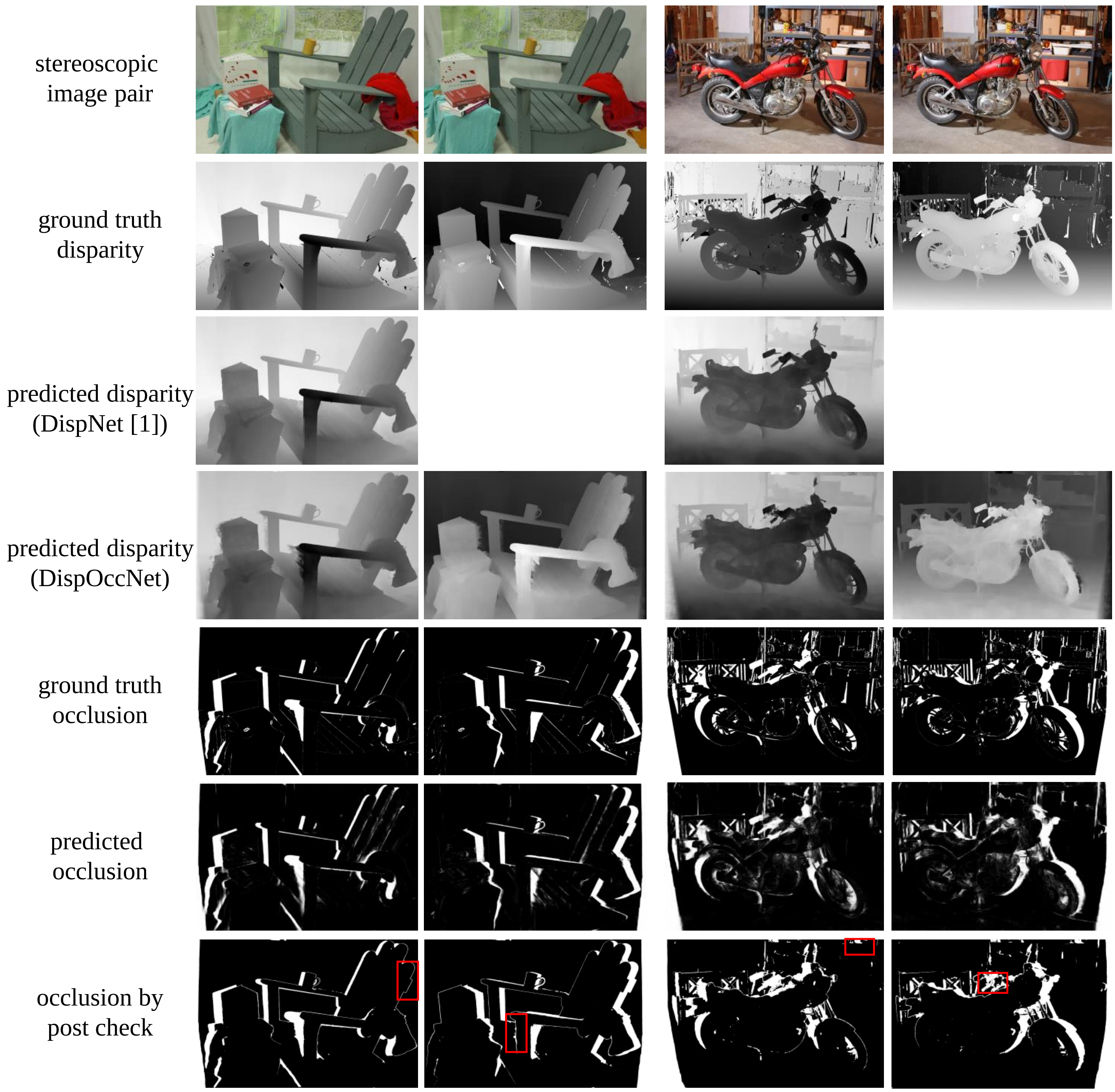

Figure 2. Some example results. Compared to *DispNet*[1], which can only generate the single directional disparity, our *DispOccNet* can obtain bidirectional disparity maps and occlusion masks with a single feed-forward pass. Our predicted disparity maps have comparable or even slightly better quality in non-occluded regions. Compared to our predicted occlusion masks, the occlusion masks generated by post consistency check contain more boundary false alarms and noises. Note that we only care about the disparity in non-occluded regions in *DispOccNet*, so the disparity map in occluded regions is not smooth as the *DispNet*.