

000
001
002054
055
056

Supplemental Material for EDIT: Exemplar-Domain Aware Image-to-Image Translation

057
058
059003
004
005
006
007060
061
062
063

Anonymous CVPR submission

008
009
010
011
012064
065
066
067
068

Paper ID 351

013
014069
070
071

1. Network Architecture

015
016
017
018
019072
073
074
075
076

Parameter Network. Our parameter network includes the convolution layers of *VGG16* followed by one fully connected layer and one group fully connected layer. The detailed architecture is shown in Table 1.

020
021
022
023
024
025
026
027
028
029
030
031
032
033
034077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092

Generator. The generator has an encoder-decoder architecture. We notice that, the more parameters needed to be generated using the parameter network, the larger model size is required for the parameter network. As shown in Table 2, our generator begins with three down-convolution layers to extract and encode the features of input images. Afterwards, 9 residual blocks are used to process the style-related features according to exemplars.

035
036
037
038093
094
095
096

2. More Results

039
040
041
042
043
044
045
046
047
048
049
050
051
052
053097
098
099
100
101
102
103
104
105
106
107

We provide more comparison results for painting \leftrightarrow photo, edge \leftrightarrow shoe/handbag from Figures 1 to 8. Overall, our EDIT outperforms the other state-of-the-art methods including NST [1], metaNST [6], WCT [4], cycleGAN [8], MUNIT [2], DRIT [3] and EGSC-IT [5], which is able to capture both the domain and the exemplar style information and preserve the content. Moreover, art2real [7] is specifically proposed to translate a painting to a photo. Although achieving good visual effect, this method is still lacking of the ability to control the style of generated results and suffer from abrupt changes across different semantic regions. The comparison results are shown in Figure 9. In addition, we further provide more results by our EDIT in Figures 10-14, which are semantically meaningful, structurally reasonable and visually striking.

References

- [1] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *CVPR*, pages 2414–2423, 2016. 1
- [2] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, pages 172–189, 2018. 1
- [3] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Diverse image-to-image translation via disentangled representations. In *ECCV*, pages 35–51, 2018. 1
- [4] Y. Li, C. Fang, J. Yang, Z. Wang, X. Lu, and M.-H. Yang. Universal style transfer via feature transforms. In *NeurIPS*, pages 385–395, 2017. 1
- [5] L. Ma, J. Xu, S. Georgoulis, T. Tuytelaars, and L. Van Gool. Exemplar guided unsupervised image-to-image translation with semantic consistency. In *ICLR*, 2019. 1
- [6] F. Shen, S. Yan, and G. Zeng. Neural style transfer via meta networks. In *CVPR*, pages 8061–8069, 2018. 1
- [7] Matteo Tomei, Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. Art2real: Unfolding the reality of artworks via a semantically-aware image-to-image translation. In *CVPR*, pages 5849–5859, 2019. 1
- [8] J. Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, pages 2223–2232, 2017. 1

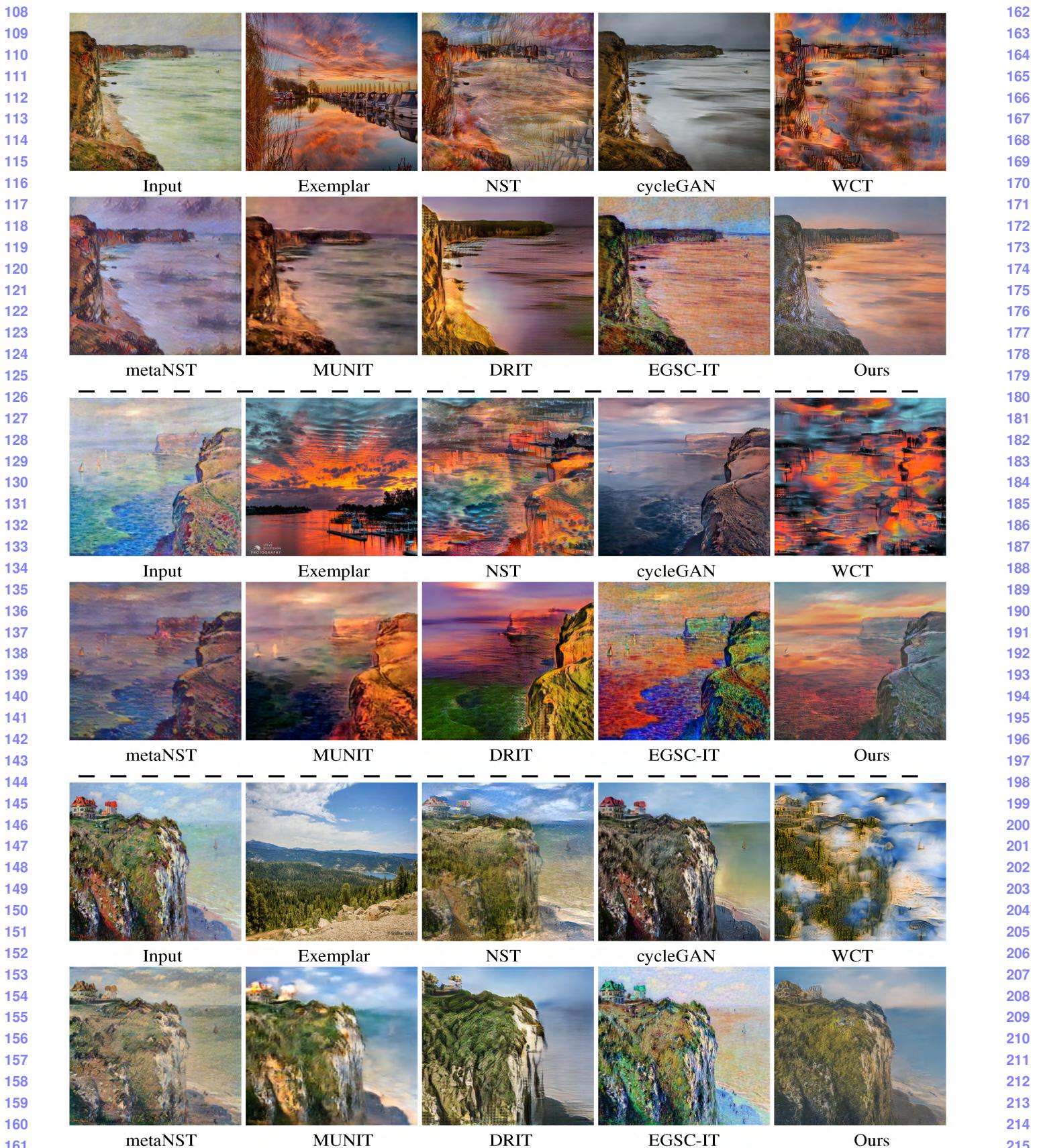


Figure 1. Visual comparisons on painting → photo.



Figure 2. Visual comparisons on painting → photo.

324 Table 1. The parameter network architecture. The c , k , s and p in the top part stand for the channel number, the kernel size, the stride step
 325 and the padding size, respectively. The N_{neuron} , N_{group} , N_{para} and M_{para}^l in the bottom part are the number of neurons in a layer of
 326 each group, the number of group of a group fully-connected layer ($N_{group}=1$ if is not a group fully-connected layer), the number of layers
 327 in the generator whose parameters need to be generated and the number of parameters in the l -th layer where $1 \leq l \leq N_{para}$, respectively.
 328

Convolution Layers						
Output Size	Operator	c	k	s	p	
256×256	Conv2d-ReLU	64	3	1	1	378
256×256	Conv2d-ReLU (relu1_2)	64	3	1	1	379
128×128	MaxPool2d	-	-	-	-	380
128×128	Conv2d-ReLU	128	3	1	1	381
128×128	Conv2d-ReLU (relu2_2)	128	3	1	1	382
64×64	MaxPool2d	-	-	-	-	383
64×64	Conv2d-ReLU	256	3	1	1	384
64×64	Conv2d-ReLU	256	3	1	1	385
64×64	Conv2d-ReLU (relu3_3)	256	3	1	1	386
32×32	MaxPool2d	-	-	-	-	387
32×32	Conv2d-ReLU	512	3	1	1	388
32×32	Conv2d-ReLU	512	3	1	1	389
32×32	Conv2d-ReLU (relu4_3)	512	3	1	1	390
16×16	MaxPool2d	-	-	-	-	391
16×16	Conv2d-ReLU (relu5_1)	512	3	1	1	392

Intermediate Processing		
Input	Operation	Output Size
relu1_2,relu2_2, relu3_3,relu4_3 relu5_3	calculate mean and standard deviation	1×2944

Fully-Connected Layers		
layer	N_{neuron}	N_{group}
1	$128 \times N_{para}$	1
2	M_{para}^l	N_{para}

355 Table 2. The generator architecture. The c , k and s in the top part is channel number, kernel size and stride step respectively. Column
 356 **Trainable** indicates whether the layer is jointly trained with the parameter network (**Trainable=True**) or generated by the parameter
 357 network (**Trainable=False**).
 358

Name	Operator	Trainable	Repeat	c	k	s
down-conv	Conv2d-IN-ReLU	True	1	32	9	2
	Conv2d-IN-ReLU	True		64	3	2
	Conv2d-IN-ReLU	True		128	3	2
residual block	Conv2d-IN-ReLU	True	9	128	3	1
	Conv2d-IN-ReLU	False		128	3	1
up-conv	Upsample-Conv2d-IN-ReLU	True	1	128	3	1
	Upsample-Conv2d-IN-ReLU	False		64	3	1
	Conv2d-Tanh	True		32	9	1

369 Table 3. The discriminator architecture. The c , k , s and p is channel number, kernel size, stride step and padding size respectively.
 370

Output Size	Operator	c	k	s	p
128×128	Conv2d-LeakyReLU	64	9	2	1
64×64	Conv2d-IN-LeakyReLU	128	3	2	1
32×32	Conv2d-IN-LeakyReLU	256	3	2	1
16×16	Conv2d-IN-LeakyReLU	512	3	2	1
17×17	ZeroPad2d	-	-	-	-
16×16	Conv2d	1	4	1	1



Figure 3. Visual comparisons on painting → photo.

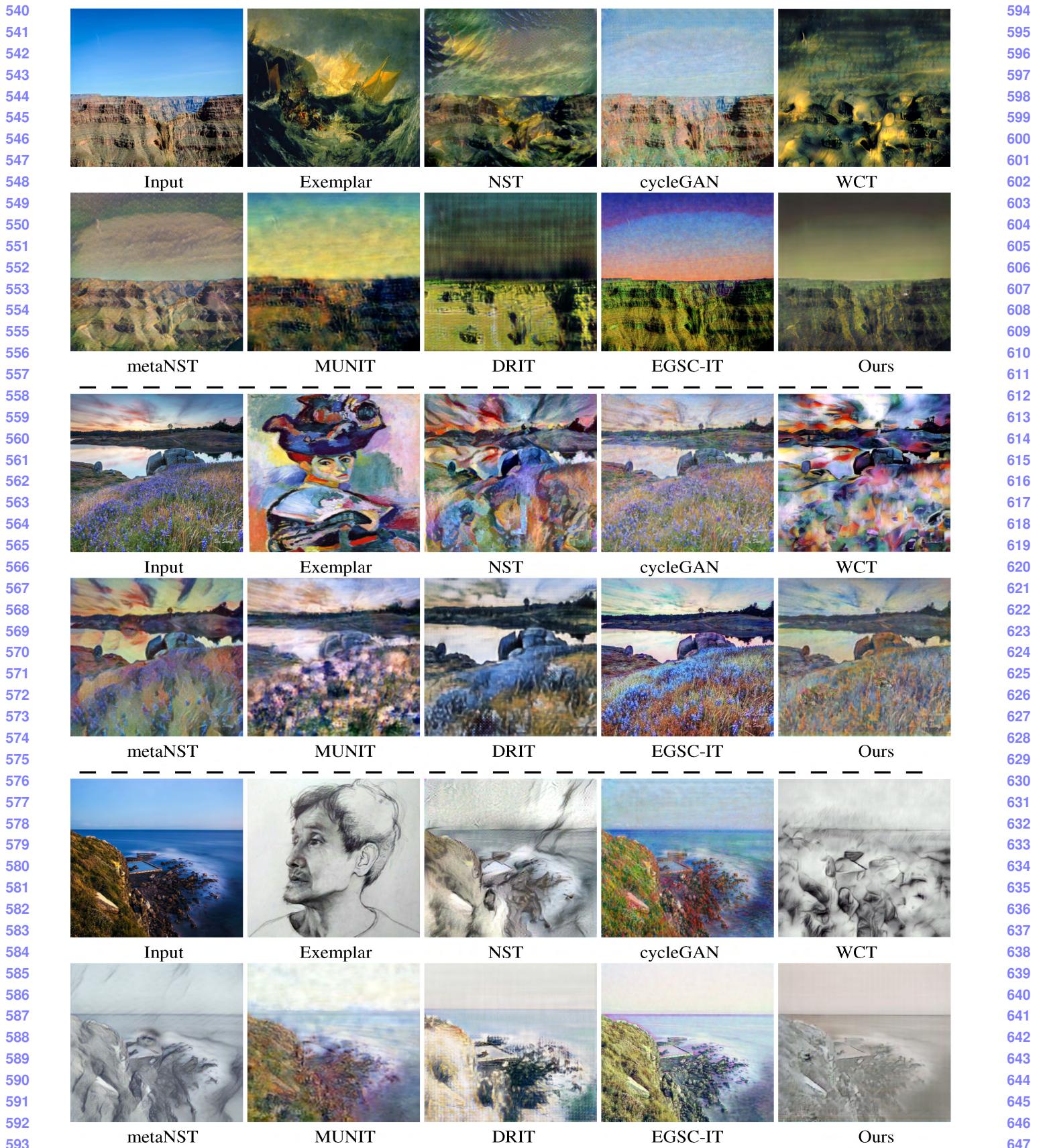


Figure 4. Visual comparisons on photo → Monet's painting



Figure 5. Visual comparisons on photo → Monet's painting

Figure 6. Visual comparisons on edge → shoe.



Figure 6. Visual comparisons on edge → shoe.

Figure 7. Visual comparisons on edge → handbag



864									918
865									919
866									920
867									921
868									922
869									923
870	Input	Exemplar	NST	WCT	cycleGAN	metaNST	DRIT	EGSC-IT	Ours
871									
872									
873									
874									
875									
876									
877									
878									
879									
880									
881									
882									
883									
884									
885									
886									
887									
888									
889									
890									
891									
892									
893									
894									
895									
896									
897									
898									
899									
900									
901									
902									
903									
904									
905									
906									
907									
908									
909									
910									
911									965
912									966
913									967
914									968
915									969
916									970
917									971

Figure 8. Visual comparisons on shoe → edge.

972
973
974
975
976
977
978
979990
991
992
993
994
995
996
997
998
9991000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
10151016
1017
1018
1019
1020
1021
1022
1023
1024
1025

Input

Figure 9. The visual comparison between art2real and our EDIT.

art2real

Ours

1070
1071
1072
1073
1074
1075
1076
1077
1078
1079

1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133



Cezanne→photo

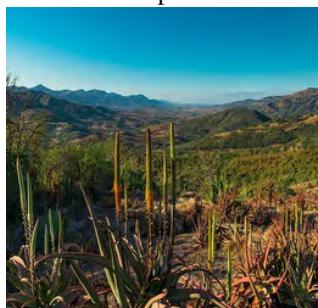
Monet→photo

Figure 10. Visual results by EDIT on painting → photo.

1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187

1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241

Input



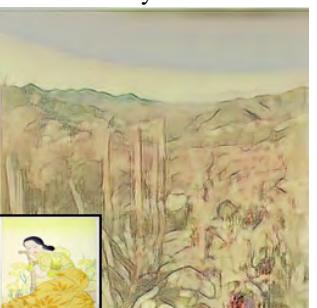
Monet



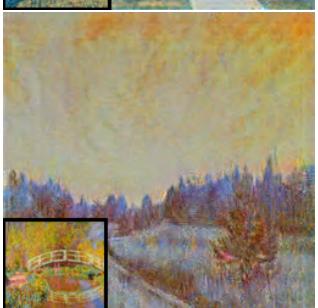
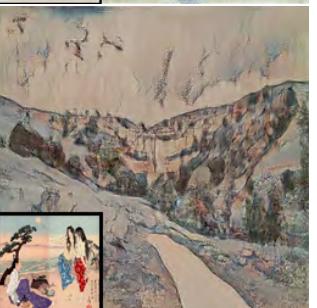
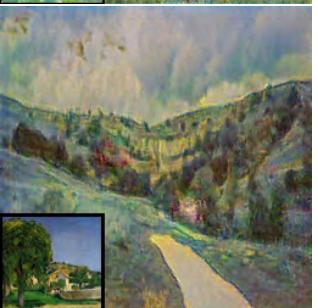
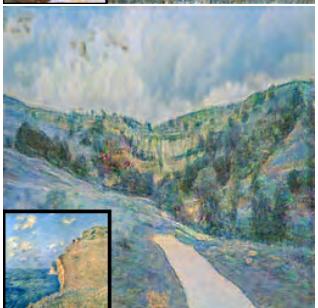
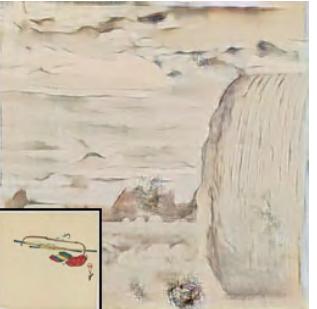
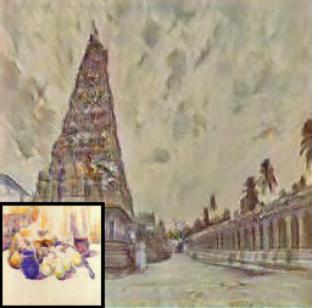
Cezanne



Ukiyoe



1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295



1296									1350
1297									1351
1298									1352
1299	Edge	Exemplar	Shoe	Edge	Exemplar	Shoe	Edge	Exemplar	Shoe
1300									
1301									
1302									
1303									
1304									
1305									
1306									
1307									
1308									
1309									
1310									
1311									
1312									
1313									
1314									
1315									
1316									
1317									
1318									
1319									
1320									
1321									
1322	Edge	Exemplar	Shoe	Edge	Exemplar	Shoe	Edge	Exemplar	Handbag
1323									
1324									
1325									
1326									
1327									
1328									
1329									
1330									
1331									
1332									<img alt



Figure 13. Visual results by EDIT on shoe/handbag → edge.

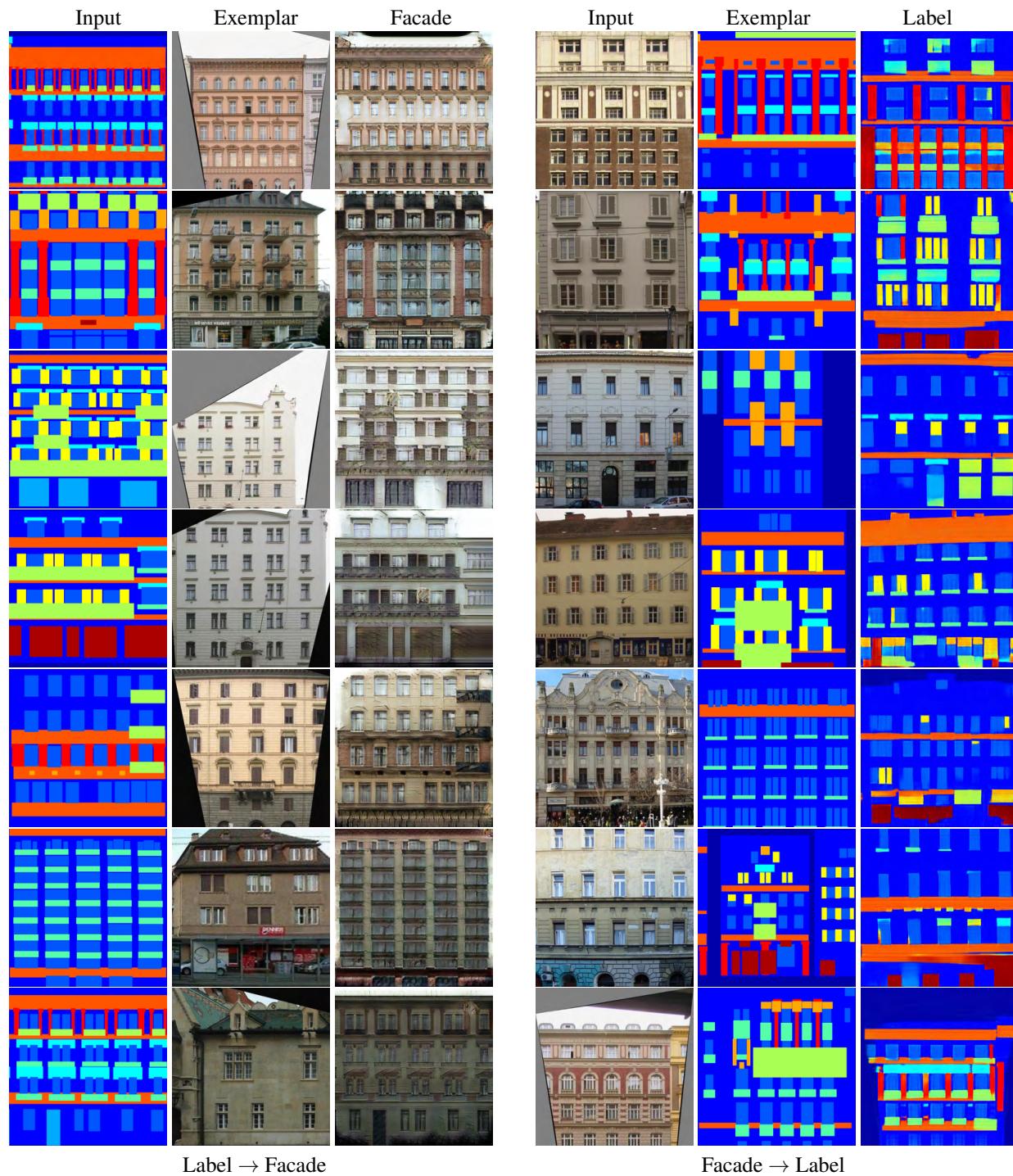


Figure 14. Visual results by EDIT on label ↔ facade.