Style transfer images

**Cropped images:**

* Habitat images: from E. caeruleum (most images) or E. olstedi (sand and boulder 2)

Preprocessing: White-balance correction (estimated camera parameters), rescaled (dimensions divided by 1.5), cropped (3000 x 1000 pixels)

* Fish images: E. caeruleum, 10 individuals

Preprocessing: rescaled (dimensions divided by 1.5), cropped (3000 x 1000 pixels)

**Post StyleTransfer Images**

Gatys et al., 2015; 2017 - PyTorch implementation with a colour control option (luminance-only transfer): [rrmina](https://github.com/rrmina/neural-style-pytorch/blob/master/neural_style_preserve_color.ipynb)

* All layers

content\_layers = ['relu4\_2']

content\_weights = {'relu4\_2': 1.0}

style\_layers = ['relu1\_2', 'relu2\_2', 'relu3\_3', 'relu4\_3', 'relu5\_3']

style\_weights = {'relu1\_2': 0.2, 'relu2\_2': 0.2, 'relu3\_3': 0.2, 'relu4\_3': 0.2, 'relu5\_3': 0.2}

* Layers 3 to 5

content\_layers = ['relu4\_2']

content\_weights = {'relu4\_2': 1.0}

style\_layers = ['relu1\_2', 'relu2\_2', 'relu3\_3', 'relu4\_3', 'relu5\_3']

style\_weights = {'relu1\_2': 0.0, 'relu2\_2': 0.0, 'relu3\_3': 0.33, 'relu4\_3': 0.33, 'relu5\_3': 0.33}

* Layers 1 and 2

content\_layers = ['relu4\_2']

content\_weights = {'relu4\_2': 1.0}

style\_layers = ['relu1\_2', 'relu2\_2', 'relu3\_3', 'relu4\_3', 'relu5\_3']

style\_weights = {'relu1\_2': 0.5, 'relu2\_2': 0.5, 'relu3\_3': 0.0, 'relu4\_3': 0.0, 'relu5\_3': 0.0}