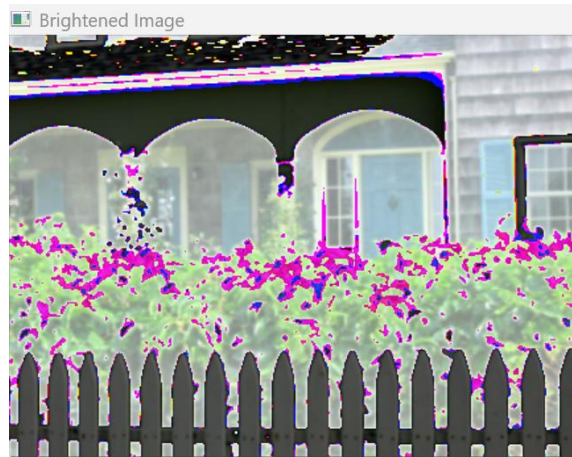


Computer Vision HW3

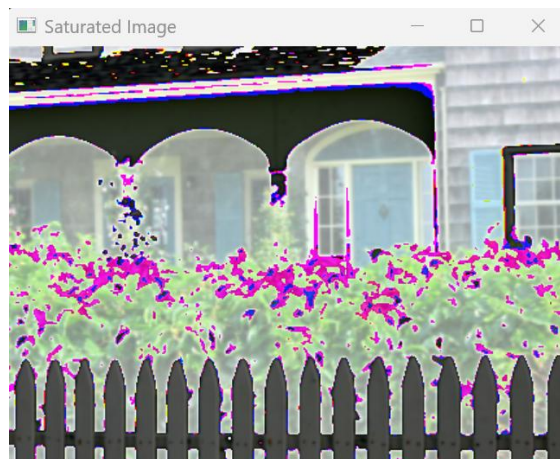
1. Torch image conversion

```
Torch image shape: torch.Size([321, 433, 3])  
Torch image dtype: torch.float32  
Torch Image pixel dtype: torch.float32
```

2. Brighten

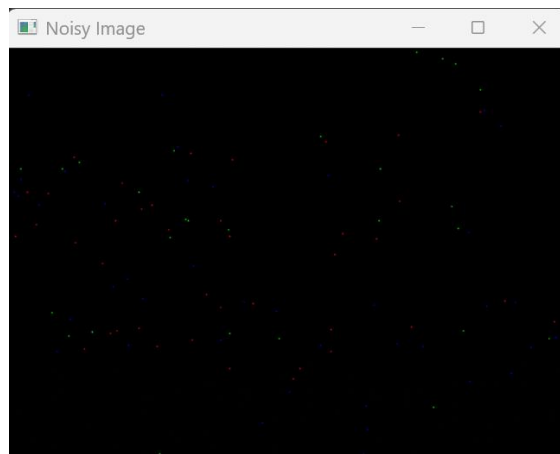


3. Saturation arithmetic



4. Noise

Note: OpenCV can't show float32 image, so I convert it to uint8.



5. Normalization image

```
Normalized mean: tensor([-1.0965e-16, -1.4047e-16, -3.0832e-17], dtype=torch.float64)
Normalized std: tensor([1.0000, 1.0000, 1.0000], dtype=torch.float64)
```

6. ImageNet normalization

```
ImageNet normalized mean: tensor([0.3005, 0.4367, 0.3942], dtype=torch.float64)
ImageNet normalized std: tensor([0.4292, 0.4441, 0.4508], dtype=torch.float64)
```

7. Dimensions rearrange

```
Shape: torch.Size([1, 3, 321, 433])
```

8. Stride

```
stride output shape: torch.Size([305, 457])
dtype: torch.float32
sample: tensor(-1300.)
```