

題目:卷軸神經網路

姓名:黃偉棟

學號:01057162

日期:16-4-2024

方法

設計 custom initializer。

已知 $k = 1, 5, 10$ ，那麼需要調整一些參數才能得到結果，就是 custom initializer 裏面的數值。

```
# Custom initializer function
def custom(shape, dtype=None):
    return keras.backend.constant(0.04, shape=shape, dtype=dtype)
```

若需要轉換成 YUV 色彩空間，還得另外定義 constant tensor。

```
# Custom initializer function for Gaussian blur
def gaussian_blur_initializer(shape, dtype=None):
    # Convert the kernel values to a TensorFlow constant
    return tf.constant([[0.0625, 0.125, 0.0625], [0.125, 0.25, 0.125], [0.0625, 0.125, 0.0625]], shape=shape, dtype=dtype)

# Custom initializer function for RGB to YUV conversion
def rgb_to_yuv_initializer(shape, dtype=None):
    # Standard conversion matrix from RGB to YUV
    return tf.constant([[0.257, 0.504, 0.098], [-0.148, -0.291, 0.439], [0.439, -0.368, -0.071]], shape=shape, dtype=dtype)

# Custom initializer function for YUV to RGB conversion
def yuv_to_rgb_initializer(shape, dtype=None):
    # Standard conversion matrix from YUV to RGB
    return tf.constant([[1.164, 0, 1.596], [1.164, -0.391, -0.813], [1.164, 2.018, 0]], shape=shape, dtype=dtype)
```

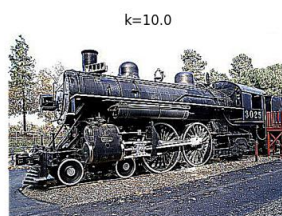
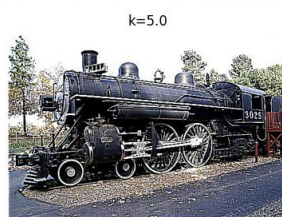
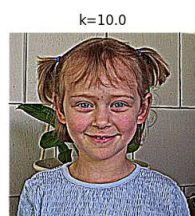
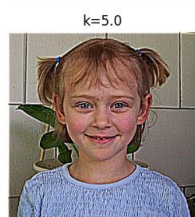
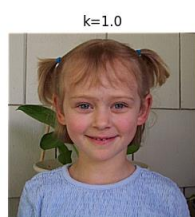
Gaussian blur – 影像處理的 tensor。

Rgb to yuv – 轉換成 yuv 的 tensor。

Yuv to rgb – 轉換成 rgb 的 tensor。

結果

Version 1



Version 2



結論

運算速度 – 在 colab, v2 只需要 5-7 秒便完成計算。v1 需要 22 秒左右才完成計算。

觀察 – v1 的 k 值越大, 影像中的白點越多, v2 的 k 值越大, 邊緣的紅色會很明顯。

參考文獻

(如果有參考文獻, 請列出)