

HW6

110590032 詹採晴

方法說明

在本次作業中，我們實現了Canny邊緣檢測算法，包括以下幾個步驟：

1. 噪聲去除：

使用高斯濾波器對圖像進行平滑處理，以減少噪聲對邊緣檢測的影響。我們定義了 `gaussian_filter` 函數，使用高斯核對圖像進行卷積操作。

2. 計算圖像梯度：

使用Sobel算子計算圖像在x和y方向上的梯度，得到梯度幅值和梯度方向。我們定義了 `sobel_operator` 函數，通過卷積計算圖像的梯度。

3. 非最大值抑制：

在梯度方向上進行非最大值抑制，保留局部最大值，去除非邊緣點。我們定義了 `non_maximum_suppression` 函數來實現這一過程。

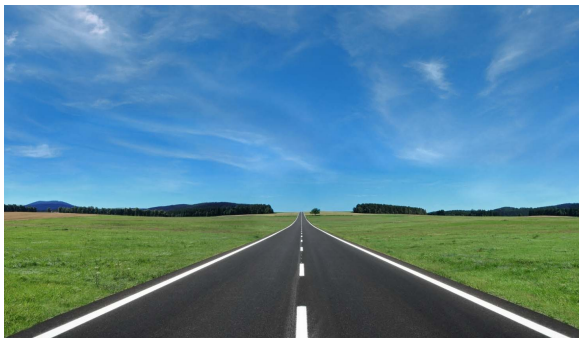
4. 雙閾值處理：

使用高低閾值對梯度圖像進行處理，確定強邊緣和弱邊緣。我們定義了 `double_threshold` 函數，將梯度值大於高閾值的設為強邊緣，小於低閾值的設為非邊緣，介於兩者之間的設為弱邊緣。

5. 邊緣連接：

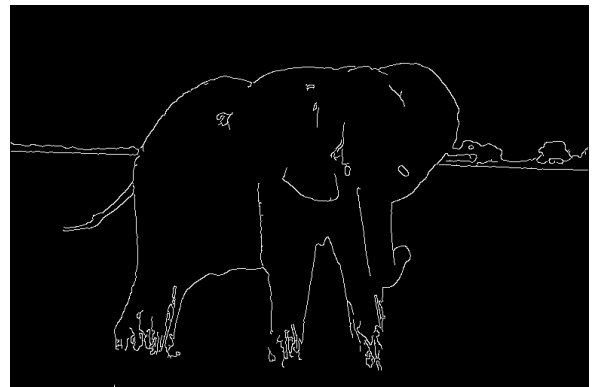
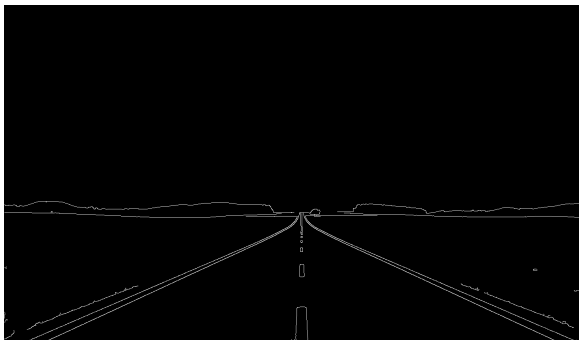
使用滯後閾值法追蹤邊緣，連接弱邊緣和強邊緣。我們定義了 `edge_tracking_by_hysteresis` 函數，將弱邊緣連接到強邊緣，去除孤立的弱邊緣點。

原始圖像





結果圖像



觀察

原本的邊緣追蹤算法是通過檢查每個弱邊緣像素的8個鄰域，如果至少有一個像素是強邊緣，則將該弱邊緣像素轉化為強邊緣，否則將其設置為0。

但這個方法的結果圖的邊緣都沒有連接起來，嘗試了多種的threshold和gaussian的sigma值，結果都沒辦法連接起來。

可能是因為這個方法僅依賴像素的局部信息，導致邊緣連接不完整或不連續。

後來在 `edge_tracking_by_hysteresis` 的部分使用了dfs的算法之後，通過遞歸搜索連接所有可能的弱邊緣像素，從而更好地連接邊緣，結果就良好非常多！