Edge Detection Experimentation using Sobel, Scharr, and Canny Methods

Kumar Ashutosh

Image Preprocessing

- Original image: `pikachu.png`
- Conversion to grayscale
- Application of Gaussian blur with kernel size (5, 5) and $\sigma = 2$ to reduce noise

Sobel Edge Detection

The Sobel operator calculates the gradient magnitude in X and Y directions using a 3x3 kernel. The combined magnitude highlights horizontal and vertical edges.

- Kernel size: 3
- Directional gradients: sobel_x and sobel_y
- Combined using: $\text{magnitude} = \sqrt{\text{sobel\_x}^2 + \text{sobel\_y}^2}$

Scharr Edge Detection

The Scharr operator is similar to Sobel but optimized for better rotational symmetry. It provides sharper edges and is better at capturing fine details.

- Operates with predefined kernels (fixed)
- Better for detecting subtle intensity changes

Comparison and Analysis

- **Sobel**: Works well on well-contrasted edges. Sometimes misses fine details. As the given image is well contrasted the Sobel works fine from very small values like 20 to very high values like 50-70.

- **Scharr**: Better gradient estimation, provides finer edge detail. In this example we do not find much difference in Scharr and Sobel. But when applied to images with difficult scenarios like shadows and background, it performs better.
- **Canny**: Best overall edge precision due to double-threshold hysteresis. The edge detection on Canny without gaussian blur sometimes includes unwanted lines like shadows as edges. It is difficult to exclude all unwanted lines in normal as well as canny mode but based on different values that I tried the above ones work the best.

Conclusion

The Canny-enhanced Scharr method provides the best overall result for this image, delivering clean and detailed edges. Sobel is computationally lighter and still effective in high-contrast scenarios. Use cases should guide the choice of operator: Scharr+Canny for precision, Sobel for speed, and Canny alone for robust edge isolation.