Image sorting and sequencing using Canny edge detection and Hough transforms

Martín Rodriguez
Maseeh College of Engineering and Computer Science
Portland State University
Portland, OR
mtr@pdx.edu

***Abstract*—Image processing gives us the ability to extract key information from images in a highly efficient manner. This paper attempts to apply concepts from classic image processing, namely Canny edge detection and Hough transforms, along with other image processing operations to the automatic classification and sorting of images for use in visual productions for accompaniment of musical performance. A metric is developed corresponding to the angle of lines detected using the Hough transform to distinguish between images containing rectangular content and those containing triangular content.**

***Index Terms*—image processing, computer vision, Canny edge detection, Hough line detection**

I. Introduction

Audio-reactive visual content can be used to enhance the experience of live music performances and create interesting accompaniment to recorded audio, such as for sharing via music video platforms. This project was particularly inspired by the work of Páraic Mc Gloughlin for the band Weval [1]. Traditionally, a visual artist would have to hand-select images or video and carefully sequence them such that they follow the dynamic, temporal, or harmonic content of the accompanying audio in some way. This process can be highly tedious if the number of images is in the thousands. Although this can be a highly subjective process, it is possible that this can be accomplished automatically using spatial, geometric, and color information collected across images to determine criteria to be applied in the image sorting process. Edge and line detection in images is a well-studied branch of image processing for reducing an image to basic components. Due to their robust nature and popular use in computer vision algorithms, the Canny edge detection algorithm [2] and the Hough transform [3] are employed.

A. Canny edge detection

One of the first steps in many image processing algorithms is edge detection. The goal of edge detection is to enhance strong color transitions which physically take the form of edges in an image and at the same time reduce the amount of information contained in the image for further processing. Canny [2] outlined a set of rules and a procedure for detecting edges in images such that they can be applied to a wide range of images parametrically.

Canny edge detection steps

1) Apply a Gaussian filter for noise reduction.
2) Find the intensity gradient in the horizontal and vertical directions and combine them to form the edge gradient G and direction θ for each pixel.

$$G = \sqrt{G_x^2 + G_x^2} \tag{1}$$

$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{2}$$

3) Non-maximum suppression using the gradient direction to thin the edges.
4) Hysteresis thresholding to separate the strong edges from weak edges.

B. Hough transform

While originally filed as a patent by Hough [3] for tracking particles in a bubble chamber, his method for finding lines in images by transforming the image into a parameter space was popularized by Duda and Hart [4], who recognized that using a Hesse normal representation solves the problem of vertical lines having unbounded slopes in the affine parameter space proposed by Hough. A mesh is created in the parameter space and point values are accumulated indicating collinear points, for which a voting scheme can be created to determine the strength of the line. A threshold is applied to select those lines with the most votes. The output from this algorithm are a pair of values, ρ and θ for each line, where ρ is the distance from the origin of the image to the closest point on the line, and θ is the angle of the line ($0°$ corresponding to a vertical line and $90°$ corresponding to a horizontal line). The Hough line transform has since been generalized to detect circles or arbitrary shapes in images, including in a probabilistic manner for increased computational efficiency [5] [6] [7] [8] [9]. Edge detection is a preliminary step in taking the Hough transform, and many algorithms employ the Canny method [10].

II. Methodology

A. Input images

Thirty images were selected from a pool generated via latent diffusion from a text input [11]. This method of image creation allows for a large amount of output with possibilities for randomness and feedback into future generations. An example prompt used was “realistic rectangular geometric windows extending toward infinity.” Viewing the output of another prompt, “realistic triangular geometric windows extending toward infinity,” a primary objective was identified: could these images be sorted from most rectangular and least triangular to most triangular and least rectangular?

B. *Reducing “line noise”*

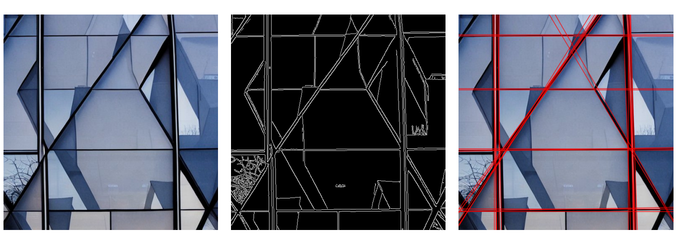
A problem was identified in the Hough transform output when applied with a simple accumulator threshold: many images contained more lines than were necessary to determine the geometric properties of the image, resulting both from smaller details receiving many votes and from overlapping lines in close spatial proximity that contained no additional information. Because the Canny edge detector uses hysteresis thresholding to select edges and the Hough line transform uses simple thresholding to select lines, it was expected that an algorithm which additionally selects only n of the most-voted edges or lines might rectify this issue. In addition, lines within the n selected lines could be ignored which have similar ρ values to a more prominent line, however, due to time limitations this was not implemented in the current project.

C. *Parameters for edge and line detection*

For edge detection, smoothing is the first step taken to reduce noise in the image. A Gaussian filter with a kernel size of 3x3 was selected. A 5x5 kernel could also be used, but this resulted in more lines from the Hough transform output than were deemed useful for most images, as can be seen in Figure 4. More lines is not always better, and past a certain point may be considered noise for the purposes of image sorting. A weak edge threshold of 50 and a strong threshold of 150 was selected. Canny recommended a similar ratio, in the range of 2:1 or 3:1 due to the likely ratio of signal to noise [2]. For the Hough transform, the distance granularity selected for ρ was 1 and the angle granularity for θ was 1°. The Hough line accumulator threshold was initially set to 200, but was reduced to 80 to detect more lines across all images, of which the 30 lines per image with the most votes were kept.

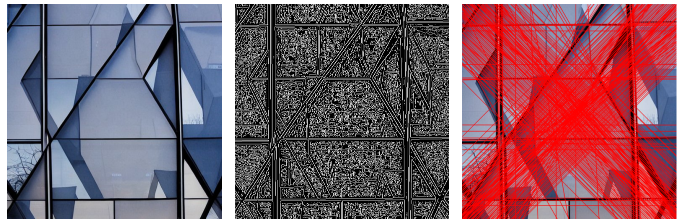
Histograms were generated for each image from the Hough line transform output corresponding to the angles of the detected lines. The angles were placed in eight bins, from 0° to 180°.

III. Results



(a) Original (b) Edges (c) Lines

Fig. 1: Canny edge and Hough line detection with 3x3 kernel



(a) Original (b) Edges (c) Lines

Fig. 2: Canny edge and Hough line detection with 5x5 kernel

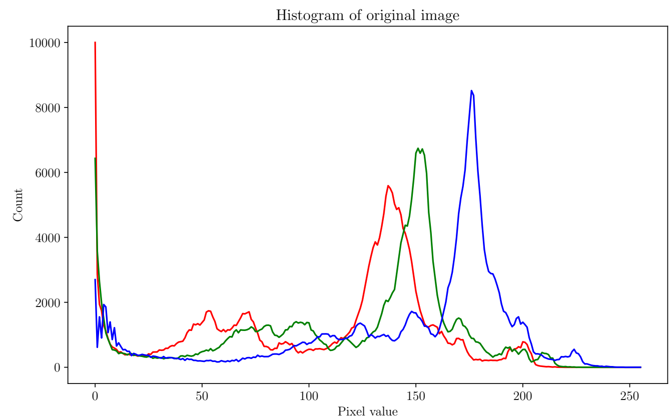
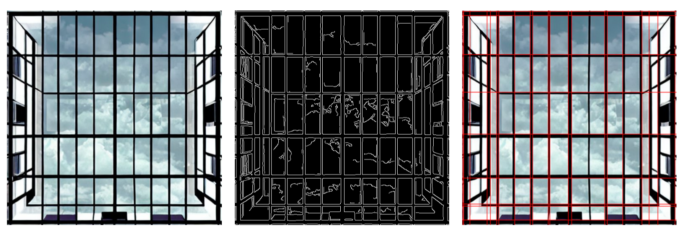
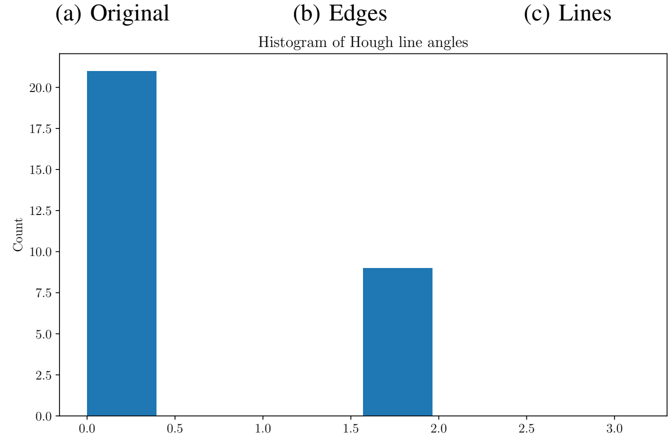


Fig. 3: RGB histogram of original image shown in Figure 1

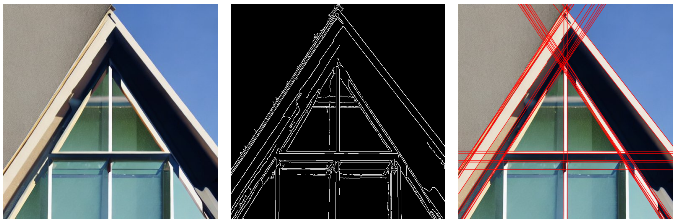


(a) Original (b) Edges (c) Lines

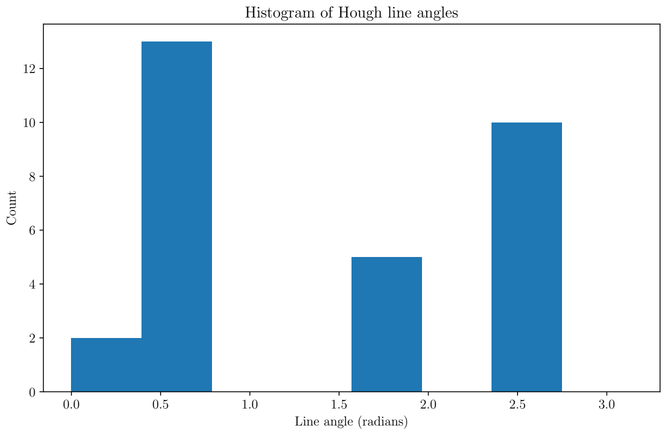


(d) Histogram of detected line angles

Fig. 4: Image with rectangular content and its line angle histogram

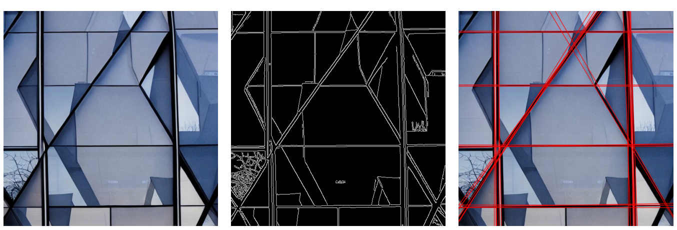


(a) Original (b) Edges (c) Lines

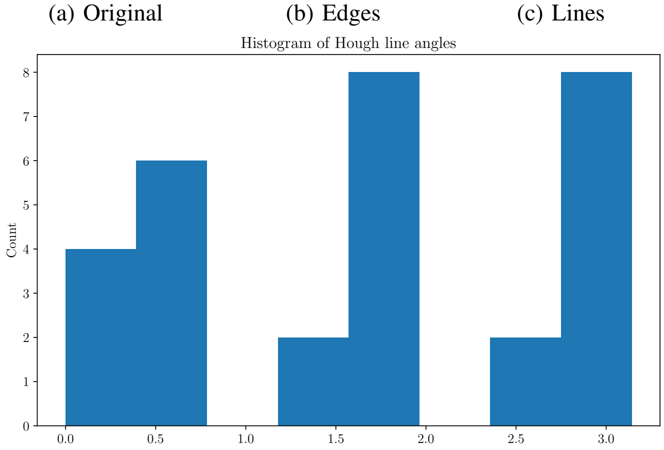


(d) Histogram of detected line angles

Fig. 5: Image with triangular content and its line angle histogram



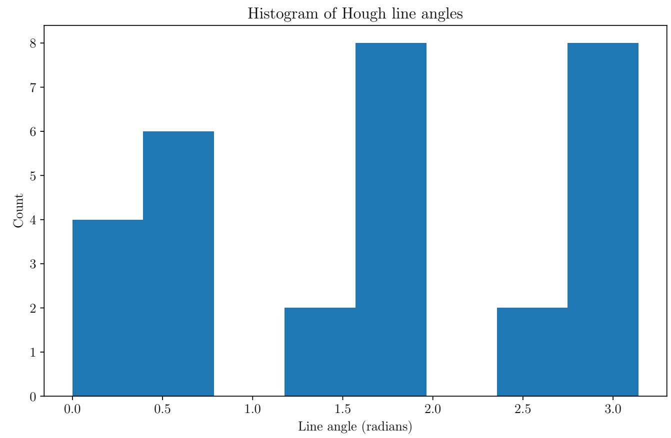
(a) Original (b) Edges (c) Lines



(d) Histogram of detected line angles

Fig. 6: Image with rectangular and triangular content and its line angle histogram

(a) Original (b) Threshold=200 (c) Threshold=80, 30 lines



(d) Histogram of detected line angles

Fig. 7: Comparison of Hough line output with simple threshold vs. selecting 30 lines with most votes

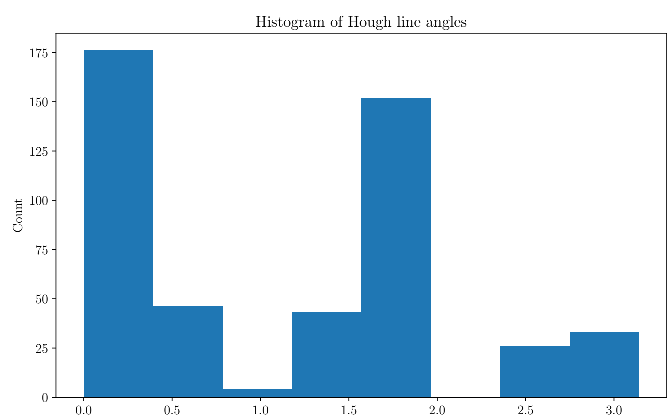
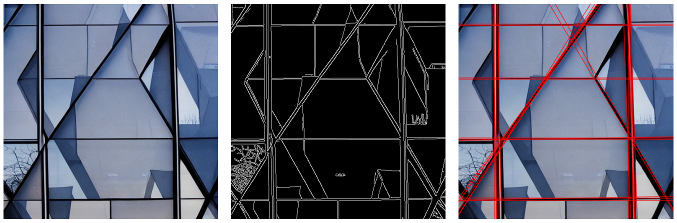
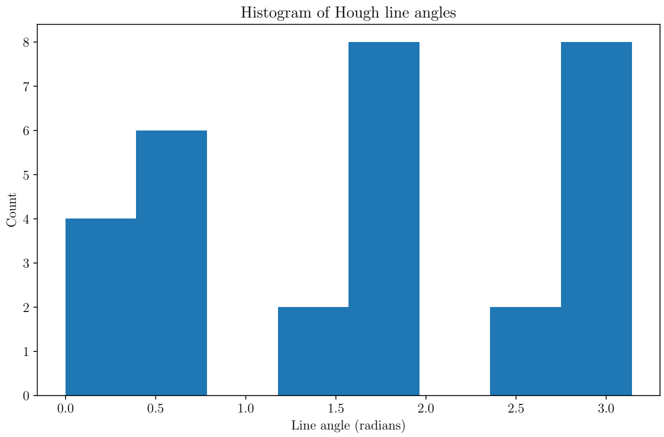


Fig. 8: Histogram of detected angles, all images



(a) Original (b) Edges (c) Lines



(d) Histogram of detected line angles

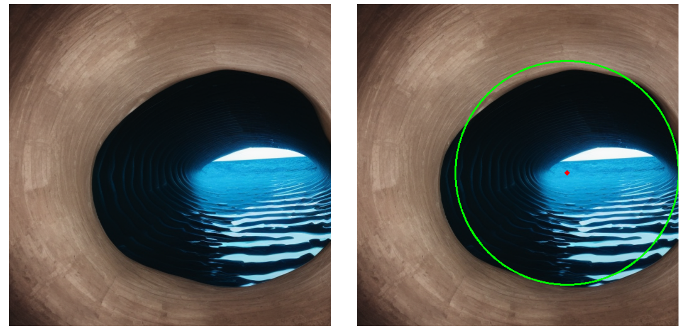
Fig. 9: Image with rectangular and triangular content and its line angle histogram

(a) Original (b) Circles

Fig. 10: Hough circle transform applied to image

(a) Original (b) Circles

Fig. 11: Hough circle transform applied to image



(a) Original (b) Circles

Fig. 12: Hough circle transform applied to image

IV. Discussion

Comparing the angle histograms of individual images, images with more rectangular geometrical content had more angles within the $0°–22.5°$, the $67.5°–90°$ or $90°–112.5°$, and the $157.5°–180°$ bins. Images with more triangular content had more angles in the intermediate bins. Of course, this supposes that all rectangles appear upright within the image and their angles near-perpendicular to the image borders. Smaller bin widths would yield more options for categorization. Rather than using the count of angles, the normalized histograms and the ratios between bin counts could yield a more robust categorization criteria which would be insensitive to the presence of more lines within a particular image.

Figure 7 shows a comparison of the Hough line output when using the simple threshold set at 200 and using a threshold of 80 and also selecting only the 30 most-voted lines. The simple threshold has trouble with the foliage or small details in the image, which it incorrectly identifies as being a major geometric tendency in the image. It is also unclear what detail of the image is causing the diagonal lines to appear when comparing the simple threshold output to the original image. The algorithm that selects the 30 most-voted lines proved to perform better on a wider range of images for the objective of separating rectangular details from triangular details and ignoring the smaller irrelevant details.

The histogram of all line angles detected in all images is shown in Figure 8. As expected, the number of line angles close to $0°$ and $90°$ outweighed the other angles, since the image output from the latent diffusion "rectangular" prompt tended to have more composite rectangles and the triangular images necessarily included angles near $90°$ as well.

Figures 10–12 show a preliminary application of the Hough circle transform [5] [6] to three images created from the latent diffusion prompt, "breathy brethren inside of a whale's mouth in the style of francis bacon." Good results were obtained with a modest amount of parameter optimization, so this method could be further developed in future work to establish a another criteria for image sorting where the line transform fails.

V. Conclusion

Prior work has shown that edge and line detection is useful in the quantization of geometric information within an image.

The Hough line transform was successfully applied to a series of images and a criteria was developed for distinguishing images with rectangular content from those with triangular content based on their detected line angles. More work needs to be done to further develop this technique, including the sorting of the images using the information obtained thus far and a possible generalization to other shapes or criteria such as red-green-blue (RGB) color distributions or value distributions after translation to the hue-saturation-value (HSV) color representation.

REFERENCES

[1] P. Mc Gloughlin, "Weval - Someday (Official Video)," https://www.youtube.com/watch?v=n-wEvzqdDZg, 2019.

[2] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[3] P. V. C. Hough, "Method and means for recognizing complex patterns," U.S. Patent US3 069 654A, Dec. 18, 1962.

[4] R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Commun. ACM*, vol. 15, no. 1, pp. 11–15, Jan. 1972.

[5] C. Kimme, D. Ballard, and J. Sklansky, "Finding circles by an array of accumulators," *Communications of the ACM*, vol. 18, no. 2, pp. 120–122, 1975.

[6] H. Yuen, J. Princen, J. Illingworth, and J. Kittler, "Comparative study of hough transform methods for circle finding," *Image and Vision Computing*, vol. 8, no. 1, pp. 71–77, 1990. [Online]. Available: https://www.sciencedirect.com/science/article/pii/026288569090059E

[7] J. Illingworth and J. Kittler, "The adaptive hough transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, no. 5, pp. 690–698, 1987.

[8] D. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 13, no. 2, pp. 111–122, 1981.

[9] N. Kiryati, Y. Eldar, and A. Bruckstein, "A probabilistic hough transform," *Pattern Recognition*, vol. 24, no. 4, pp. 303–316, 1991.

[10] H. Kälviäinen, P. Hirvonen, L. Xu, and E. Oja, "Comparisons of probabilistic and non-probabilistic hough transforms," in *Computer Vision — ECCV '94*, J.-O. Eklundh, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994, pp. 350–360.

[11] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 10 684–10 695.