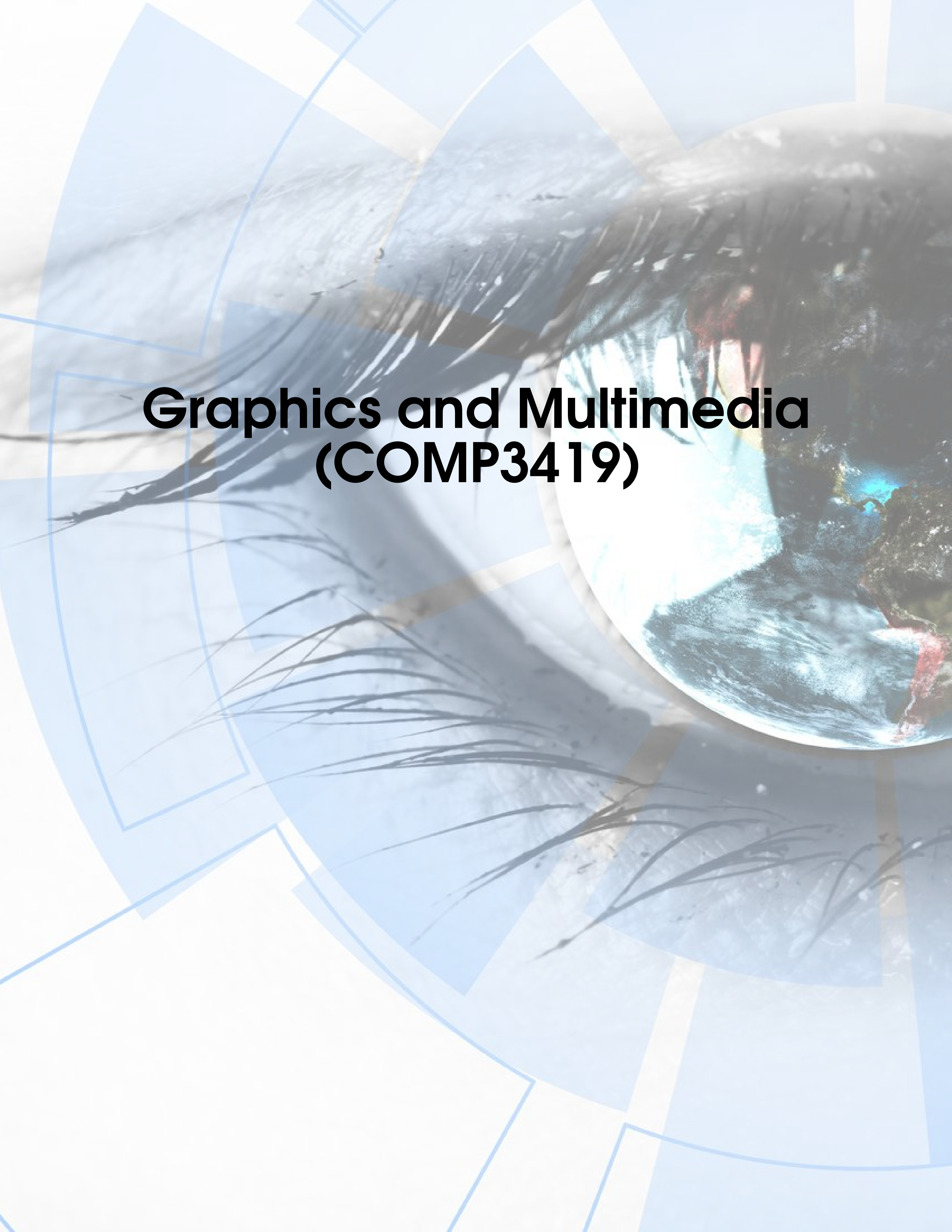
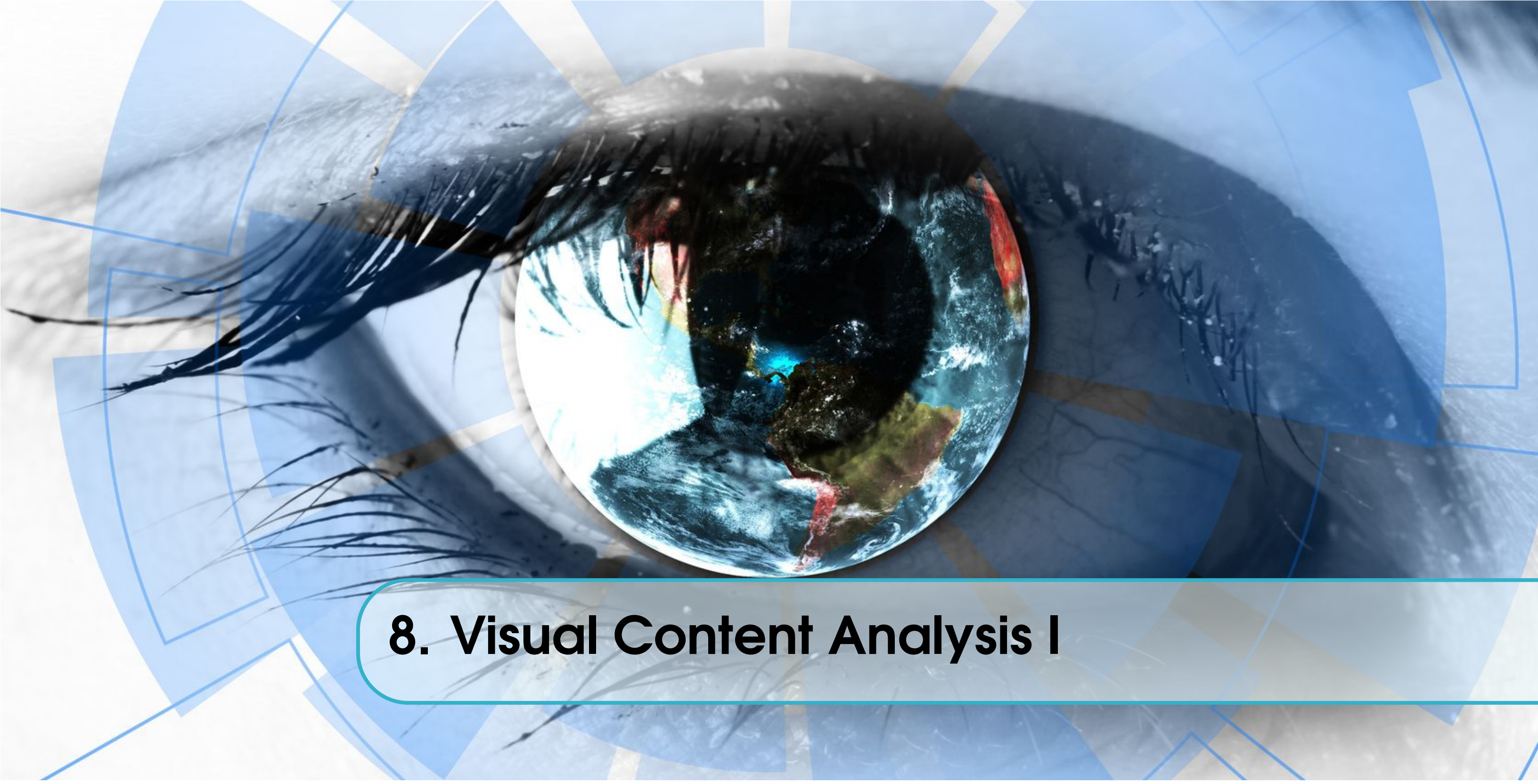


Graphics and Multimedia (COMP3419)



8. Visual Content Analysis I

Hough Transform is a feature extraction technique used in image analysis, computer vision, and digital image processing. The hough space is defined using θ (the rotation angle of a line from the line $y = 0$) and ρ (the perpendicular distance between this line and $(0,0)$), where θ ranges from 0 to 2π. The sampling frequency of θ can be customised; ρ ranges from 0 to ∞. By default, 0.1 is chosen as the sampling frequency. For more information about Hough Transform, please refer to http://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm.

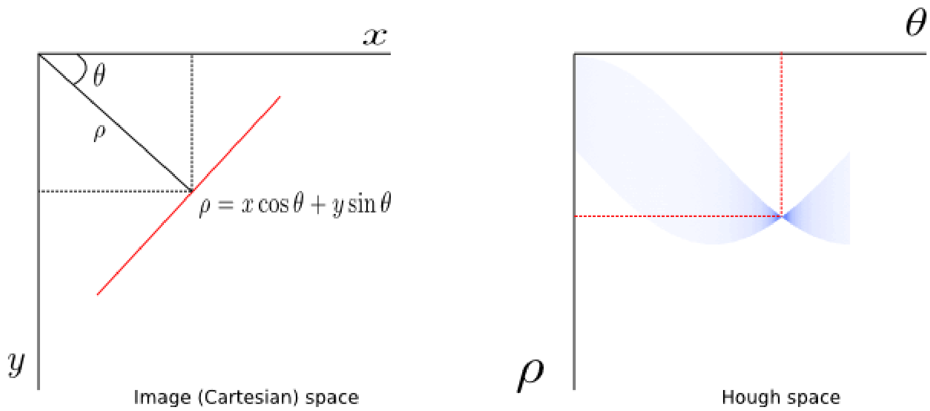
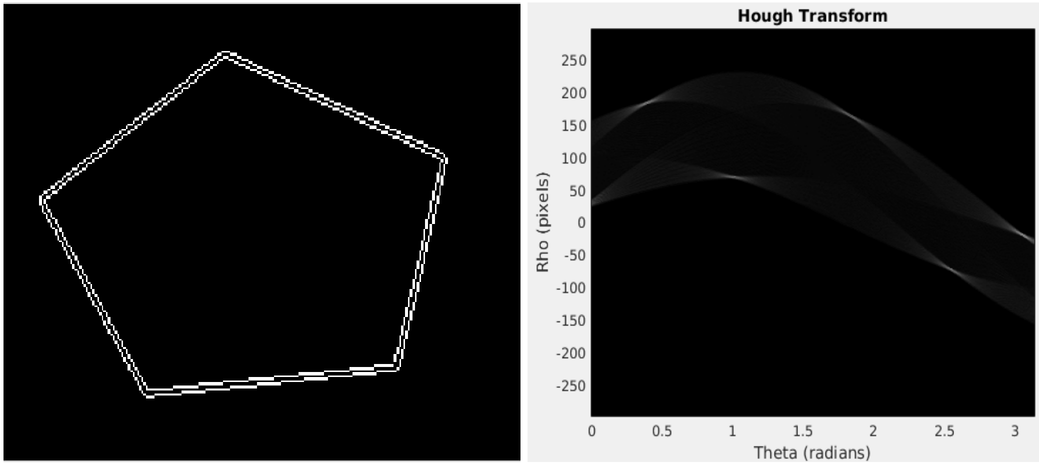


Figure 8.1: An illustration of the Hough transform.

8.1 Hough Transform

In this task, please implement the Hough transform and apply it to image '*binary.png*'.

- Download '*binary.png*' from eLearning.
- Convert '*binary.png*' into gray scale and binarise it. The binary image would look similar to Fig. 8.2(a).



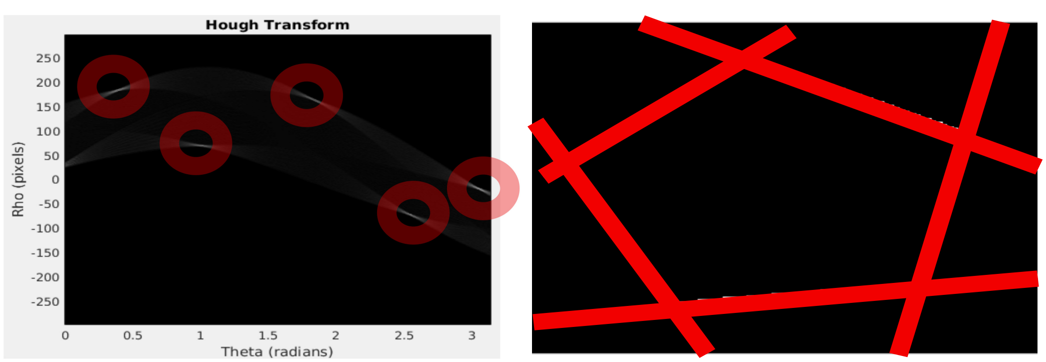
(a) Example of binary image (b) Example of Hough Transform

Figure 8.2: An example result of Hough Transform.

- Initialise a 2D matrix H with 0s. The size of H is defined by the sampling rate of θ and ρ. e.g. if 0.1 is used as the sampling frequency for θ, the width of H should be $\lceil 2\pi/0.1 \rceil$. It is notable that the upper bound of ρ should be $\sqrt{M^2+N^2}$, where M and N are the width and height of the input image. The matrix H will be used for storing the Hough transform result.
- Apply Hough Transform to the binarised image, to map it from (x,y) space into (θ,ρ) space following equation:

$$xcos\theta + ysin\theta = \rho \tag{8.1}$$

Each (x,y) coordinate of the foreground pixels corresponds to a list of discrete (θ,ρ) pairs in Hough space, which can be obtained by substituting θ with values ranging from 0 to 2π in order to get ρ. For each pair of (θ,ρ), increment the value in $H(\theta,\rho)$ by 1. To visualise H as image, normalise the values $H(x,y)$ in H by $256 * \frac{H(x,y)}{max(H)}$. The successful transformation would be similar to Fig. 8.2(b).



(a) Local maximum of Hough Transform (b) Visualisation of Inverse Hough Transform

8.2 Line Detection

Find the coordinates of the local peaks of straight lines shown in Fig 8.3(a). The simplest way of finding these peaks is by applying some form of threshold, but other techniques may yield better results in different circumstances – determining which lines are found as well as how many. Try to implement a simple local peak searching algorithm with maxima suppression as below

Data: Hough Transform image H, number of peaks to find N
Result: A list L of coordinates in Hough space (θ_i, ρ_i)
while *The number of found peaks is smaller than* N **do**
  $(\theta, \rho) \leftarrow$ Find coordinate in H with maximum value;
  Append (θ, ρ) to L;
  Assign 0 to the area around (θ, ρ);
end

Algorithm 1: Simple algorithm to find local peaks in Hough Space with maxima suppression.

8.3 Inverse Hough Transform

Program the inverse hough transform of previous points and draw them in (x, y) space. The visualisation would be similar to Fig. 8.3(b). With the list L of peaks (θ_i, ρ_i), draw a line for each (θ_i, ρ_i). To draw a line, you need to convert (θ, ρ) to the form $y = kx + b$ at first; then sample two points on this line where $y = 0$ and $y = height$ to draw a segment. Please note lines captured in Hough space do not contain information about lengths.