Week 5 Homework Q2

Telmen Enkhbold

San Fransico Bay University

CE480 - Java and Internet Application

Dr. Chang, Henry

10/12/2023

# Author Note

# The Question

Use array element as an index to another array element.

* + Step 1: Design a program to compute the histogram of a 2D array and display the result. Where the histogram is the frequency counts of the population of elements of each array value. For example, the histogram of the following 10 \* 10 array. (Assume the range of the array value is from 0 to 255)
  + 1 3 5 7 9 3 4 4 5 6
  + 1 3 5 7 9 3 4 4 5 6
  + 1 3 5 7 9 3 4 4 5 6
  + 1 3 5 20 25 24 33 5 6 4
  + 1 3 5 22 35 24 32 5 6 4
  + 1 3 5 20 28 34 23 5 6 4
  + 1 3 5 21 25 27 23 5 6 4
  + 1 3 5 7 9 3 4 4 5 6
  + 1 3 5 7 9 3 4 4 5 6
  + 1 3 5 7 9 3 4 4 5 6
  + is
  + gray-level #-of-pixels
  + 1 10
  + 3 16
  + 4 16
  + 5 20
  + 6 10
  + 7 6
  + 9 6
  + 20 2
  + 21 1
  + 22 1
  + 23 2
  + 24 2
  + 25 2
  + 27 1
  + 28 1
  + 32 1
  + 33 1
  + 34 1
  + 35 1

Note:

* + - Sample C code

You only need to have two arrays in your program

#define max 256

#define N 10

void main()

{

/\* Image data \*/

int image[N][N]={

{1,3,5,7,9,3,4,4,5,6},

{1,3,5,7,9,3,4,4,5,6},

{1,3,5,7,9,3,4,4,5,6},

{1,3,5,20,25,24,33,5,6,4},

{1,3,5,20,35,24,32,5,6,4},

{1,3,5,20,28,34,23,5,6,4},

{1,3,5,21,25,27,23,5,6,4},

{1,3,5,7,9,3,4,4,5,6},

{1,3,5,7,9,3,4,4,5,6},

{1,3,5,7,9,3,4,4,5,6}

};

/\* Histogram of the image data \*/

int hist[max]={0};

/\* The code size of calculating a histogram from

\* an image data is less than 5 lines

\*/

....

....

}

* + Step 2: Please also verify your program with this image
  + 1 3 5 7 9 3 4 4 5 6
  + 1 20 25 24 3 5 6 4 2 4
  + 1 22 35 24 3 5 6 4 5 7
  + 1 20 28 34 2 5 6 4 8 9
  + 1 3 5 7 9 3 4 4 5 6
  + 1 3 5 7 9 3 67 4 5 6
  + 1 3 5 7 9 78 54 94 5 6
  + 1 3 5 7 9 99 98 54 5 6
  + 1 3 5 7 9 3 64 4 5 6
  + 1 3 5 7 9 3 4 4 5 6

* + Step 3: Implement the code using object-oriented approach. For example, for Java
  + class Image
  + {
  + private int[][] pixels;
  + public Image(int[][] pixels) {
  + ....
  + }
  + public int[] histogram() {
  + ....
  + return histogram;
  + }
  + }
  + }
  + public class TestImage {
  + public static void main(String[] args) {
  + int[][] pixelArray = {
  + {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},
  + {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},
  + {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},
  + {1, 3, 5, 20, 25, 24, 33, 5, 6, 4},
  + {1, 3, 5, 22, 35, 24, 32, 5, 6, 4},
  + {1, 3, 5, 20, 28, 34, 23, 5, 6, 4},
  + {1, 3, 5, 21, 25, 27, 23, 5, 6, 4},
  + {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},
  + {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},
  + {1, 3, 5, 7, 9, 3, 4, 4, 5, 6}
  + };
  + Image image = new Image(pixelArray);
  + ........
  + }

Note:

* + - You can get a [hint](https://hc.labnet.sfbu.edu/~henry/sfbu/course/image/pattern_recog/slide/histogram_hint.html) from ChatGPT
  + References
    - [Yuhong Luo](https://hc.labnet.sfbu.edu/~henry/sfbu/course/image/pattern_recog/hw/2019_fall/wk3_19509_YuhongLuo.pdf) - Java implementation, 2019 Fall
    - [The histogram of image intentisty](http://ftp.utcluj.ro/pub/users/nedevschi/IP/IP_Labs_2011/ipl_03e.pdf) ([local copy](https://hc.labnet.sfbu.edu/~henry/sfbu/course/image/pattern_recog/hw/nedevschi))
  + [MapReduce solution](https://hc.labnet.sfbu.edu/~henry/sfbu/course/cloud_computing/week1/mid/mid1113_ans.html#Programming:%20Histogram)

Screenshot

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer program

Description automatically generated

The Source Code

------------------------------------Histogram Processor----------------------------------

public class HistogramProcessor {

    public static void main(String[] args) {

        System.out.println("Histogram for Image1:");

        Image1 image1 = new Image1();

        int[] histogram1 = processHistogram(image1.getPixelArray());

        displayHistogram(histogram1);

        System.out.println("\nHistogram for Image2:");

        Image2 image2 = new Image2();

        int[] histogram2 = processHistogram(image2.getPixelArray());

        displayHistogram(histogram2);

        System.out.println("\nHistogram for Image3:");

        Image3 image3 = new Image3();

        int[] histogram3 = processHistogram(image3.getPixelArray());

        displayHistogram(histogram3);

    }

    // Function to compute the histogram for a given pixel array

    public static int[] processHistogram(int[][] pixelArray) {

        int[] histogram = new int[256]; // Assuming the range is from 0 to 255

        for (int i = 0; i < pixelArray.length; i++) {

            for (int j = 0; j < pixelArray[i].length; j++) {

                int pixelValue = pixelArray[i][j];

                histogram[pixelValue]++;

            }

        }

        return histogram;

    }

    // Function to display the histogram

    public static void displayHistogram(int[] histogram) {

        System.out.println("Gray-level\t#-of-pixels");

        for (int i = 0; i < histogram.length; i++) {

            if (histogram[i] > 0) {

                System.out.println(i + "\t\t" + histogram[i]);

            }

        }

    }

}

-----------------------------------------------Image1--------------------------------------------------------

public class Image1 {

    private int[][] pixelArray = {

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 20, 25, 24, 33, 5, 6, 4},

        {1, 3, 5, 22, 35, 24, 32, 5, 6, 4},

        {1, 3, 5, 20, 28, 34, 23, 5, 6, 4},

        {1, 3, 5, 21, 25, 27, 23, 5, 6, 4},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6}

    };

    public int[][] getPixelArray() {

        return pixelArray;

    }

}

-------------------------------------------------------------Image2-------------------------------------------

public class Image2 {

    private int[][] pixelArray = {

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 20, 25, 24, 3, 5, 6, 4, 2, 4},

        {1, 22, 35, 24, 3, 5, 6, 4, 5, 7},

        {1, 20, 28, 34, 2, 5, 6, 4, 8, 9},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 67, 4, 5, 6},

        {1, 3, 5, 7, 9, 78, 54, 94, 5, 6},

        {1, 3, 5, 7, 9, 99, 98, 54, 5, 6},

        {1, 3, 5, 7, 9, 3, 64, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6}

    };

    public int[][] getPixelArray() {

        return pixelArray;

    }

}

---------------------------------------------------Image3---------------------------------------------------

public class Image3 {

    private int[][] pixelArray = {

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 20, 25, 24, 33, 5, 6, 4},

        {1, 3, 5, 22, 35, 24, 32, 5, 6, 4},

        {1, 3, 5, 20, 28, 34, 23, 5, 6, 4},

        {1, 3, 5, 21, 25, 27, 23, 5, 6, 4},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6},

        {1, 3, 5, 7, 9, 3, 4, 4, 5, 6}

    };

    public int[][] getPixelArray() {

        return pixelArray;

    }

}