

Example Questions – Binary Images

2. נתונה תמונת מיקרוסקופ של סיבים בעוביים שונים. התמונה ביטורית (הרקע בדרגת אפור 0). השתמש בכלים של עיבוד תמונה כדי לחשב את התונים הבאים:



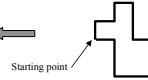
1. מספר הסיבים בתמונה.
 2. העובי הממוצע של הסיבים בתמונה.
 3. האורך הממוצע של הסיבים בתמונה.
- ניתן להניח שהסיבים מבודדים ("א אינם" נוגעים" אחד בשני) וכן שהסיבים אינם על שפת התמונה.



1. נתונה סדרית Chain-Code בקשירות 4 לפי הקוד הבא:

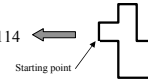
א. תן אלגוריתם בודק האם הסדרית מגדירה עקומה סגורה. עקומה סגורה: נקודת התחלה = נקודת סיום. לונגטא:

12123323344114



ב. במדה והתקומה אכן סגורה, תן אלגוריתם המחשב מתוך הסדרית את השטח התחום בעקומה.

Area = 7



3. א. למדנו מספר הנדרות ל"מרחק" בתמונת דיגיטליות:
 de - מרחק אוקלידי
 d4 - מרחק בשכנת 4
 d8 - מרחק בשכנת 8

איוה מכין האפשרויות הבאות נכונות:

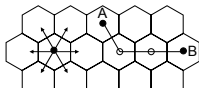
1. $d_4 \leq d_8 \leq d_e$
2. $d_8 \leq d_e \leq d_4$
3. $d_e \leq d_8 \leq d_4$
4. $d_8 \leq d_4 \leq d_e$

5. יותר מתשובה אחת נכונה - היחס הלוי במקרה.

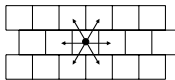
הוכח את תשובתך!

ב. בתמונה שדגומה בסריג משושה נגדיר מרחק d_6 .

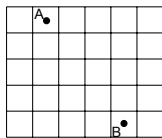
דוגמא:
 $d_6(A,B) = 3$



בתמונה שדגומה בסריג ריבועי ניתן לדמות סריג משושה ע"י הסטה של כל שורה שניה ב-1/2 פיקסל ימינה. כך לכל פיקסל יש 6 שכנים:

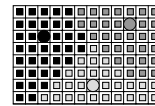


שאלה: מה המרחק d_6 של 2 הפיקסלים A, B המסומנים:



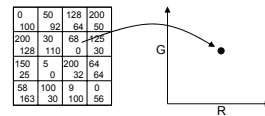
ג. במקום לחסות כל שורה ב-1/2 פיקסל ימינה, נסיט 1/2 פיקסל שמאלה. האם המרחקים d_6 השמורים? הוכח.

1. נתונה K נקודות על סריג ריבועי. דיאגרמת Voronoi מתארת מיפוי של כל פיקסל בסריג לנקודה הקרובה ביותר אליו (פיקסל שמרחקו זהה למספר נקודות - ממושה שרירותית לאחת מהנקודות. ראה דוגמא. (מסמן נקודה. מ מסמן שהנקודה בצבע זה קרובה ביותר לפיקסל זה).



תאר אלגוריתם יעיל לבניית דיאגרמת Voronoi כאשר המרחקים הם d_4 .

3. נתונה תמונה פסודו-צבעונית: RG בכל פיקסל שני ערכים בתחום 0...255. המסמלים את "כמות" האדום ואת "כמות" הירוק בפיקסל.

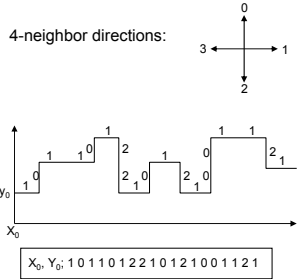


ניתן לייצג כל צבע כנקודה במרחב דו-ממדי (RG). כיוון שהערכים שלמים בתחום 0...255, מרחב הצבע הדו-ממדי יהיה מרחב בדיד בגודל 256×256 .

מבקשים לבצע כמות (Quantization) של תמונה זו ל-k צבעים בלבד. בהנתן k הצבעים, תאר שיטה מהירה ופשוטה לשיך כל פיקסל בתמונת ה-RG לצבע הקרוב ביותר מתוך k הצבעים (מדדו מרחק במטריקה d_e , d_4 או d_8 לבחירתכם). על מורכבות זמן הריצה להיות בלתי תלוייה ב-k.

Question:

A one parameter function can be described by its starting point and chain code as follows:



Give an algorithm that calculates the function integral from its chain code (the area enclosed by the function).

שאלה מספר 4

במהלך הקורס למדנו אלגוריתם 2-pass המוצא מרחק מינימלי של כל פיקסל בתמונה לאחד מהפיקסלים בקבוצה שהוגדרו כגרעין (seed).

האלגוריתם הוגדר עבור מרחק - d4 ועבור מרחק - d8.

א. האם ניתן להרחיב האלגוריתם כך ש-2 מעברים על התמונה מחושב המרחק המינימלי d המוגדר באופן הבא:

$$p = (p_x, p_y), \quad q = (q_x, q_y)$$

$$d(p, q) = |p_x - q_x|$$

אם כן, תאר את האלגוריתם, אם לא, הסבר ותן דוגמא נגדית.

ב. ענה על הסעיף הקודם עבור המרחק:

$$d(p, q) = |p_y - q_y|$$

ג. ענה על הסעיף הקודם עבור המרחק:

$$d(p, q) = \min(|p_x - q_x|, |p_y - q_y|)$$

ד. נגדיר תמונה תלת-מימדית $I(x, y, z)$ כך שמיקומו של כל פיקסל מוגדר ע"י שלשה (x, y, z) . האם ניתן להרחיב את אלגוריתם ה-2-pass המוצא מרחק מינימלי d6 של כל פיקסל תלת-מימדי בתמונה לאחד מהפיקסלים התלת-מימדיים בקבוצת הגרעין (seed)? מה מספר המעברים המינימלי הדרוש? תאר האלגוריתם והסבר.

מרחק d6 מוגדר באופן הבא:

$$p = (p_x, p_y, p_z), \quad q = (q_x, q_y, q_z)$$

$$d6(p, q) = |p_x - q_x| + |p_y - q_y| + |p_z - q_z|$$

שאלה מספר 4

(Updated from the class slides – red squares mark the lines that changed)

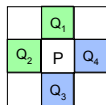
For each pixel calculate the d_x distance from a pixel in set S. Assume 4-neighborhood

2 passes:

Pass 1: scan image left-to-right and top-to-bottom

Pass 2: scan image right-to-left and bottom-to-top.

For each pixel P mark as follows:



Pass 1: consider all neighbors Q_i of P that have been scanned: \square

$$d'(P, S) = \begin{cases} 0 & \text{if } P \in S \\ \min \{d'(Q_1, S), d'(Q_2, S)+1\} & \text{if } P \notin S \end{cases}$$

Pass 2: consider all neighbors Q_i of P that have been scanned: \square

$$d''(P, S) = \min \{d'(P, S), d''(Q_3, S), d''(Q_4, S)+1\}$$

Exact same as above except exchange Q_1 with Q_2 and Q_3 and Q_4 .

1st Pass: Maintain 2 values for each pixel: $d'_x(P, S)$ and $d'_y(P, S)$

Each calculated according to previous 2 sections.

2nd Pass: calculate $d''_x(P, S)$ and $d''_y(P, S)$ as in the previous 2 sections And then perform $\min(d''_x(P, S) \text{ and } d''_y(P, S))$

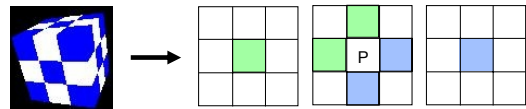
ד.

2 passes suffice:

Pass 1: scan image according to increasing x, y, z lexicographically (i.e. left-to-right, top-to-bottom and front-to-back).

Pass 2: scan image decreasing z, y, x lexicographically (i.e. Back-to-front, bottom-to-top and right-to-left).

For each pixel P mark the cube of 4-neighbors as follows:



The algorithm itself is exactly as defined in class for 2D:

Pass 1: consider all neighbors of P that have been scanned $N_1 = \square$

$$d'(P, S) = \begin{cases} 0 & \text{if } P \in S \\ \min_{Q \in N_1} \{d'(Q, S)+1\} & \text{if } P \notin S \end{cases}$$

Pass 2: consider all neighbors of P that have been scanned $N_2 = \square$

$$d''(P, S) = \min_{Q \in N_2} \{d'(P, S), d'(Q, S)+1\}$$