



אוניברסיטת אריאל בשומרון

פקולטה: הנדסה

מחלקה: הנדסת חשמל ואלקטרוניקה

שם הקורס: ראייה ממוחשבת, עיבוד תמונה וצילום חישובי

קוד הקורס: 4377010-1

תאריך בחינה : 5/2/2017 סמ' א מועד א

משך הבחינה: שעתיים

שם המרצה: ד"ר אופיר פלא

חומר עזר: סגור

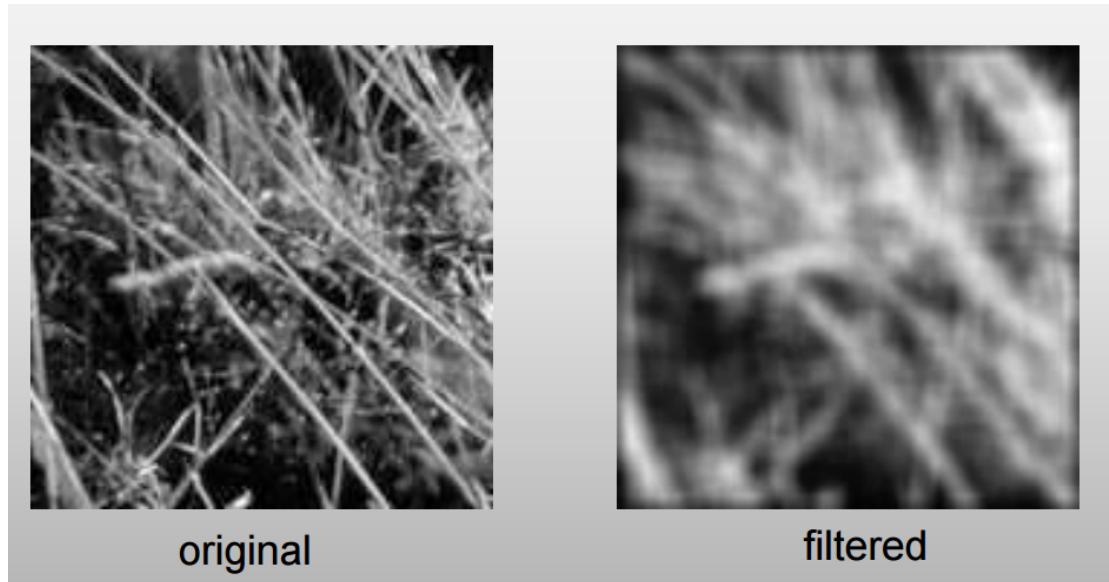
שימוש במחשבון: לא

הוראות כלליות:

- הניסוח הוא בלשון זכר מטעמי נוחות ומתייחס לכולם/ן!
- אין בחירה במבחן. יש לענות על כל השאלות.
- ניתן להסתמך על כל סעיף במבחן גם אם לא פתרתם אותו על מנת לפתור סעיף אחר במבחן.
- ניתן לרשום "לא יודע/ת" על סעיף ולזכות ב 20% מהנקודות המוקנות לסעיף הספציפי.
- במידה ולסעיף ניתנה תשובה ובנוסף נרשם לגבי הסעיף "לא יודע/ת" אזי הניקוד שיינתן לסעיף יהיה 0 מבלי שהתשובה תיקרא.
- אם לא רשום דבר בסעיף או שזה כלל לא נמצא ההנחה היא שנרשם "לא יודע/ת" עבור אותו סעיף.

שאלה 1 (10 נקודות):

נתונה תמונה (original) ונתונה תוצאה של הפעלת קונבולוציה עם מסיכת טשטוש על גבי התמונה המקורית (התוצאה היא תמונת filtered). בתמונת התוצאה ניתן לראות קוים מקבילים לצירי התמונה אשר לא היו קיימים בתמונה המקורית:



א. (5 נקודות) המסכה היא בגודל 5×5 כתוב את המסכה הנ"ל.
ב. (5 נקודות) הצע מסיכה אחרת שלמדנו עליה שלא תגרום לקוים המקבילים הללו להופיע. אין צורך לכתוב את המסכה אלא רק לתת את שמה.

שאלה 2 (20 נקודות):

בתרגיל קיבלתם את הקוד הבא אשר מסובב תמונה:

```
1 function imr = myImRotate(im, theta)
2 % imr = myImRotate(im, theta)
3 % This function rotates image im by angle degrees in a
4 % counterclockwise direction around its center point.
5 % It might creates black borders.
6
7 theta= degtorad(theta);
8
9 if (~ismatrix(im))
10     error('im should be a grayscale image (matrix)');
11 end
12
```

```

13 Y= size(im,1);
14 X= size(im,2);
15 diagonal_length= sqrt( (Y-1)^2 + (X-1)^2 );
16 imr= zeros(ceil(diagonal_length), ceil(diagonal_length));
17
18 Yr= size(imr,1);
19 Xr= size(imr,2);
20 mid_point= diagonal_length/2;
21
22 inv_rotation_matrix= [
23     cos(theta) -sin(theta);...
24     sin(theta) cos(theta)];
25
26 for yr= 1:Yr
27     for xr= 1:Xr
28
29         %
30         yr_t= yr-mid_point;
31         xr_t= xr-mid_point;
32
33         % inv rotation & translation
34         [xr_yr_t_invr_vec]= inv_rotation_matrix*[yr_t; xr_t];
35         yr_t_invr= xr_yr_t_invr_vec(1)+mid_point-((Yr-Y)/2);
36         xr_t_invr= xr_yr_t_invr_vec(2)+mid_point-((Xr-X)/2);
37         y_u= floor(yr_t_invr);
38         y_d= ceil(yr_t_invr);
39         x_d= floor(xr_t_invr);
40         x_u= ceil(xr_t_invr);
41
42         % compute new grayvalues from 4 bounding points
43         if ( (y_u>=1)&&(y_u<=Y) && ...
44             (y_d>=1)&&(y_d<=Y) && ...
45             (x_u>=1)&&(x_u<=X) && ...
46             (x_d>=1)&&(x_d<=X) )
47             % y_u  a    b
48             % y    *
49             % y_d  c    d
50             %   x_d x  x_u
51             ag= im(y_u,x_d);
52             bg= im(y_u,x_u);
53             cg= im(y_d,x_d);
54             dg= im(y_d,x_u);
55             imr(yr,xr)= 0.25*ag + 0.25*bg + 0.25*cg + 0.25*dg;
56         end

```

57 end

58

59 end

א. (10 נקודות) הסבר לשם מה צריך את שורות 43-46.
ב. (10 נקודות) תאר בעייה שראיתם בתרגיל שנוצרת לאחר מספר סיבובים של תמונה שהיא לא בעיית הטשטוש ואשר ניתנת לשיפור על ידי שינוי בשורות 36-43 (כולל תוספת שורות). תאר את השינוי (במילים אין צורך בקוד).

שאלה 3 (20 נקודות):

הסבר בקצרה 2 סיבות לשימוש במרחקי בסיס thresholded במרחק ה Earth Mover's Distance עבור 2 SIFTs. תזכורות: הגדרת מרחק ה EMD:

$$\widehat{EMD}_\alpha(P, Q) = \left(\min_{\{f_{ij}\}} \sum_{i,j} f_{ij} d_{ij} \right) + \left| \sum_i P_i - \sum_j Q_j \right| \times \alpha \max_{i,j} \{d_{ij}\}$$

s.t :

$$\sum_j f_{ij} \leq P_i, \quad \sum_i f_{ij} \leq Q_j, \quad \sum_{i,j} f_{ij} = \min\left(\sum_i P_i, \sum_j Q_j\right), \quad f_{ij} \geq 0$$

SIFT הוא היסטוגרמה של זוויות במיקומים מרחביים:

.	*	-	←
↖	↗	↘	↙
↗	←	↘	↖
↓	↑	↗	↖

אם j, i הם אינדקסים של SIFT ו $t(i)$, $x(i)$, $y(i)$ מתארים את הזווית, מיקום x ומיקום y של האינדקס הנ"ל ב SIFT בהתאמה, 8 הוא מספר הזוויות בכל "תא" מרחבי (יש 16 "תאים" מרחביים בצירור למעלה) אזי דוגמא למרחק בסיס thresholded (עם $\text{threshold}=3$) הוא:

$$d_{ij} = \min(\min(|t(i) - t(j)|, 8 - |t(i) - t(j)|) + |x(i) - x(j)| + |y(i) - y(j)|, 3)$$

שאלה 4 (50 נקודות):

להלן ה loss של פונקציית הרגרסיה "Least Absolute Deviations" עם רגולרייזר L1:

$$\sum_{j=1}^d |w_j| + C \sum_{i=1}^n |\vec{w}^T \vec{x}_i - y_i|$$

א. (20 נקודות) כתבו פסאודו קוד של אלגוריתם שמוצא את הוקטור \vec{w} האופטימלי ל loss הנ"ל בהינתן הקלט להלן ע"י $\text{stochastic subgradient}$ decent. הקלט:

$$\{\vec{x}_i, y_i\}_{i=1}^n \quad \vec{x}_i \in \mathbb{R}^d, y_i \in \mathbb{R}$$

ב. (20 נקודות) בשיעור ראיתם כיצד אפשר לכתוב משוואות loss ללא שימוש בפונקציית max או ערך מוחלט אלא אך ורק ע"י שימוש במשתני עזר ומגבלות לינאריות על פונקציה לינארית. הראו כיצד אפשר לעשות זאת במקרה של פונקציית הרגרסיה "Least Absolute Deviations" עם רגולרייזר L1.
 ג. (10 נקודות) נתונה לכם פונקצית מטלב:

`[v]= solve_lin_prog(a, A,b)`

`% returns the solution to the optimization problem:`

`% v= argmin_v a'v s.t: A*v>=b`

כלומר, מתמטית היא פותרת את הבעיה:

$$\arg \min_{\vec{v}} \vec{a}^T \vec{v} \quad \text{s.t:} \quad A\vec{v} \geq \vec{b}$$

כתבו את המטריצה A והוקטורים a,b שיינתנו לפונקציה `solve_lin_prog`

על מנת למצוא את הוקטור \vec{w}

האופטימלי ל loss של פונקציית הרגרסיה "Least Absolute Deviations" עם רגולרייזר L1.

בהצלחה!!!