

Assignment 2:

Stereo Matching

Due Date: 19/12/2016

רקע

נתונות זוג תמונות (image_left.png, image_right.png) המציגות את אותה הסצנה, עם הזזה אופקית בין התמונות. נרצה לבנות את תמונת העומק על מנת לשערך את המבנה התלת-מימדי של התמונה. במהלך התרגיל תחילה נמצא תמונת עומק פשוטה באמצעות SSD, ובהמשך ננסה לשפר אותה באמצעות טכניקות החלקה שונות.

הסבר על הקבצים המצורפים:

- :run_script.m

סקריפט ההרצה הראשי. מפעיל את פונקציית compute_all פעם אחת עבור הקבצים המצורפים, ופעם שניה עבור הקבצים שלכם.

- :compute_all.m

פונקציית העל. טוענת את התמונות וקובץ הפרמטרים, ומריצה את כל סעיפי התרגיל הבית עליהן. הפונקציה עובדת באופן אינקרמנטלי, ומציגה שלב אחרי שלב את תוצאות החלקים בתרגיל (מומלץ להשתמש בה להצגת התוצרים גם טרם מומשו כל הפונקציות).

הערה:

באפשרותכם לערוך ולשנות את הקבצים הנ"ל, אולם נדרש כי בפתרון שתגישו ניתן יהיה להריץ את הגרסה המקורית של הקבצים, והם יספקו תוצאות נכונות.

- :data_creator.m

פונקציית עזר ליצירת קובץ פרמטרים, ויצירת קבצי תמונה. השימוש בפונק' זו לא מחייב.

שימו לב: הבהרות ותשובות לשאלות נפוצות יעודכנו באתר ה Moodle.

חלק א': חישוב מטריצת המרחקים

בשלב הראשון של התרגיל נחשב את מטריצת המרחקים, אשר תשמש אותנו בשאר חלקי התרגיל. נתחיל בהמחשה פשוטה.

נניח שנתונות לנו שתי התמונות הבאות, וכן נתון כי הן עברו רקטיפיקציה (ולכן נדרש לחפש התאמה רק לאורך ציר ה-x):

Left					Right				
1	2	3	4	5	1	1	1	1	1
6	7	8	9	10	2	2	2	2	2
11	12	13	14	15	3	3	3	3	3
16	17	18	19	20	4	4	4	4	4

מגדירים את הפרמטרים הבאים (הערכים לדוגמא):

$dsp_range = 2$ הוא תחום ה-disparity שיש לסרוק, כלומר הערכים $[-2:2]$ סביב כל פיקסל (סה"כ 5 ערכים).
 $win_size = 3$ הוא גודל החלון בפיקסלים סביב הפיקסל הנבדק (סה"כ 9 פיקסלים בחלון).

מטריצת המרחקים $ssdd$ תציין את מרחק ה-SSD (Sum Square Diff) לכל פיקסל לכל ערך disparity. לפיכך, גודל המטריצה שתתקבל יהיה $4 \times 5 \times 5$ (כאשר 4×5 נובע מגודל התמונות, והגודל 5 במימד השלישי עקב 5 ערכי disparity אפשריים).

חישוב לדוגמא: הערך $ssdd(2,3,2)$ – שורה 2, עמודה 3, וה-disparity הוא 1.

Left					Right				
1	2	3	4	5	1	1	1	1	1
6	7	8	9	10	2	2	2	2	2
11	12	13	14	15	3	3	3	3	3
16	17	18	19	20	4	4	4	4	4

ניקח את תמונה Left, נלך לפיקסל בשורה 2 עמודה 3 ונסמן חלון בגודל 3×3 סביבו.

מכיוון שה-disparity הוא 1- אז החלון המתאים בתמונה Right הוא סביב הפיקסל בשורה 2 עמודה 2.

ההפרשים המתקבלים:

$$ssdd(2,3,2) = \begin{array}{|c|c|c|} \hline 2-1 & 3-1 & 4-1 \\ \hline 7-2 & 8-2 & 9-2 \\ \hline 12-3 & 13-3 & 14-3 \\ \hline \end{array} \cdot 2 = 1+4+9+25+36+49+81+100+121 = 426$$

1. חשבו באופן דומה את $ssdd(2,3,3)$, $ssdd(2,3,4)$ וכן את $ssdd(3,4,1)$, $ssdd(3,4,2)$.

2. בנו פונקציה המחשבת את מטריצה מרחקי ה-SSD בין התמונות המתקבלות, באופן הדומה לדוגמה שהומחשה לעיל.

השתמשו ב-API הבא:

ssdd = ssd_distance (im_left, im_right, win_size, dsp_range)

קלט:

- im_left התמונה המצולמת במצלמה השמאלית, מטריצה מסוג double.
- im_right התמונה המצולמת במצלמה הימנית, מטריצה מסוג double.
- win_size גודל החלון, מספר שלם אי-זוגי.
- dsp_range מייצג כחצי מתחום ה-disparity, מספר שלם.
- כלומר תחום ה-disparity שעליו נחפש הוא [-dsp_range:dsp_range].

פלט:

- ssdd סכום ההפרשים הריבועיים לכל פיקסל בחלון בגודל win_size X win_size סביבו, עבור $2 \times \text{dsp_range} + 1$ ערכי ה-disparity האפשריים.
- התוצאה הינה מטריצה תלת מימדית בגודל nRows X nCols X nLabels, כאשר nRows ו-nCols הוא מספר השורות והעמודות ב-im_left בהתאמה, ו-nLabels מייצג את מספר ערכי ה-disparity האפשריים, קרי: $nLabels = 2 \times \text{dsp_range} + 1$.

הערה: הערכים win_size ו-dsp_range נטענים מתוך הקובץ params.mat. באפשרותכם לעדכן ערכים אלה על ידי עדכון הערכים השמורים בקובץ. במידה ועשיתם זאת, רשמו במסמך הפתרון את הערכים שקבעתם. יש לוודא כי הערכים בהם השתמשתם קבועים לאורך כל התרגיל.

שימו לב: מכיוון שהתמונות עברו רקטיפיקציה, אנחנו נדרשים לחפש התאמה רק לאורך ציר ה-x.

שימו לב 2: יש לנרמל את המטריצה כך שהמחיר המינימאלי בה יהיה 0 והמקסימאלי 255.

חלק ב': תמונת עומק נאיבית (ללא החלקה)

נרצה להציג את תמונת העומק המתקבלת ישירות ממטריצת המרחקים.

3. בנו פונקציה הבונה ממטריצת מרחקי ה-SSD תמונת עומק ללא החלקה, על ידי בחירת העומק המיטבי בכל פיקסל.

השתמשו ב-API הבא:

label_no_smooth = naive_labeling(ssdd)

קלט:

Ssdd – מטריצת המרחקים כפי שחישבתם ב-ssdd_distance.

פלט:

label_no_smooth – מטריצה בגודל nRows X nCols (מספר השורות והעמודות ב-ssdd) המכילה את ההשמה המיטבית המתחשבת במרחק ה-SSD בלבד.

4. נתחו את התוצאה (בקצרה), ותארו מהן הבעיות המתקבלות משימוש ב-SSD בדרך זו והסיבות לכך.

טיפ: ניתן לראות את איכות התוצאות גם על ידי ביצוע forward mapping של התמונה השמאלית לפי מטריצת ה-disparity (קרי, להזיז פיקסלים של התמונה השמאלית בציר ה-x, כשההזזה היא ערך ה-disparity בכל פיקסל), ולבדוק כמה התמונה המתקבלת דומה לתמונה הימנית - אבל לא חובה.

חלק ג': החלקת תמונת עומק – שימוש ב-Dynamic Programming

כעת נרצה לשפר את תמונת העומקים על ידי שימוש ב-Dynamic Programming.

נגדיר פונקציית ציון אשר עובדת על פרוסות אופקיות של מטריצת המרחקים (שורה יחידה), כאשר המימד של כל פרוסה הוא $1 \times nCols \times nLabels$, וכדי לפשט, נסתכל עליה במימדים $nLabels \times nCols$, כאשר $nLabels$ הוא מספר ערכי ה-disparity השונים אותם אנו בודקים, ו- $nCols$ הוא אורך השורה.

(פונקציית הציון/עלות שלנו תחשב למעשה את כמות/עלות השינוי בעומק של הפיקסל ביחס לסביבתו.)

5. ממשו פונקציית ציון לפרוסה של מטריצת ה-ssdd, על ידי שימוש ב-Dynamic Programming, כאשר הציון בכל ערך נתון על ידי:

$$L(d, col) = C(d, col) + M(d, col) - \min\{L(:, col - 1)\}$$

כאשר d ו- col הם מספרי השורה והעמודה בפרוסה, בהתאמה. שימו לב כי "מנרמלים" את ערכי העמודה הנוכחית ביחס לערך המינימאלי בעמודה הקודמת, על מנת לשמור על ערכים אחידים לאורך כל העמודות. $M(d, col)$ מציין את מחיר המסלול האופטימאלי עד למיקום הנוכחי, ונבחר להיות **המינימום** בין הערכים הבאים:

(א) ציון המסלול L בעמודה הקודמת עבור אותו ערך של d : $L(d, col - 1)$
(ב) ציון המסלול L המיטבי בעמודה הקודמת עבור ערך d גדול/קטן ב-1, ועוד "קנס" במחיר $P1$:

$$P1 + \min\{L(d - 1, col - 1), L(d + 1, col - 1)\}$$

(ג) ציון המסלול L המיטבי בעמודה הקודמת עבור ערך d רחוק ב-2 **ומעלה**, ועוד "קנס" במחיר $P2$.

$$P2 + \min\{L(d + k, col - 1) \forall k: |k| \geq 2\}$$

(ניתן לנרמל את $P2$ בערך הגרדיאנט באותה הנקודה).

השתמשו ב-API הבא:

$$L_slice = dp_grade_slice(C_slice, P1, P2)$$

קלט:

C_slice	– פרוסת מחירים/מרחקים (כרגע, פרוסה אופקית של מטריצת ssdd, כמתואר לעיל). מטריצה בגודל $nLabels \times nCols$
$P1$	– פרמטר המציין "קנס" על מעבר בין ערכי d סמוכים.
$P2$	– פרמטר המציין "קנס" על מעבר בין ערכי d מרוחקים.

פלט:

L_slice	– פרוסת ציונים. מטריצה בגודל L (זהו $nLabels \times nCols$) המציינת את הציון לכל מיקום וערך d את ציון המסלול הטוב ביותר.
------------	---

6. ממשו פונקציה המקבלת מטריצת מרחקים מלאה (ssdd), ומוציאה תמונת עומקים מתוכה, המשתמשת ב-Dynamic Programming.

בצעו זאת על ידי הפעלת פונקציית `dp_grade_slice` מהסעיף הקודם על כל אחת מהפרוסות במטריצת המרחקים המלאה, לקבלת מטריצת מחירים מלאה L (בגודל זהה לגודל מטריצת ssdd).

לבסוף, בחרו עבור כל פיקסל במטריצה L (קרי, עבור כל שורה ועמודה) את ערך ה-disparity המיטבי. לבחירת ערך ה-disparity המיטבי בכל נקודה, ניתן פשוט לבחור את הערך שעבורו L הוא מינימלי באותו הפיקסל (אין חובה לבצע back-tracking ומציאת המסלול הטוב ביותר).

השתמשו ב-API הבא:

`label_smooth_dp = dp_labeling(C, P1, P2)`

קלט:

- C – מטריצת מחירים/מרחקים (כרגע, כל מטריצת ssdd).
מטריצה בגודל $nRows \times nCols \times nLabels$.
- $P1$ – פרמטר המציין "קנס" על מעבר בין ערכי d סמוכים.
- $P2$ – פרמטר המציין "קנס" על מעבר בין ערכי d מרוחקים.

פלט:

- `label_smooth_dp` – מטריצה בגודל $nRows \times nCols$ (מספר השורות והעמודות ב- C) המכילה את ההשמה המיטבית המתקבלת באמצעות שיטת Dynamic Programming.

הערה: הערכים $P1$ ו- $P2$ נטענים מתוך הקובץ `params.mat`. באפשרותכם לעדכן ערכים אלה על ידי עדכון הערכים השמורים בקובץ. במידה ועשיתם זאת, רשמו במסמך הפתרון את הערכים שקבעתם. יש לוודא כי הערכים בהם השתמשתם קבועים לאורך כל התרגיל.

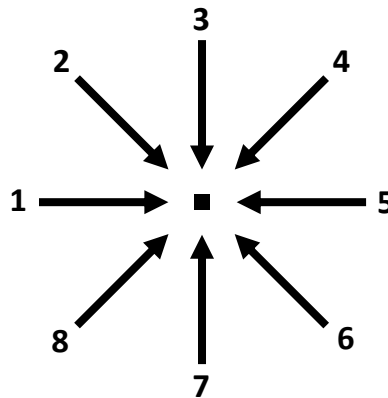
7. דונו בתוצאות שקיבלתם והשוו אותם לתוצאות שהתקבלו בשיטה הנאיבית. צרפו את תמונות התוצאה למסמך הפתרון.

טיפ: ניתן לראות את איכות התוצאות גם על ידי ביצוע `forward mapping` של התמונה השמאלית לפי מטריצת ה-disparity (קרי, להזיז פיקסלים של התמונה השמאלית בציר ה- x , כשההזזה היא ערך ה-disparity בכל פיקסל), ולבדוק כמה התמונה המתקבלת דומה לתמונה הימנית.

חלק ד': החלקת תמונת עומק – שימוש ב-Semi-Global Mapping

כעת נשפר את החלקת התמונה עוד יותר על ידי שימוש בשיטת Semi-Global Mapping. שיטה זו מרחיבה את שיטת Dynamic Programming על ידי מתן ציון לכל ערך disparity על ידי שקלול מסלולים מיטביים ממספר כיוונים (במקום שימוש בכיוון התקדמות יחיד כפי שעשינו ב-DP).

8. ממשו פונקציה המוציאה פרוסות מתוך מטריצת מרחקים תלת מימדית, בהתאם לכיוון התקדמות המתקבל כקלט.
נגדיר 8 כיוונים באופן הבא:



כך למשל, כיוון 1 הוא הכיוון האופקי שהשתמשנו בו בחלק הקודם, וכיוון 5 הוא כיוון אופקי, עם סדר עמודות הפוך.
שימו לב כי בכיוונים האלכסוניים, אין צורך לבצע אינטרפולציות למיניהן על ערכים שנפלים בין פיקסלים, אלא לעבור רק על הפיקסלים עצמם.
טיפ: היעזרו בסימטריות כדי לקצר לעצמכם עבודה.

9. עדכנו את פונקציית ציון הפרוסה `dp_grade_slice` כך שתתמודד עם פרוסות באורכים קצרים יותר (צפויות פרוסות קצרות במסלולים האלכסוניים, כולל פרוסות באורך 1).

10. ממשו פונקציה המקבלת מטריצת מרחקים מלאה (ssdd), ומוציאה תמונת עומקים מתוכה, המשתמשת ב-Semi-Global Mapping.

בצעו זאת על ידי קבלת מטריצת ציונים L_1, \dots, L_8 באמצעות הפעלת Dynamic Programming על פני כל אחד מהמסלולים 1-8, ולבסוף בניית מטריצה L על ידי מיצוע המטריצות L_1, \dots, L_8 . היעזרו בפונקציות `dp_grade_slice` והפונקציה מסעיף 8.

לבסוף, בחרו עבור כל פיקסל במטריצה L (קרי, עבור כל שורה ועמודה) את ערך ה-disparity המיטבי. לבחירת ערך ה-disparity המיטבי בכל נקודה, ניתן פשוט לבחור את הערך שעבורו L הוא מינימאלי באותו הפיקסל (אין חובה לבצע back-tracking ומציאת המסלול הטוב ביותר).

השתמשו ב-API הבא:

label_smooth_sgm = sgm_labeling(C, P1, P2)

קלט:

- C – מטריצת מחירים/מרחקים (כרגע, כל מטריצת ssdd).
מטריצה בגודל $nRows \times nCols \times nLabels$.
- P1 – פרמטר המציין "קנס" על מעבר בין ערכי d סמוכים.
- P2 – פרמטר המציין "קנס" על מעבר בין ערכי d מרוחקים.

פלט:

- label_smooth_sgm – מטריצה בגודל $nRows \times nCols$ (מספר השורות והעמודות ב-C) המכילה את ההשמה המיטבית המתקבלת באמצעות שיטת Semi-Global Mapping.

הערה: הערכים P1 ו-P2 נטענים מתוך הקובץ `params.mat`. באפשרותכם לעדכן ערכים אלה על ידי עדכון הערכים השמורים בקובץ. במידה ועשיתם זאת, רשמו במסמך הפתרון את הערכים שקבעתם. יש לוודא כי הערכים בהם השתמשתם קבועים לאורך כל התרגיל.

11. דונו בתוצאות שקיבלתם והשוו אותם לתוצאות שהתקבלו בשיטות הקודמות. צרפו את תמונות התוצאה למסמך הפתרון.

טיפ: ניתן לראות את איכות התוצאות גם על ידי ביצוע forward mapping של התמונה השמאלית לפי מטריצת ה-disparity (קרי, להזיז פיקסלים של התמונה השמאלית בציר ה-x, כשההזזה היא ערך ה-disparity בכל פיקסל), ולבדוק כמה התמונה המתקבלת דומה לתמונה הימנית.

חלק ה': בדיקה על תמונה נוספת

12. השתמשו בזוג תמונות שלכם (ודאו שהתמונות עברו רקטיפיקציה), והריצו את הקוד המלא על תמונות אלו. מומלץ להשתמש בתמונות מהאתר <http://vision.middlebury.edu/stereo/data> על מנת שהסקריפט יעבוד, יש לקרוא לקובץ התמונה השמאלית "my_image_left.png", ולקובץ התמונה הימנית "my_image_right.png". שימו לב כי לשם כך יש להמיר את התמונות לפורמט PNG (טעינת תמונה ל-MATLAB ושמירתה על ידי `imwrite` לקובץ בסיומת `png` יעשה את העבודה).

כמו כן, יש ליצור קובץ פרמטרים בשם "my_params.mat" אשר יכיל את הערכים שלכם לפרמטרים השונים (ניתן לראות את הפרמטרים הנדרשים ב-"params.mat").

ניתן להיעזר בקובץ העזר `data_creator.m` ליצירת הקבצים הנ"ל.

סעיפי בונוס:

13. הציעו וממשו שיטת ציון שונה לבניית מטריצת המרחקים, אשר תחליף את ה-SSD. צרו קובץ הרצה חלופי ל-`run_script.m` אשר יריץ את כל השלבים בשימוש במטריצת המרחקים החדשה. השוו בין התוצאות שקיבלתם בתרגיל לתוצאות החדשות ודונו בהבדלים.
14. הציעו וממשו אלגוריתם חלופי להחלקת תמונת העומק (ניתן להשתמש ב-SSD או בשיטה שמומשה בשאלה הקודמת), אשר תחליף את השיטות שהוצגו בתרגיל. צרו קובץ הרצה חלופי ל-`run_script.m` אשר יריץ את שיטת ההחלקה החדשה. השוו בין התוצאות שקיבלתם בתרגיל לתוצאות החדשות ודונו בהבדלים.

הוראות הגשה:

- ההגשה בזוגות.
- יש להגיש מסמך המכיל התייחסות לכלל הסעיפים בתרגיל, המציג את כל התוצאות ועונה על כל השאלות.
- יש לצרף את כל פונקציות ה-MATLAB שהוגדר להן API בתרגיל, וכן כל הפונקציות הנלוות אליהן שכתבתם. בדיקת התרגיל תכיל הרצה אוטומטית של פונקציות אלו ובדיקתן.
- יש לצרף את זוג התמונות שלכם (`my_image_left.png`, `my_image_right.png`) וכן קובצי פרמטרים מעודכנים (`params.mat`, `my_params.mat`).
- את כל הקבצים המצורפים יש לכווץ ל-`zip`.
- יש לבדוק שאתם מצליחים להריץ את `run_script.m` המצורף, ללא ביצוע שינויים בו, וכאשר תיקיית העבודה של MATLAB היא התיקייה בה נמצא הקובץ.
- בכדי לא לאבד (הרבה) נקודות, מומלץ לבדוק שהקוד שכתבתם פועל גם על מחשב בלתי תלוי במחשב בו כתבתם אותו (למשל במעבדות באוניברסיטה), ולוודא שהרצה של כל אחת מהפונקציות מניבה תוצאה רצויה.
- את הפתרון עם כל הקבצים הרלוונטיים (`zip`) יש להגיש למייל: ATICV2016@Gmail.com
את הנושא של המייל יש לנסח באופן הבא:
Assignment #2 ID:your_id_number#1- your_id_number#2
- איחור בהגשה או בזוגות יגרור הורדה בציון.

בהצלחה!