



עבודה מספר 2 בקורס עיבוד תמונות רפואיות 67705

נושא: רגיסטרציה

פרופ' ליאו יוסקוביץ

אחראי לתרגיל: רפאל ויונטי

מועד הגשה: 12.1.15

מבוא: באופן כללי, בין שתי סריקות של אותו פציינט בזמנים שונים אין התאמה טובה. בין שני מועדי הסריקה הפציינט זז. למרות זאת, לעיתים קרובות נרצה שהתמונות יהיו בהתאמה מושלמת לצרכי השוואה. אם האיבר שמעניין אותנו הוא קשיח, כמו המוח למשל, אזי ניתן למצוא את התנועה הזו בעזרת רגיסטרציה ריגידית.

רגיסטרציה ריגידית היא תהליך של מציאת סיבוב והזזה בין זוג תמונות. בתמונות דו ממדיות רגיסטרציה ריגידית מאופיינת בשלושה פרמטרים: שני פרמטרי הזזה, וזווית סיבוב. בתמונות תלת מימדיות, כמו MRI או CT, רגיסטרציה ריגידית מאופיינת בשישה פרמטרים: שלושה פרמטרי הזזה, ושלוש זוויות סיבוב. סט של הזזות וסיבובים כאלו נקרא "טרנספורמציה". בתהליך רגיסטרציה אנו מחפשים את הטרנספורמציה בין שתי תמונות, ולעיתים אנו נדרשים גם להפעיל את הטרנספורמציה על אחת התמונות על מנת ליישר אותה כלפי התמונה השנייה.

אלגוריתמים למציאת רגיסטרציה ריגידית מתחלקים באופן כללי לשני סוגים: מבוססי תמונה, או מבוססי נקודות עניין. באלגוריתמים מבוססי נקודות עניין אנו תחילה מזוהים אזורים קטנים בשתי התמונות שמתאימים האחד לשני, כלומר מייצגים כנראה את אותה נקודה באיבר הנסרק. ברגע שיש לנו מספיק התאמות כאלה ניתן לחשב את הטרנספורמציה. באלגוריתמים מבוססי תמונה עושים ההיפך: תחילה מנחשים טרנספורמציות כלשהן, מפעילים אותן על התמונות (כלומר מיישרים תמונה אחת כלפי השנייה), ואז מודדים את הדמיון בין התמונות אחרי היישור, ובוחרים את הטרנספורמציה בעלת הדמיון החזק ביותר.

1. אלגוריתמים מבוססי נקודות עניין (15%)

בחלק זה אנו נמצא רגיסטרציה ריגידית בין שתי תמונות אופטיות **דו ממדיות**.

רקע: באלגוריתמים אלו מוצאים תחילה 'התאמות' שהן נקודות דו מימדיות שאנו יודעים היכן הן נמצאות בשתי התמונות. לאחר מכן, מתוך ההתאמות, מחשבים את הרגיסטרציה הריגידית.

קחו את התמונות brain1.tif, brain2.tif. מיצאו לפחות 10 התאמות בין התמונות, רישמו את המיקומים שלהן בשתי התמונות.

בעזרת המיקומים, מיצאו את הסיבוב סביב מרכז התמונה (במעלות) ואת ההזזה (בפיקסלים) בין שתי התמונות מתוך הנקודות.

הזיזו וסובבו את תמונה 2 כך שתתאים לתמונה 1.

הזיזו וסובבו באותו האופן את brain2_seg.tif והציגו אותה על brain1.tif

2. אלגוריתמים מבוססי תמונה (75%)

בחלק זה אנו נמצא רגיסטרציה ריגידית בין סריקות תלת ממדיות.

כפי שכתבנו לעיל, באלגוריתמים מבוססי תמונה מנחשים תחילה טרנספורמציות כלשהן, מפעילים אותן על התמונות (כלומר מיישרים תמונה אחת כלפי השנייה), ואז מודדים את הדמיון בין התמונות אחרי היישור, ובחרים את הטרנספורמציה בעלת הדמיון החזק ביותר. חלק חשוב בתיאור האלגוריתם הוא נתינת ציון דמיון בין שתי תמונות.

בתחום עיבוד התמונה נפוצים ציונים ומרחקים רבים, וביניהם NCC, SSD, SAD, MI ועוד. בסעיף זה נממש ונבדוק את שני הציונים הבאים:

- Normalized Cross Correlation (NCC)

- Mutual Information (MI)

הציון NCC משמש במקרים שהתמונות דומות בערכי האפור שלהן, למשל שתי סריקות באותו פרוטוקול סריקה. הציון MI משמש במקרים שהתמונות דומות במבנה שלהן אבל שונות מאוד בערכי האפור, למשל שתי סריקות MRI, האחת בפרוטוקול T1 והשניה בפרוטוקול T2.

א. (10%) ממשו במטלב את שתי הפונקציות הבאות, לפי מה שלמדתם בשיעור, או מויקיפדיה:

Function grade = Normalized_Cross_Correlation(image1, image2)

Function grade = Mutual_Information(image1, image2)

ב. (10%) ממשו במטלב את הפונקציה

Function shift = Find_Shift(image1, image2, methodNum, dimNum)

שמוצאת את ההזזה לאורך הציר dimNum, בעזרת מדד הדמיון methodNum, בין שתי תמונות באותו הגודל כאשר:

- התוצאה shift היא בכמה פיקסלים שלמים יש להזיז את התמונה image1 כדי שתתאים לתמונה image2

- dimNum יקבל 1 עבור ציר הX, 2 עבור ציר הY או 3 עבור ציר הZ

- methodNum יקבל 1 עבור NCC או 2 עבור MI.

ג. (10%) השתמשו בפונקציות Find_Shift ו Mutual_Information:

a. מוצאו באיזה ציר היתה ההזזה, ואת גודל ההזזה, בין הסריקה התלת מימדית brainMRI2.nii ל brainMRI1.nii. שימו לב שזוהי 'דוגמת צעצוע' בה ידוע שההזזה בין התמונות היא רק לאורך ציר ראשי אחד מבין השלושה, ואין סיבוב כלל.

b. הזיזו את תמונת הסגמנטציה brainMRI2_seg.nii באותו אופן, והציגו אותה על brainMRI1.nii.

ד. (10%) ממשו במטלב את הפונקציה

Function rot = Find_Rotation(image1,image2,methodNum,dimNum)

שמוצאת את הסיבוב במעלות סביב הציר dimNum, בעזרת מדד הדמיון methodNum, בין שתי תמונות באותו הגודל כאשר:

- התוצאה rot היא בכמה מעלות יש לסובב את התמונה image1 סביב הציר dimNum, כדי שתתאים לתמונה image2.
- dimNum יקבל 1 עבור ציר הX, 2 עבור ציר הY או 3 עבור ציר הZ
- הסיבוב הוא סביב מרכז התמונה.
- methodNum יקבל 1 עבור NCC או 2 עבור MI.

ה. (15%) השתמשו בפונקציות Find_Rotation ו Normalized_Cross_Correlation:

- a. מיצאו סביב איזה ציר היה הסיבוב, ואת גודל הסיבוב במעלות, בין הסריקה התלת מימדית brainMRI3.nii ל brainMRI1.nii. שימו לב שזוהי 'דוגמת צעצוע' בה ידוע שהסיבוב בין התמונות הוא רק סביב ציר ראשי אחד מבין השלושה, ואין הזזה של מרכז התמונה.
- b. סובבו את תמונת הסגמנטציה brainMRI3_seg.nii באותו אופן, והציגו אותה על brainMRI1.nii.

ו. (10%) מיצאו את מטריצת ההיסטוגרמה המשותפת של brainMRI1.nii ו brainMRI2.nii אחרי רגיסטרציה. מטריצת ההיסטוגרמה המשותפת היא שלב ביניים בחישוב ציון הMI. הציגו אותה כתמונה, וענו:

איך המטריצה נראית?

האם ניתן לתרגם, בעזרת המטריצה, את הצבעים של תמונה 1 כך שיהיו זהים לאלו של תמונה 3?

מדוע זה כך?

ז. (10%) מיצאו את מטריצת ההיסטוגרמה המשותפת של brainMRI1.nii ו brainMRI3.nii אחרי רגיסטרציה. הציגו אותה כתמונה, וענו:

איך המטריצה נראית?

האם ניתן לתרגם, בעזרת המטריצה, את הצבעים של תמונה 1 כך שיהיו זהים לאלו של תמונה 2?


מדוע זה כך?

מדוע קבעו את שני הפרוטוקולים T1 ו T2 כך שההיסטוגרמה המשותפת של תמונות ברגיסטרציה מושלמת נראית כך?

הנחיות

- הפתרון צריך לכלול, בקבצים נפרדים, את פונקציות המטלב:
Mutual_Information.m, Find_Shift.m, Find_Rotation.m , Normalized_Cross_Correlation.m
- סקריפט Ex2.m מטלב שיוצר באופן אוטומטי את התשובות לכל השאלות הנ"ל. את הנקודות שמצאתם באופן ידני הכניסו לסקריפט עצמו, כך שתוכלו גם לענות אוטומטית על שאלה 1.
- כל קבצי קוד אחרים/פונקציות עזר הנדרשים להפעלת הנ"ל
- זמן ריצה סביר הוא כ-5 דקות ב-PC סטנדרטי לכל סעיף, אך כמובן שהן אמורות להיות מהירות בהרבה. ינתן ניקוד על יעילות ומהירות הקוד.
- מסמך ושמו Ex2.docx עם:
 - שם המגיש, login, ות.ז.
 - תיאור הפתרון המוצע לכל פונקציה שמומשה, ופונקציית עזר.
 - התשובות לשאלות שבתרגיל.
 - צילומי מסך וגרפים שידגימו את התוצאות שלכם ואת דרכי הפיתרון.
- חלק מהציון יהיה על תיעוד, הגשת קוד מסודרת ומסמך ברור. (10%)

רמזים

- הפונקציה imrotate במטלב מסובבת גם תמונות תלת מימדיות, אך רק סביב ציר אחד.
- פונקציות עזר נוספות במטלב: permute, sparse, imshow, rgb2gray.
- כפתור  במטלב נותן קורדינטות מדוייקות של דקירות העכבר בתמונות דו ממדיות.