

**פרויקט מסכם - קורס עיבוד תמונה**

תוכן עניינים

[מבוא 3](#_Toc443736575)

[מטרת הפרוייקט 3](#_Toc443736576)

[הנחות בסיס והגדרות 3](#_Toc443736577)

[חייכן תקין 3](#_Toc443736578)

[אלגוריתם 4](#_Toc443736579)

[כללי 4](#_Toc443736580)

[פירוט עיקרי האלגוריתם 4](#_Toc443736581)

[קלט פלט 7](#_Toc443736582)

[הוראות הפעלה 7](#_Toc443736583)

[סיכום ומסקנות 11](#_Toc443736584)

# מבוא

הרבה פעמים אנחנו נתקלים במהלך הגלישה שלנו באינטרנט, במסמך איכותי שנרצה לעשות בו שימוש כזה או אחר, לדוגמא:

* כאשר אנו קוראים מצגת באנגלית עבור הלימודים, ואנו נתקלים בשקופית לא ברורה כל כך, ע"מ לתרגם אותה עלינו להעתיק למילון מקוון את הטקסט של השקופית. הרבה פעמים השקופית אינה ניתנת לעריכה טקסטואלית שכן מדובר בקובץ תמונה שבתוכו טקסט, וללא אפליקציה ייעודית– נוכל רק להעתיק ידנית את השקופית למילון.
* כאשר נרצה לשכור דירה חדשה, נקבל חוזה כקובץ PDF שלא ניתן לערוך. בעזרת הרעיון שלנו נוכל לצלם מסך, להכניסו לקוד וכך לערוך את החוזה לדרישותינו.

לאחר שלמדנו את הנושאים המגוונים בקורס, החלטנו לנסות לזהות תמונות טקסט ולכתוב אותן בקובץ הניתן לעריכה. הנחנו הנחות מקלות, ולאחר שהאלגוריתם עבד בהצלחה עבורן, הורדנו חלק מההנחות וניסינו לשפר את הפרוייקט כך שיעבוד עבור מקרה כללי יותר. השתמשנו בהפיכת התמונה לבינארית , פקודות ועיצוב GUI כולל פונקצית save ובדיקה מעמיקה אודות יכולות MATLAB והצלחנו לכתוב , בזמן המוגבל שהיה לנו, קוד אופטימלי לרוב סוגי הקלטים העונים על ההנחות שלנו. נתקלנו בקשיים שעל חלקם התגברנו ועל חלקם פחות, אותם נפרט בהמשך.

# מטרת הפרויקט

ליצור קוד שיקבל כקלט תמונת טקסט (עם הנחות שעליהן נפרט בהמשך), ויוציא כפלט קובץ notepad שהמשתמש יבחר היכן לשמור על המחשב שלו, בתוך הקובץ יהיה הטקסט שהיה בתמונה.

מטרתינו היא להגיע לתוצאות מדוייקות בלפחות 70% מהמקרים שננסה.

אמנם נתבקשנו להניח מספר הנחות מקלות, אך החלטנו לאתגר את עצמנו בעיקר במשימה של קריאת קלט שמורכב באותיות בגדלים שונים.

# הנחות

**קלט תקין:**

1. קובץ תמונה.
2. כתוב בפונט arial.
3. הטקסט יהיה שחור על רקע לבן.
4. **עדיפות** ללא אות i וללא התו '!' , פונט גדול יביא תוצאות טובות יותר, ככל שהפונט קטן התוצאות נהיות פחות ופחות טובות.

**פלט תקין:**

1. ייתכן ונקבל\_\_\_\_ כאשר בתמונת הקלט יש את האות i או התו !

# האלגוריתם

1. הכנו קובץ PNG שמכיל בתוכו את התוים שאנו מעוניינים שהקוד שלנו יידע לזהות. התווים בגופן arial בגודל פונט 20 בערך (התמונה משנה את הגודל טיפה), וכוללים את כל האותיות הלועזיות (ראשיות וקטנות) ומספר סימני פיסוק, קובץ זה הינו הdatabase שלנו.
2. האלגוריתם שלנו הופך את התמונה לבינארית ואז קורא לפונקציה של matlab שלמדנו בכיתה - bwlabel אשר ממספרת את הצורות בתמונה (במקרה שלנו האותיות), לאחר מכן אנחנו קוראים לפונקציה שהגדרנו בשם comb אשר אחראית על חיבור (ידני) של צורות שלוגית הן אות אחת, למשל הפס והנקודה של ה i. בשלב זה משתמשים ב regionprops כדי למצוא את הbounding box של כל אות, שבעזרתו נוכל לשלוף מטריצה אישית לכל אות. מכיוון שאנו מניחים כי המטריצות של האותיות בתמונת הקלט יהיו אולי בגודל שונה, השתמשנו בפונקציית imresize של matlab כדי להפוך את כל מטריצות האותיות שלנו ל100 על 100 וכך יהיה לנו גודל סטנדרט בו משווים בין תמונת הקלט לתמונת database. לבסוף בעזרת סקריפט dictionary.m שבו הכנסנו ידנית נתונים לפי תמונת הdatabase, נכין מערך cell סופי שמחזיק לכל מטריצת אות את האות המתאימה לה.
3. מקבלים את תמונת הקלט ומבצעים עליה מניפולציות דומות על מנת להכניס גם אותה לתוך מערך cell ולחלק אותה לצורות בדיוק כמו שעשינו לתמונה הקודמת, רק שהפעם לא נוכל לחבר צורות לוגיות השייכות לאותו אות ידנית כמו מקודם, כך שכעת יש לנו 2 מערכי cell – באחד יש את כל "בנק" התווים , ובשניה יש את רק את התווים שיש בתמונת הקלט. למערך השני נוסיף גם את מיקום מרכז האות בטקסט כדי להתגבר על הבעיה הבאה:

* השיטה שאנו משתמשים בה לצורך חלוקת האותיות (bwlabel) קוראת את האותיות מלמעלה למטה, וזה משבש לנו את התוכניות (במידה ובקלט יש יותר משורה אחת), שכן עלינו לקרוא משמאל לימין. כאן נתקלנו באתגר לא פשוט שבסיומו הצלחנו החלטנו להתמודד עם הבעיה באמצעות מיון : נבחר ערך סף שהוא גובה של אות, ובאמצעותו נמיין את האותיות לפי קטוגוריות (שורות). כאשר אות רחוקה ממרכז אות שקובעת את סטנדרט השורה **פחות** מהערך סף שקבענו, נשנה את ערך הY שלה ככה שיתאים לסטנדרט השורה. כאשר אות רחוקה **יותר** מהערך סף ממרכז אות שקובעת את הסטנדרט, נקבע סטנדרט שורה נוסף ונבדוק את האותיות הבאות **גם** לפי הסטנדרט החדש. לבסוף נמיין את כל האותיות קודם לפי ערך הY ואז לפי ערך הX בעזרת sortrows של matlab. עכשיו קיבלנו את הסדר הנכון של האותיות.

4 . כעת, אנו מגיעים לפונקציית ההשוואה, אשר עבור כל מטריצת אות במערך הקלט נשווה עם כל האותיות במערך ה database. ההשוואה היא עבור כל פיקסל, נספור את כמות ערכי הפיקסלים הדומים בכל מטריצה, ולבסוף נבחר את האות שהביאה את התוצאה הכי גבוה.

עכשיו רק נותר לכתוב את האותיות לקובץ notepad: נרצה לדעת מתי להוסיף רווח בין המילים. גם כאן נתקלנו בבעייה לא פשוטה, והחלטנו שוב להשתמש בערך סף הפעם על ערכי x. מדדנו את המרחק בין אותו לאות כל עוד הן באותה שורה, לאחר מכן עשינו ממוצע של כל המרחקים וקיבלנו את ערך הסף שלנו. עכשיו לפני שנדפיס כל אות נבדוק את המרחק לאות הבאה ובמידה והוא גדול מערך הסף נוסיף רווח. לצערנו התוצאות אינן מושלמות, ניסינו לשחק עם ערך הסף : למשל להכפילו ב1.5 או ב0.8 , אך לא מצאנו ערך שתמיד עונה על הדרישות.

# שיטות מיוחדות

1. Bounding-Box
2. Label

# קשיים ואתגרים

קשיים איתם התמודדנו בהצלחה ☺ :

קריאת קלט עם מספר שורות באופן תקין - שיטת ה bwlabel קוראת את הצורות בתמונה מלמעלה למטה, כלומר שאם קיים קלט עם 2 שורות, אזי השיטה לא תנתח את האותיות לפי הסדר הנכון (משמאל לימין), כפי שהוסבר למעלה היינו צריכים למיין את הקלט לפי מיקום הצורות (=האותיות) וכך לכתוב בצורה תקינה את הפלט.

קשיים איתם התמודדנו בהצלחה חלקית:

קלט בגדלים שונים – אתגר משמעותי איתו נתקלנו היה קלטים בגדלים קטנים. הבעייה היא כי צורה של אות קטנה מבחינת פיקסלים יכולה להיות שונה מאוד מאותה אות בגודל גדול. לשם המחשה קו באלכסון כאשר האות גדולה ייראה כמו מדרגות שהאות קטנה, ופונקציית imresize אינה יודעת לתקן את זה. בעייה נוספת היא ש imresize מוסיף רעשים למטריצה. לאחר הרבה ניסוי וטעייה ניקינו את המטריצה לאחר שינוי הגודל בצורה שרוב האותיות גם בגדלים יחסית קטנים נקלטות נכון על ידי האלגוריתם.

# קשיים איתם לא הצלחנו להתמודד:

קריאה תקינה של האות i עדיין לא קוראת באופן מלא. לפעמים נקרא בטעות את הנקדות של "? ,"! " ,"i" ,"j"

ניסינו להשתמש בקריאת צורות באמצעות שיטת extractFeatures, אך ללא הצלחה.

ניסינו לקרוא "." (נקודה) ובהתאם להחליט מה לכתוב לפי האות הבאה שנקרא (אם זה פס אז נדע כי הנקודה היא חלק מ i) אך גם ללא הצלחה, בשל דמיון רב בין הקו של i לקו של l ולקו של ! ,ובשל בעיות דומות נוספות .

ניסינו גם כאן להשתמש בערך סף כדי למצוא את הנקודות השייכות לכל אות אך הסתבכנו במימוש ולא הספקנו למצות את ניסיון זה.

מכיוון שהתמונה הראשונה שהכנסנו לקוד שלנו (עליה פורט בסעיף 1) מגיעה בגודל ספיציפי, ואחד האתגרים שלנו הוא שהקלט יוכל להיות במספר גדלים שונים, היינו צריכים ליצור צורה אחידה לכל אות. לכן החלטנו שכל אות מה"בנק אותיות" שלנו תשתנה לגודל 100/100 , וכן כל אות בקלט – וכך הצלחנו להשוות בצורה זהה ברוב המוחלט של המקרים בין האותיות.

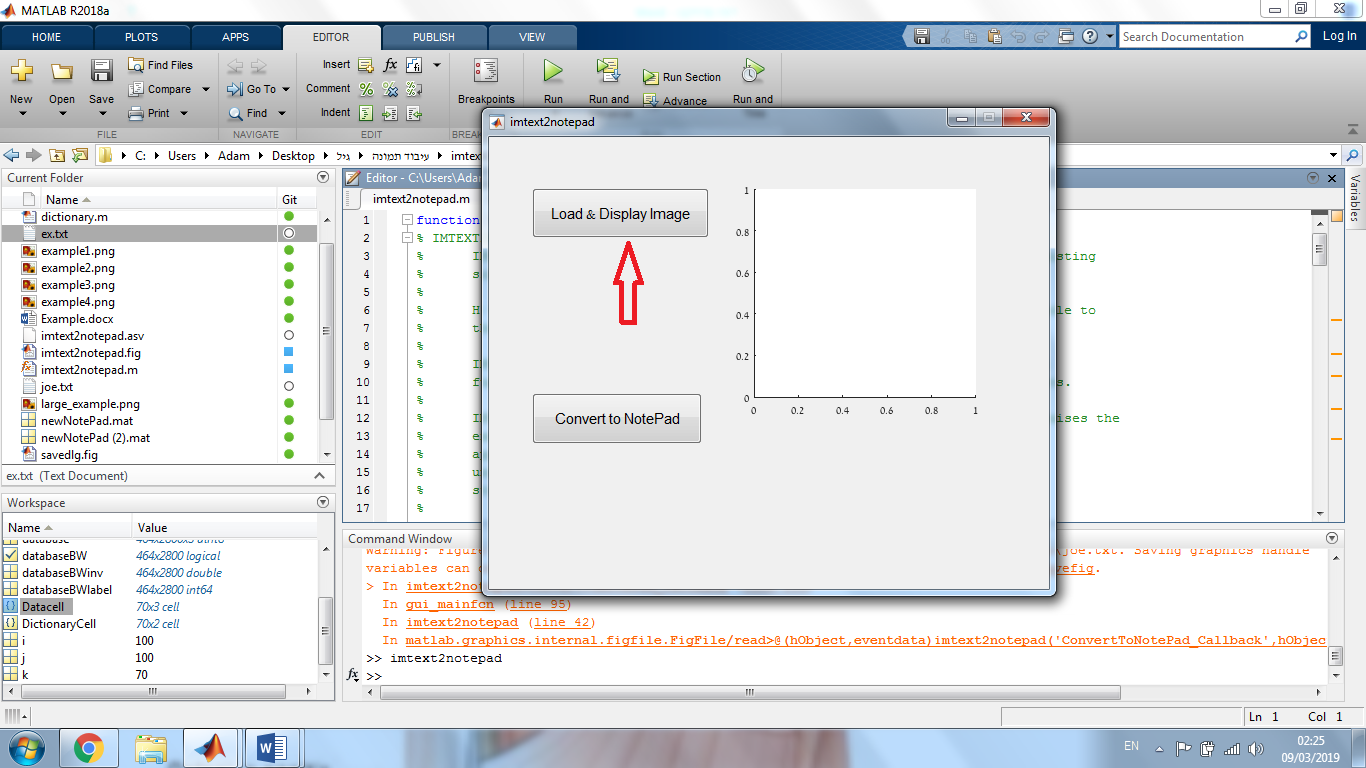
לאחר קריאה מעמיקה חוזרת של כלל החומרים הרלוונטים שלמדנו בקורס, ומחקר באינטרנט, הבנו שאנחנו עומדים להתמודד עם מספר קשיים מרכזיים, והם:

1. בשל ההחלטה לא להשתמש בספריות קיימות (שכן אז לא היה לנו פרויקט) היינו צריכים "ללמד" את הקוד שלנו איך שנראית כל אות. היה עלינו לעבור אות-אות, לחבר אותיות בעיתיות (כאלו שבנויות משתי צורות, כדוגמת j), לזהות את הסדר הכרונולוגי בזיהוי האותיות של המחשב, ולכתוב ידנית כל צורה מהי.
2. קלט בגדלים שונים – מכיוון שהתמונה הראשונה שהכנסנו לקוד שלנו (עליה פורט בסעיף 1) מגיעה בגודל ספיציפי, ואחד האתגרים שלנו הוא שהקלט יוכל להיות במספר גדלים שונים, היינו צריכים ליצור צורה אחידה לכל אות. לכן החלטנו שכל אות מה"בנק אותיות" שלנו תשתנה לגודל 100/100 , וכן כל אות בקלט – וכך הצלחנו להשוות בצורה זהה ברוב המוחלט של המקרים בין האותיות.
   * שינוי הגודל גורם לתמונה הבינארית להיטשטש, מוסיף לה רעשים ונוספו לה עוד מספרים למטריצת התמונה, תיקנו זאת באופן ידני ובאמצעות השיטה medfilt2 לתיקון רעשים.

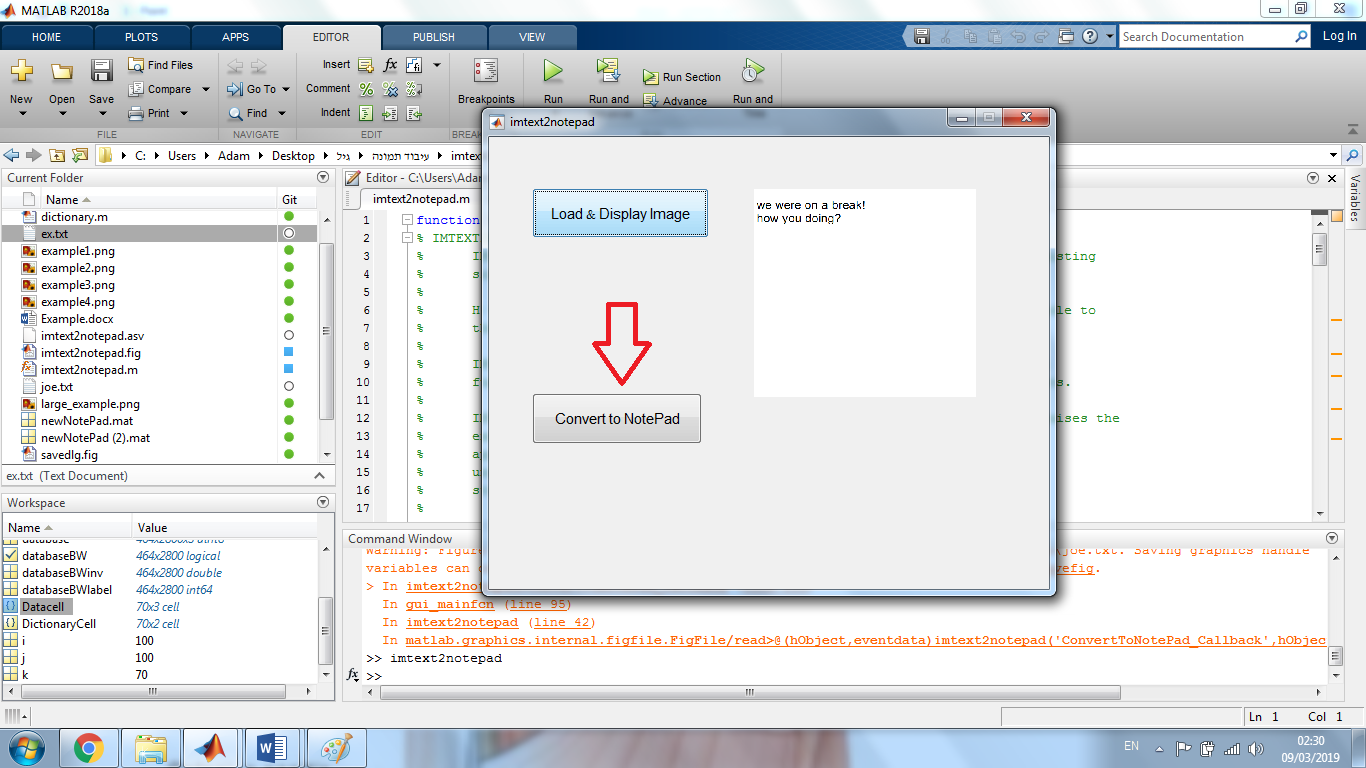
הדרכת הפעלה

הפרוייקט שלנו מופעל דרך הgui שהוא imtext2notepad.m

לאחר הפעלת הgui יש לטעון תמונת טקסט לבחריתך



לאחר מכן יש ללחוץ על הכפתור ולקבוע איפה לשמור



עכשיו ניתן להינות מהתוצאות!