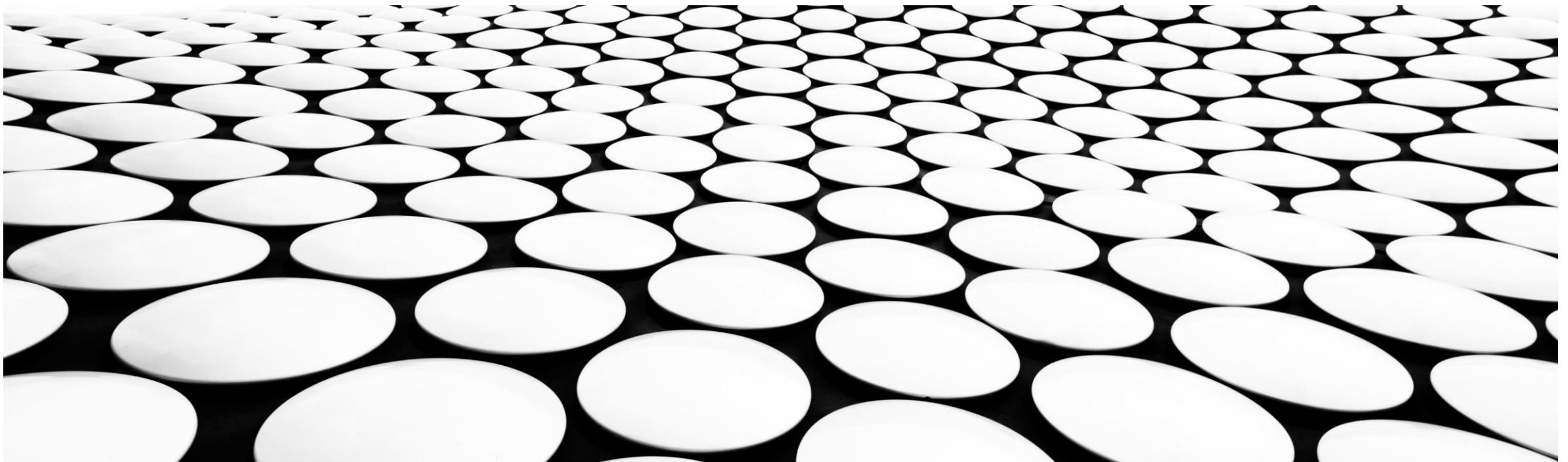


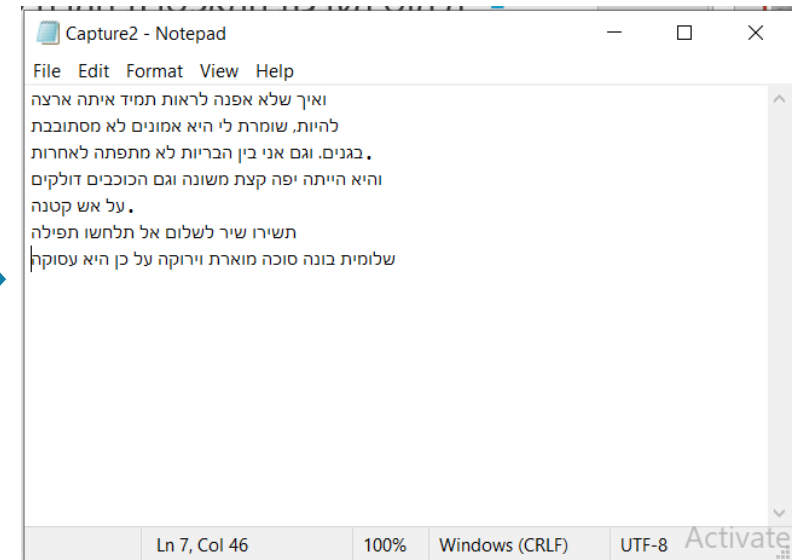
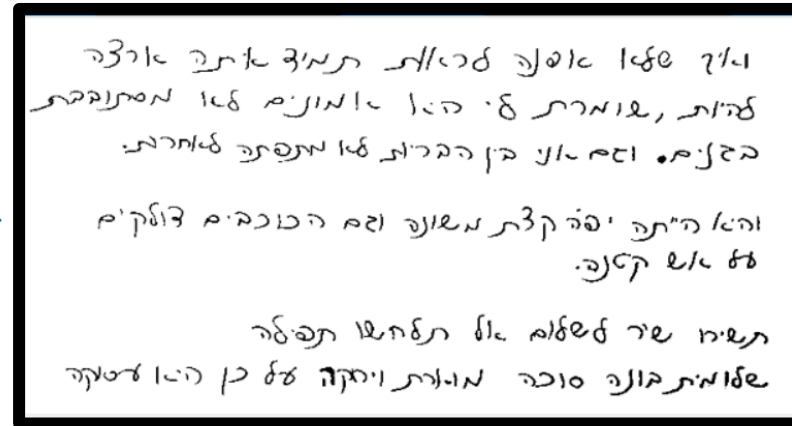
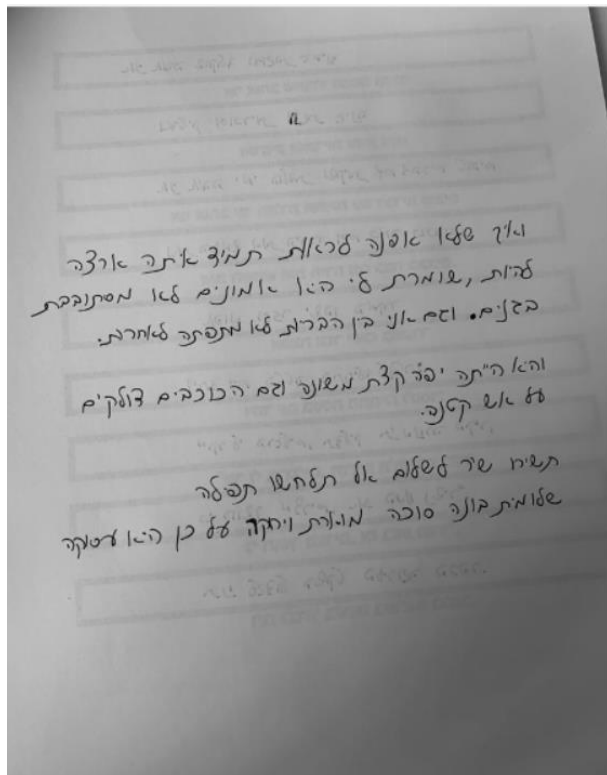
מתמונה לטקסט - חילוץ כתב יד בעברית SHECODES

עדי רוזנטל, אוקטובר 2020



מטרת הפרויקט

מימוש מערכת המאפשרת המרה של תמונה המכילה טקסט בעברית (בכתב יד / דפוס) לכדי כתב מחשב, אותו ניתן לערוך ולעצב בצורה נוחה.

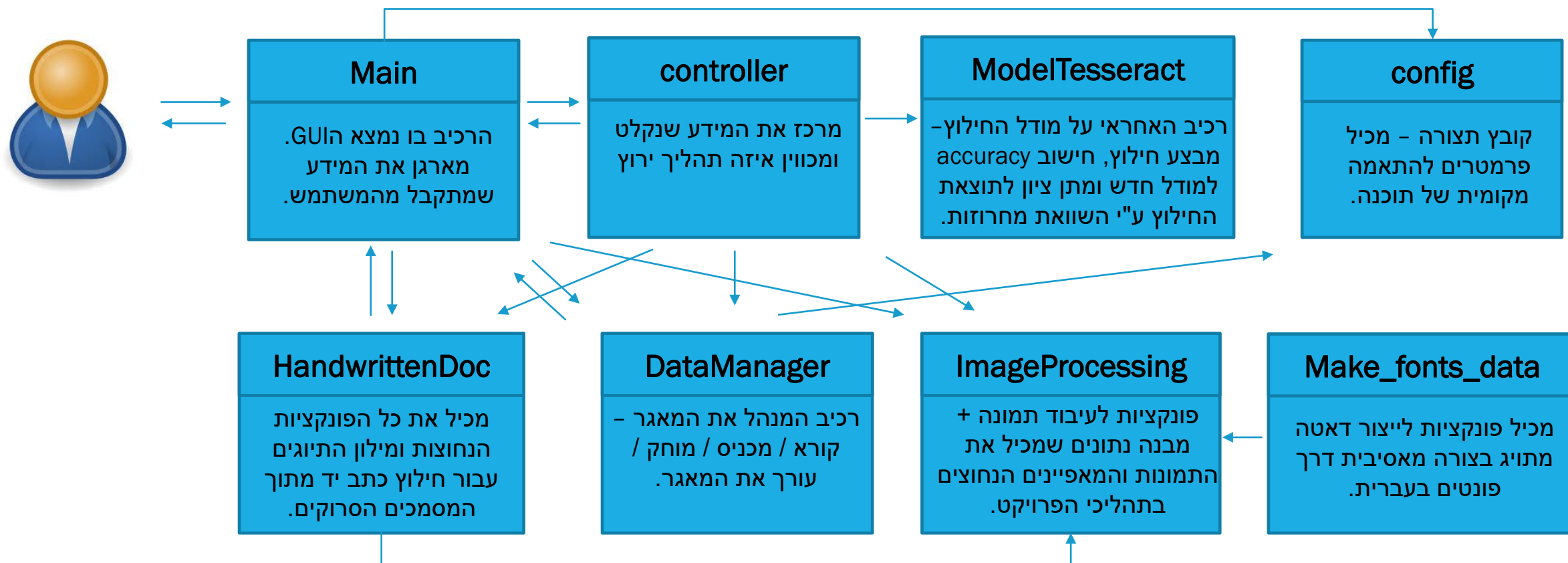




שלושה תהליכים עיקריים בפרויקט:

1. איסוף וחילוץ של דאטה מתויג בכתב יד בעברית.
2. אימון רשת נוירונים tesseract המבוססת OCR.
3. יצירת ממשק נוח למשתמש, לחילוץ טקסט מתוך תמונה ואיסוף דאטה עבור אימון הרשת.

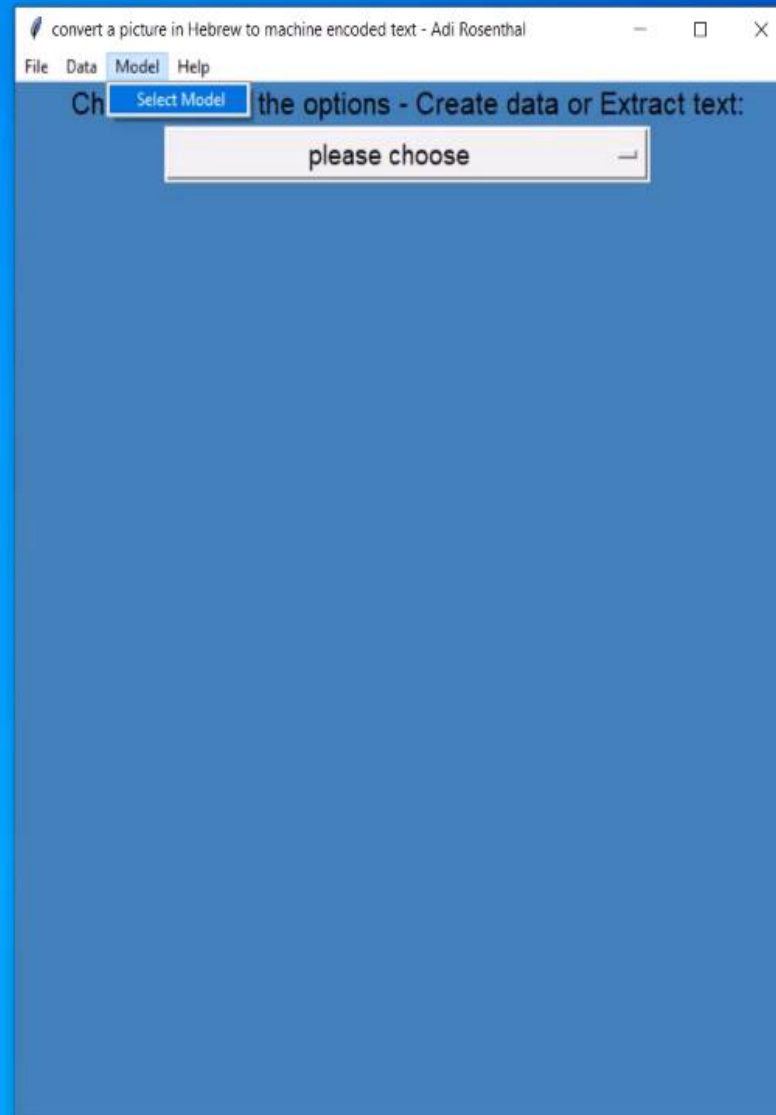
ארכיטקטורה



מבנה נתונים (DB)

תהליך טיוב המערכת לחילוץ טקסט, דורש אימון של רשת הנוירונים עם כמות גדולה של דאטה.

- תצורת הדאטה: קובץ תמונה (TIF) + קובץ טקסט (txt).
- הדאטה צריך להיות מאורגן כך שבכל תמונה מופיעה שורה אחת של כתב יד + קובץ טקסט בעל שם זהה, המכיל את טקסט הכתוב בתמונה.
- הדאטה צריך להיות מדויק ובאיכות גבוהה.
- כדאי לבצע גיוון בכתבי היד (כדי למנוע overfitting).



וידאו
בסיס הנתונים

יצירה והבניה של מידע

■ שלוש שיטות עיקריות בהן השתמשתי ליצירת דאטה בפרויקט:

1. סריקה וחילוץ של שורות טקסט מתוך דפים מוכנים ("Template"), אליהם היו צריכים המשתמשים להעתיק שורות של שירים בכתב ידם, לפי ההנחיות בדף.
2. סריקה של תמונה עם טקסט ותיוג התמונה בתוך ממשק המערכת.
3. איסוף פונטים בכתב יד בעברית, חילוץ שורות טקסט ותיוגם.

סריקה וחילוץ של שורות טקסט מתוך דפים מוכנים

2

אני אוהב שוקולד ואזלות גבינה
אני אוהב שוקולד ואזלות גבינה

וארטיק וסוכריות וזאת גינה
וארטיק וסוכריות וזאת גינה

אני אוהב יאני הולצת ושקיות עם דברים טובים
אני אוהב יאני הולצת ושקיות עם דברים טובים

ואתר השמש ואתר הירח ואם כמה כוכבים
ואתר השמש ואתר הירח ואם כמה כוכבים

אפונה ואסר ישבו במקור
אפונה ואסר ישבו במקור

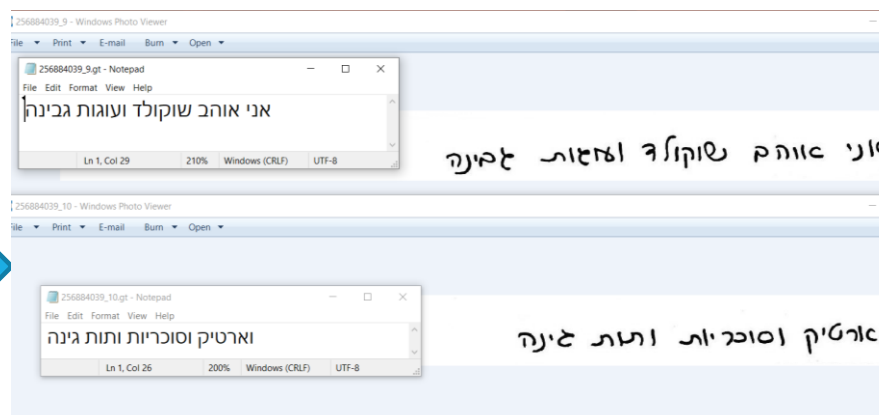
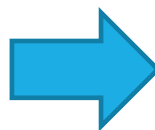
וזה עם בטטה נמאלץ לקטר
וזה עם בטטה נמאלץ לקטר

וזה עם בטטה נמאלץ לקטר
וזה עם בטטה נמאלץ לקטר

כי חושך אפנים, אלא באו נשיר
כי חושך אפנים, אלא באו נשיר

תל אביב קטנה קטנה קטנה
תל אביב קטנה קטנה קטנה

זהו תוכן הדף
זהו תוכן הדף

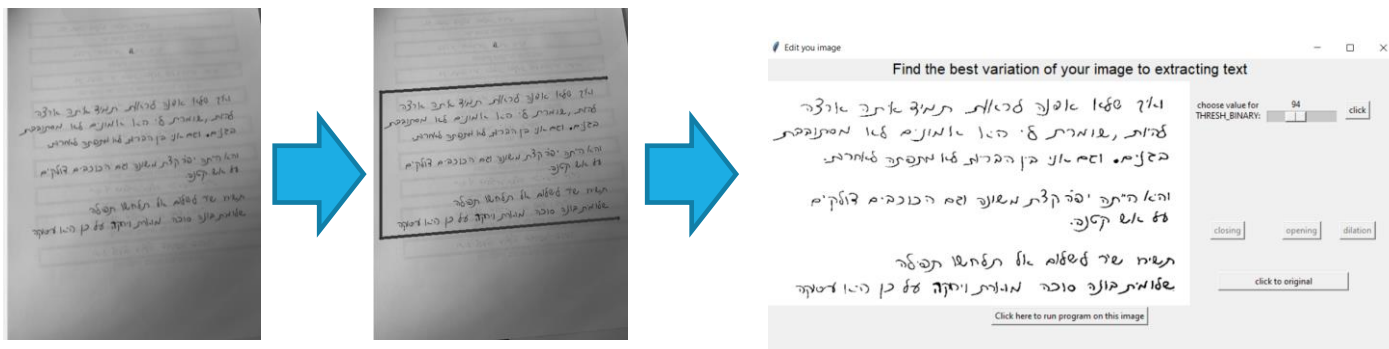


תהליך:

- קבלת קובץ PDG / קובץ תמונה.
- במידה ומתקבל PDF מתבצעת המרה ל-PNG.
- עיבוד קבצי התמונה ע"י המשתמש – יישור וניקוי רעשים.
- חילוץ מלבנים מהתמונה.
- ניקוי "רעשים" בתיחומי המלבנים לפי גודל.
- ניקוי של מלבנים מושחרים.
- חיתוך של התמונה לפי תיחום המלבן.
- תיוג התמונה לפי מילון מוגדר:

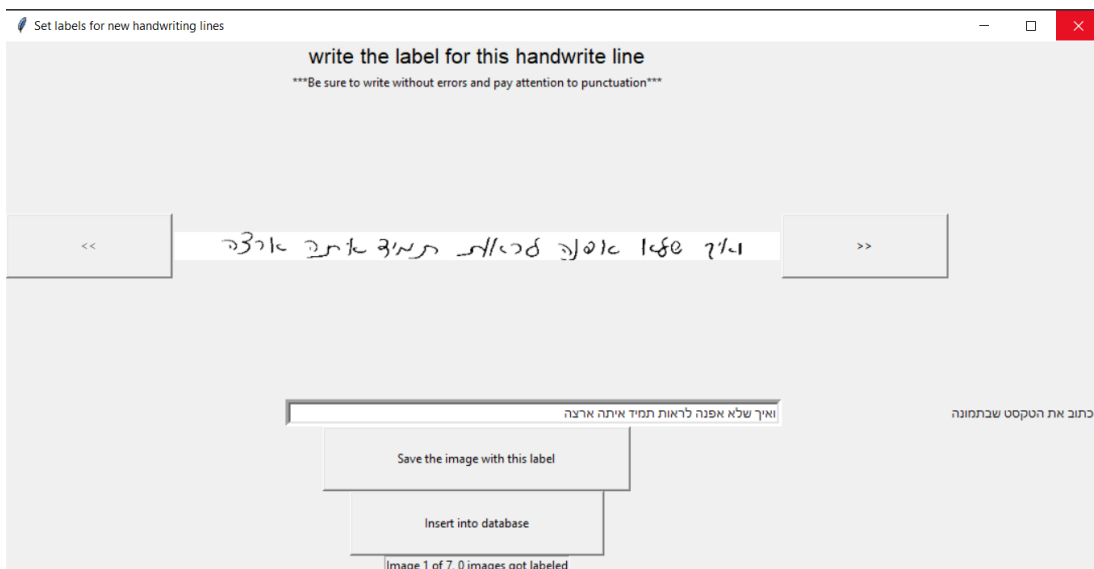
label -> (page, line)

תיוג תמונות על ידי המשתמש



תהליך :

- קבלת קלט - תמונה עם טקסט בכתב יד.
- עיבוד קבצי התמונה ע"י המשתמש – יישור וניקוי רעשים.
- הפרדה של הטקסט לשורות על ידי ערך מינימלי של צבע בכל שורה.
- הצגה של השורות בממשק תיוג עבור המשתמש.

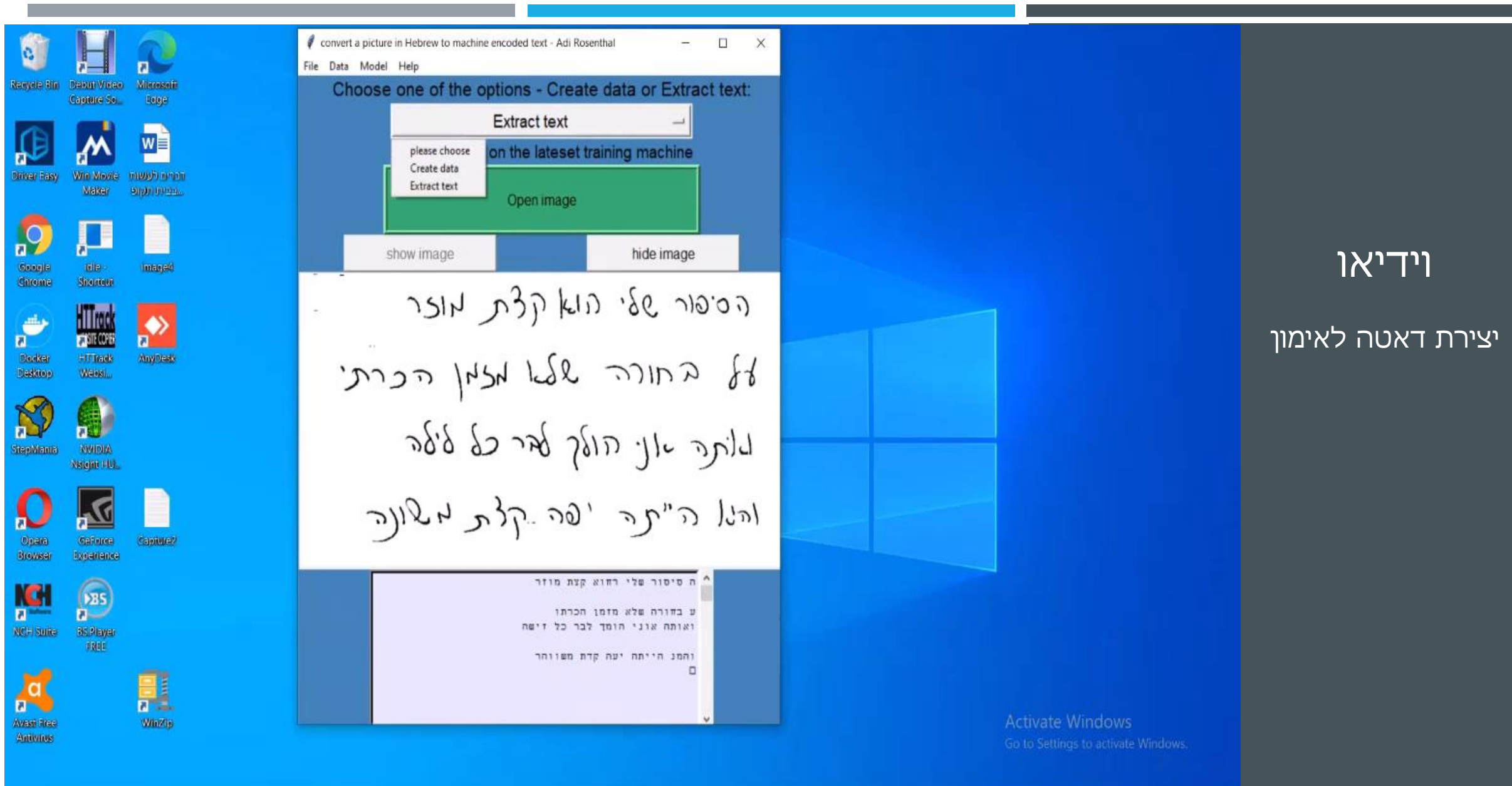


תהליך:

- ריכוז פונטים של כתב יד:
 - חיפוש פונטים שקיימים באינטרנט
 - איסוף כ-30 כתבי יד מאנשים שונים ויצירת פונטים ע"י אתר caligraphr.
- ריכוז טקסט ארוך בעברית.
- הפרדה של הטקסט לשורות של כ-6 מילים בשורה.
- השמה של הפונטים במסמך וורד, המרה לPDF ובסוף לקובץ TIF + יצירת קובץ טקסט מותאם לתמונות.
- חילוץ שורות טקסט מתוך התמונות.
- תיוג השורות לפי מספר שורה במסמך טקסט.

איסוף דאטה בעזרת פונטים





וידאו

יצירת דאטה לאימון

אימון הרשת - TESSERACT

שלושה שלבים עיקריים במהלך הפרויקט עבור תהליך אימון הרשת:

1. מחקר כלים ואפשרויות לאימון הרשת + יצירת סביבה מתאימה לאימון.

2. תהליך האימון.

3. בדיקה של הרשת המאומנת.

תהליך האימון

```
docker_steps.txt
~/Documents/shecodes/train/train-tesseract

1 OUT DOCKER:
2 cd ~/Documents/shecodes/train/train-tesseract
3 sudo docker-compose -f docker.dev.yml up
4 sudo docker exec -ti train-ocr-heb bash
5
6 IN DOCKER:
7 cd /app/src/tesstrain/data
8 rm -r heb?-ground-truth
9 mkdir hebX-ground-truth
10 cp -a /app/data/hebX-ground-truth/* /app/src/tesstrain/data/hebX-ground-truth/.
11 cd /app/src/tesstrain/
12 make training MODEL_NAME=heb? START_MODEL=heb LANG_TYPE=RTL PSM=7 TESSDATA=/usr/
   local/share/tessdata
13 cp /app/src/tesstrain/data/heb?.traineddata /usr/local/share/tessdata/
14
15 OUT DOCKER:
16 sudo docker container ls|
17 sudo docker cp CONTAINERID:/app/src/tesstrain/
   share/tesseract-ocr/4.00/tessdata
18
19
```

```
root@dd8575c1527d: /app/src/tesstrain
root@dd8575c1527d:/app/src/tesstrain# make training MODEL_NAME=heb9 START_MODEL=
heb7 LANG_TYPE=RTL PSM=7 TESSDATA=/usr/local/share/tessdata
find data/heb9-ground-truth -name '*.gt.txt' | xargs cat | sort | uniq > "data/h
eb9/all-gt"
combine_tessdata -u /usr/local/share/tessdata/heb7.traineddata data/heb7/heb9
Extracting tessdata components from /usr/local/share/tessdata/heb7.traineddata
Wrote data/heb7/heb9.lstm
Wrote data/heb7/heb9.lstm-unicharset
Wrote data/heb7/heb9.lstm-recoder
Wrote data/heb7/heb9.version
Version string:4.1.0
17:lstm:size=3039635, offset=192
21:lstm-unicharset:size=4653, offset=3039827
22:lstm-recoder:size=724, offset=3044480
23:version:size=5, offset=3045204
unicharset_extractor --output_unicharset "data/heb9/my.unicharset" --norm_mode 3
"data/heb9/all-gt"
Bad box coordinates in boxfile string!
```

- ארגון של הדאטה והעברתו לסביבה המתאימה.

• DOCKER

• UBUNTU 20

• Virtual machine

- שימוש בקוד פתוח Tesstrain שעוזר להכין את הדאטה עבור אימון.

• .txt + tif -> .box .lstm

• Tesseract training tools

- העברה של הרשת המאומנת (hebX.traineddata) לתקית התקנה של Tesseract.

בדיקת הרשת

תוצאות של אימוני הרשת

his PC > Documents > shecodes_finalProject > models-trained		
Name	Date modified	Type
test_models	10/14/2020 8:14 AM	File folder
heb.traineddata	7/31/2020 5:04 PM	TRAINEDDATA File
heb3.traineddata	10/8/2020 9:23 AM	TRAINEDDATA File
heb4.traineddata	10/11/2020 4:22 PM	TRAINEDDATA File
heb-fonts3.traineddata	10/7/2020 9:28 AM	TRAINEDDATA File
heb-fonts4.traineddata	10/7/2020 2:42 PM	TRAINEDDATA File
instruction train on docker	10/7/2020 9:37 AM	Text Document

דו"ח לבדיקות רשת

```
heb7_13_10_17_52 - Notepad
File Edit Format View Help
DATE : 13_10 17_52
Model : heb7

Validation folder: C:\Users\Adi Rosental\Documents
\shecodes_finalProject\models-trained\test_models, length
data: 32.0

Algorithm of compare strings : SQ

ואיך שלא אפנה לראות תמיד איתה ארצה
Output: ואיך שלא אפנה לראות תמיד אייתה ארצה
♣Befor: ך<«6 ׁ ׁ(0]20_ה ש_ריוכ
♣Percent coincidence after train: 92.96%
diff between befor train and after: 71.75%

להיות, שומרת לי היא אמונים לא מסתובבת
Output: להיות, שומרת לי היא אמונים לא מסתובבת
♣Befor: ׁ ׁ122#כר/ & ׁשן < /עאקי 6& ׁתלנבלר
♣Percent coincidence after train: 94.74%
diff between befor train and after: 70.05%

בגנים. וגם אני בין הבריות לא מתפתה לאחרות
text:
Ln 1, Col 1 100% Windows (CRLF) UTF-8
```

```
from difflib import SequenceMatcher as SQ
```

```
def Check_model_tesseract(self, folder_validation, folder_output_txtfile, psm=7, compare_methods = "SQ"):
```

- הרצה של הרשת החדשה על הדאטה לבדיקות (בתיקיית validation).

- הרשת לא אומנה על הדאטה הנ"ל.

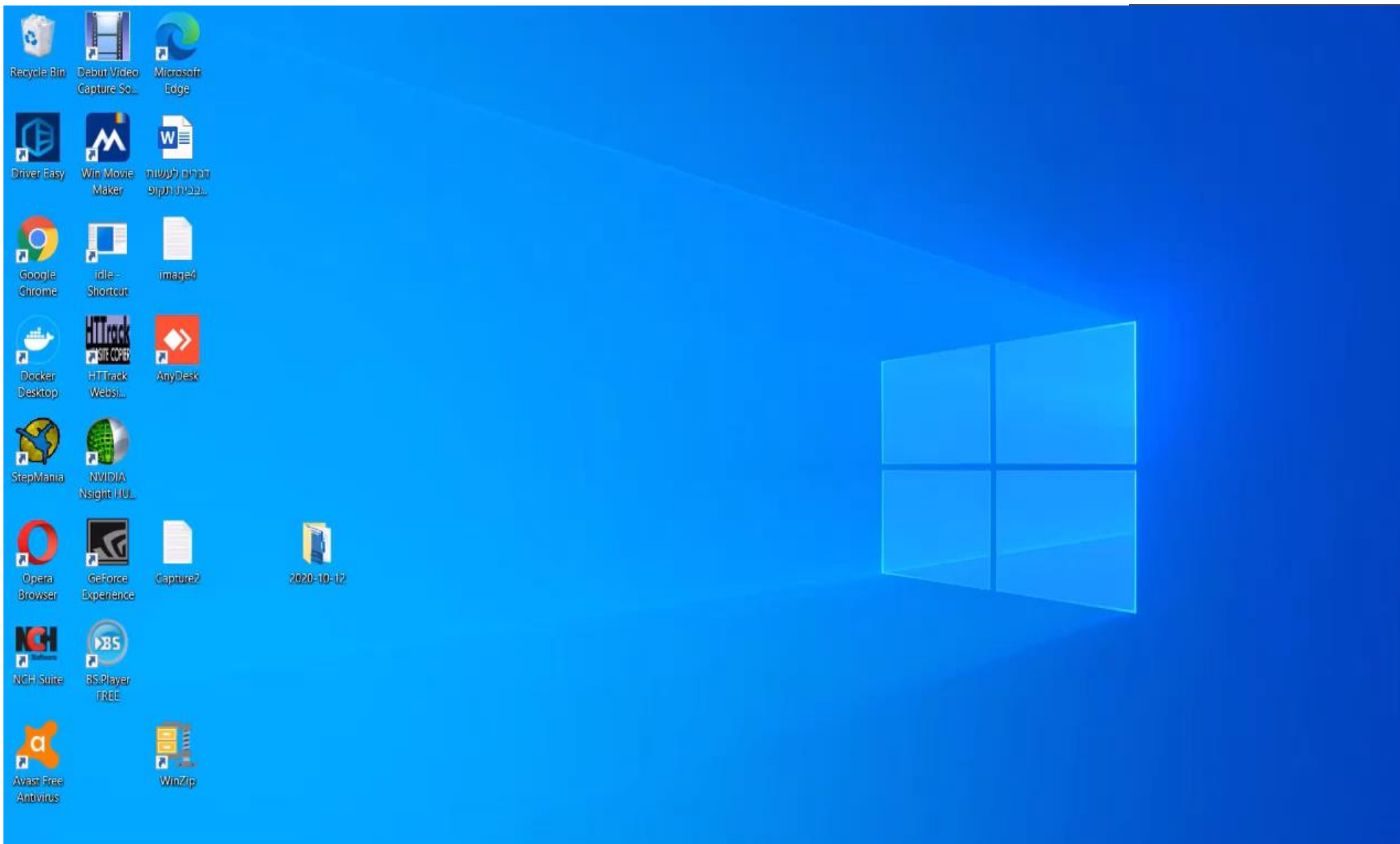
- השוואה של תוצאת הרשת לתיוג "אמת"

- שימוש בפונקציית SequenceMatcher המבוססת על אלגוריתמים להשוואת מחרוזות – LCS – longest contiguous matching.

- האלגוריתם לא מתייחס למשקלים של אותיות – כמו למשל אותיות דומות וכו'.

- יצירת דו"ח.

- השוואה בין תוצאות הרשת המאומנת על כל שורה בvalidation, לתוצאת הרשת של אותם המשפטים לפני האימון והערכה של אחוז שיפור.



וידאו

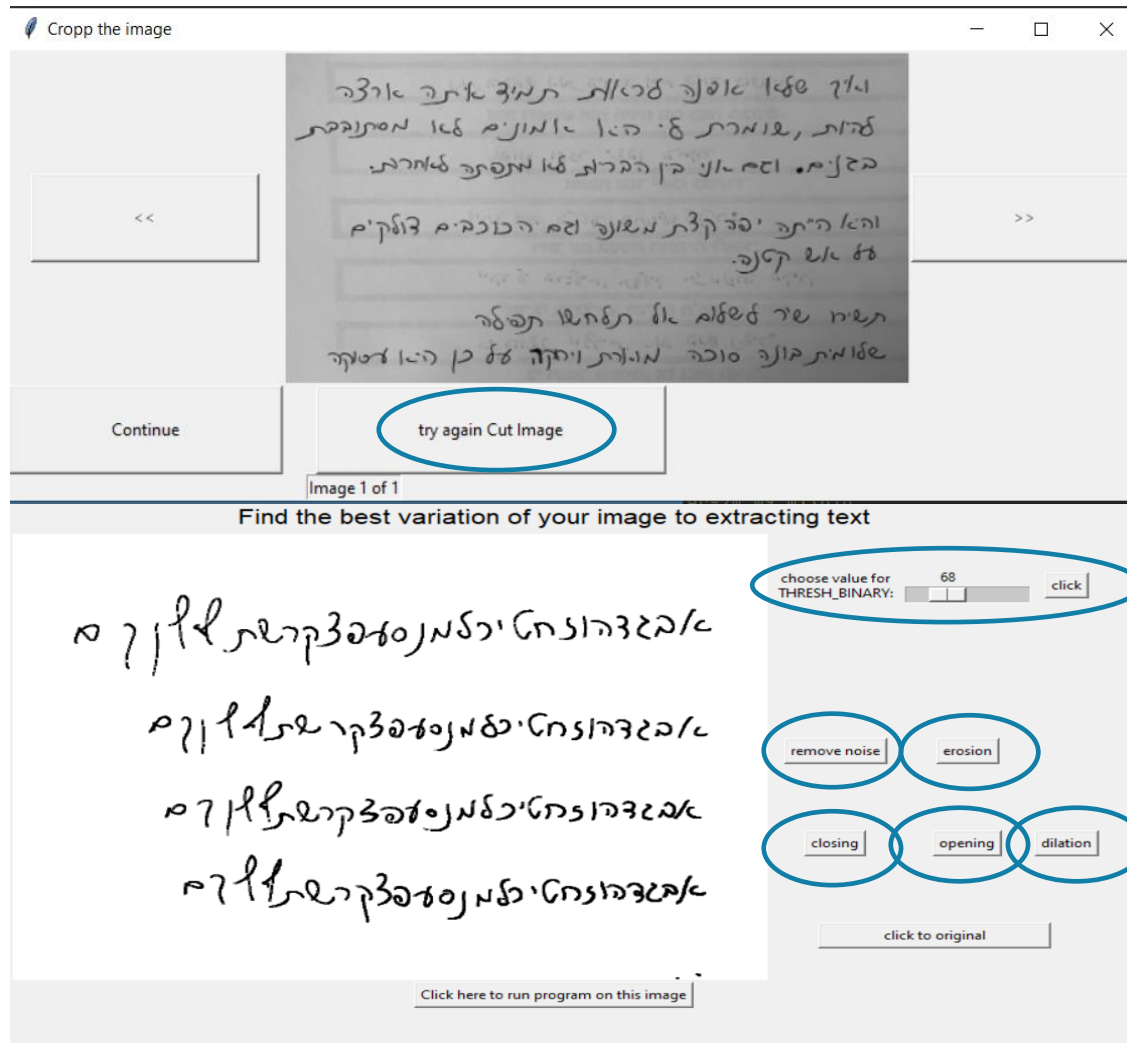
אימון מודל

חילוץ טקסט מתוך תמונה

- עיבוד תמונה לפני חילוץ.
- Pytesseract.
- השוואת תוצאות הרשת ל"אמת".



עיבוד תמונה לפני חילוץ



- כדי שהחילוץ יהיה אופטימלי כדאי לבצע עיבוד תמונה, לפני שימוש ברשת נוירונים.

- בממשק ישנן מספר אפשרויות להטיב עם תוצאת הרשת :

1. מרכז של הטקסט על ידי סימון האזור בו מצוי הטקסט אותו אנחנו רוצים לחלץ.

2. ניקוי רעשים בעזרת פונקציות לעיבוד תמונה (OpenCV):

Binary – thresholding

medianBlur

Dilate

Erosion

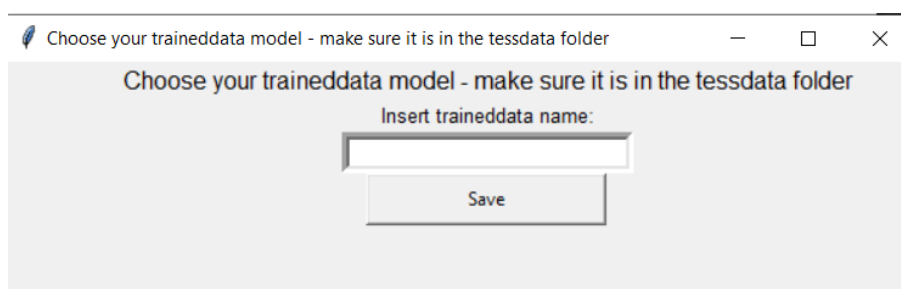
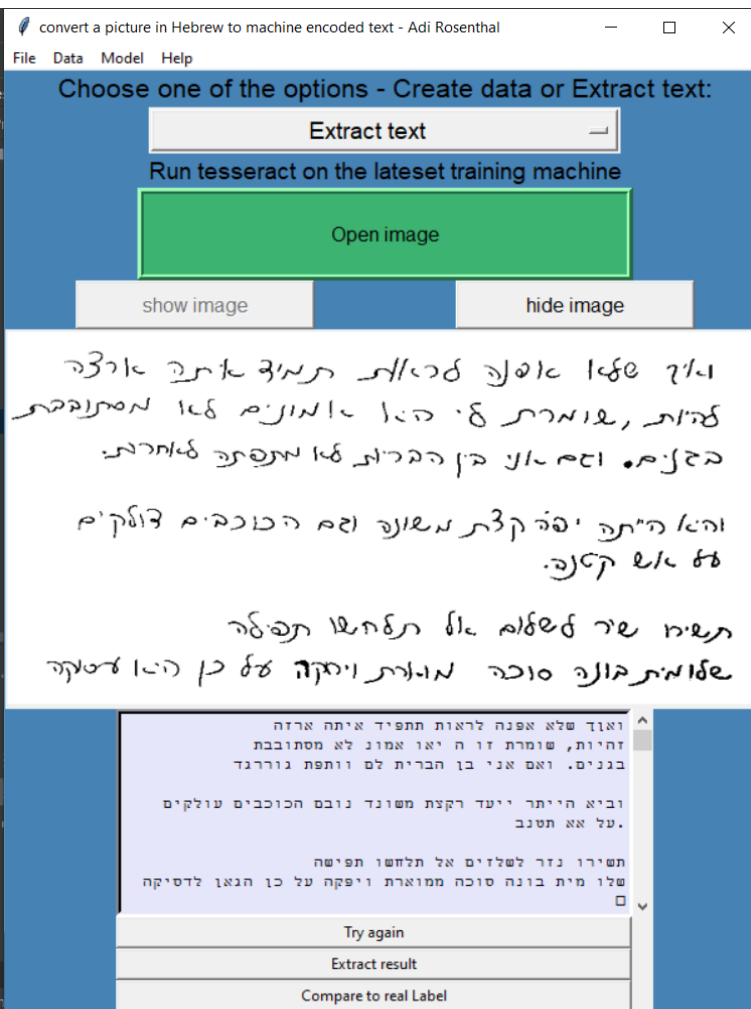
dilate

חילוץ טקסט

תהליך חילוץ הטקסט:

- אופציה לבחירת רשת נוירונים בעזרתה נרצה לחלץ את הטקסט.
- הרשת הנבחרת צריכה להיות שמורה בתיקיית tesseract.
- באופן דיפולטי הרשת מוגדרת להיות הרשת הכי איכותית.
- החילוץ עצמו מתבצע ע"י שימוש בספריה pytesseract שמתממשת לתוכנה של tesseract – מקבלת תמונה ושם של רשת.
- שמירת תוצאות הרשת כקובץ טקסט (Extract result).

```
def ExportTextTesseract(self, image):  
    print(self.lang)  
    str = pytesseract.image_to_string(image, lang=self.lang)  
    return str
```



השוואת תוצאות

Compare result to the real label

Write the real Label for the image:

Result using trained tesseract model:

ואיך שלא אענה לראות תמיד אתה ארצה להיות, שומרת לי הא אטמונים לא מסתובבת בגנים: וגם אני בין חבריות לא מתפתח לאחרת והיא היתה יפה קצת משונה וגם הכוכבים דולקים על אש קטנה תנשיתו שיר לשלום אל תלחשו תפילה שלומית בונה סוכה מומרת ויחקה על כן היא לטקה

Please write the real label of the text :

ואיך שלא אפנה לראות תמיד איתך ארצה להיות, שומרת לי היא אמונים לא מסתובבת בגנים. וגם אני בין חבריות לא מתפתח לאחרת והיא הייתה יפה קצת משונה וגם הכוכבים דולקים. על אש קטנה תשירו שיר לשלום אל תלחשו תפילה שלומית בונה סוכה מוארת וירוקה על כן היא עסוקה

Calculate match

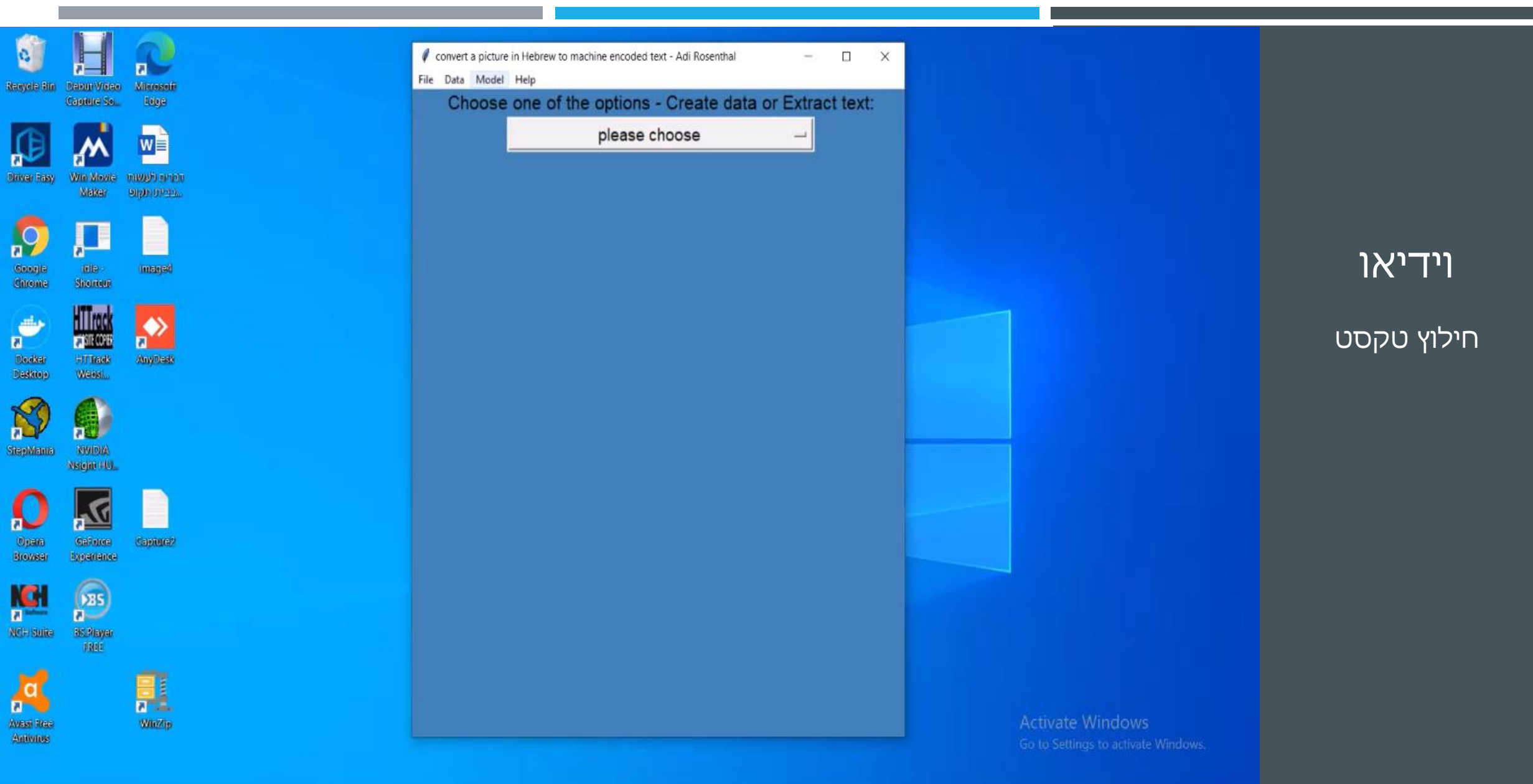
The match percent according similarity - 92.15%

Want to insert your true label into database?

- כדי לבחון את טיב התוצאה שהתקבלה בחילוץ, קיימת אופציה עבור המשתמש להכניס את הטקסט הנכון של התמונה – ה"אמת".

- ההשוואה בין הטקסטים מתקבלת כפרמטר באחוזים על ידי פונקציית "SequenceMatcher".

(לאחר הכנסת התוצאות אמת ניתן גם להכניס את תמונת הקלט עם הטקסט שהמשתמש הכניס בבדיקה אל לתוך בסיס הנתונים)



מה למדתי בפרוייקט?


- Creating Virtual machine
- Using Linux
- Work with Dockers
- Git + Github
- Tesseract / OCR
- Design + Architecture
- Using open source
- OpenCV
- Tkinter
- PIL (Image)
- Difflib
- Training tesseract
- Similarity between strings
- Designing and planning a GUI

צעדים להמשך...

- שיפור תוצאות הרשת בשיטות נוספות כמו:
 - NLP.
 - שימוש בתיקון על פי מילון.
 - סינון של מילים ואותיות- בשילוב עם התניות בשפה העברית.
 - https://github.com/NLPH/NLPH_Resources
- בניית אתר אינטרנט / אפליקציה נוחה למשתמש.
- הבנה מעמיקה יותר של אלגוריתם tesseract ותהליך האימון כדי להשפיע יותר על האימון.
- בדיקה יותר מעמיקה של תוצאות הרשת.

קישורים

- Tesseract-OCR : <https://github.com/tesseract-ocr/tesstrain>
- <https://medium.com/@guiem/how-to-train-tesseract-4-ebe5881ff3b7>
- Train-tesseract and docker files for train : <https://github.com/guiem/train-tesseract>
- Tesstrain : <https://github.com/tesseract-ocr/tesstrain>
- <https://tesseract-ocr.github.io/tessdoc/ImproveQuality>
- <https://www.makeuseof.com/tag/create-virtual-machine-using-windows-10-hyper-v/>
- Learn OpenCV: <https://www.youtube.com/watch?v=N81PCpADwKQ&t=6764s>
- Learn TKinter: <https://www.youtube.com/watch?v=YXPyB4XeYLA>
- Learn docker: <https://www.youtube.com/watch?v=i7ABlHngi1Q>
- SequenceMatcher in python : <https://towardsdatascience.com/sequencematcher-in-python-6b1e6f3915fc>



שאלות?