**SW Test course (67778) - 2021**

**Final Project - Team ‘Artium’**

Aviram Yitzhak - 312181506

Shahar Mor - 203713417

Tom Nahum - 316240985

Omer Salman - 208966549

**Question 1: Feature Breakdown**

There are two major features that we are required to test: Command Line Configurations and Configurations File. These two interfaces allow us to run tesseract with different parameters. For that, we need to verify that we receive the same output for each parameter and image - both from the CMD and configurations file. For the same reason, we also need to check what happens when sending conflicting configurations from the two interfaces (like whitelisting a character in the CMD and blacklisting it in the config file). We also need to check the relation of the CMD and config file with the “-l” parameter that represents the language of the text in the image. For example, conflicting values for the “tessedit\_load\_sublangs” parameter and the “-l” parameter.

|  |  |
| --- | --- |
| **Feature** | **Comments** |
| **CMD Language** |  |
| English, Spanish, Hebrew and Swedish |  |
| Invalid Language |  |
| **CMD Parameters** |  |
| Each of the valid parameters with valid value |  |
| Each of the valid parameters with invalid value |  |
| **Config File** |  |
| Each of the valid parameters with valid value |  |
| Each of the valid parameters with invalid value |  |
| **Connections between interfaces** |  |
| Conflicting CMD and config file parameters |  |
| Conflicting language and config file parameters |  |
| Conflicting language and CMD parameters |  |

**Question 2: page\_separator**

**What it does**

Adds a newline and then the input string as a page separator to the end of the output text (the default character is the FF character - Format Feed). For many reasons, the user might want to change the separator to a different character (for example, reading a single sentence that is made from multiple images).

**How it behaves**

If the user wants to separate the text with the string S, then:

**tesseract imageFileName outputFileName -c page\_separator=”S”**

will generate the outputFileName text file containing:

**<Text from image>**

**S**

**Anomalies/Bugs (For the bug reports, please see the Appendix):**

1. When using the condition without double quotes, the condition **page\_separator=** is valid.
2. When using the condition **page\_separator=^**, the separator is discarded.
3. When using the page\_separator condition with double quotes and/or backlash characters, some of the output might be discarded.
4. When using the page\_separator condition with a vertical bar and without double quotes, an error occurs.

**Question 3: Metamorphic Relations**

1. **Basic: Image file formats**

If the input file format is valid, using

**tesseract imageFileName outputFileName**

will generate an output file with the given name without error.

**Test Case**

|  |  |
| --- | --- |
| **Action** | **Expected Result** |
| **Setup** | |
| Create several image files (jpg, jpeg, etc.). |  |
| Create several non-image files (txt, doc, etc.). |  |
| **Test** | |
| Run the command with each image file name and the output file name “output”. | The output file was generated. |
| Run the command with each non-image file name and the output file name “output”. | The output file was generated, and an error message appeared. |

1. **Configuration: Page Separator**

When using the page\_separator configuration parameter with the string S,

**tesseract imageFileName outputFileName -c page\_separator=”S”**

the generated output file must contain S in its last line.

**Test Case**

|  |  |
| --- | --- |
| **Action** | **Expected Result** |
| **Setup** | |
| Pick a random string S of random length. |  |
| Create a random image file (contains random characters, random font, etc.) |  |
| The image must be created according to the given specifications (E. All font sizes between 8 and 72 points). |  |
| **Test** | |
| Run the command with the image file name, the output file name “output” and the page\_separator configuration parameter with the string S. | The output file was generated.  The last line contains exactly S. |
| (\*) Can run multiple times using different strings, images, languages, etc.. | |

1. **Language: -l the language flag**

When using the language flag,

**tesseract imageFileName outputFileName -l lang**

the generated output file must only contain characters from language lang.

**Test Case**

|  |  |
| --- | --- |
| **Action** | **Expected Result** |
| **Setup** | |
| Choose one of the available languages – we denote it as lang. |  |
| Create an image file that contains random characters. |  |
| **Test** | |
| Run the command with the image file name, the output file name “output” and the language flag with the chosen lang. | The output file was generated.  The output file only contains characters from lang. |
| (\*) Can run multiple times using different images and the 4 languages. | |

**Question 4: Test Strategy**

**4.1 Equivalence Classes**

|  |  |
| --- | --- |
| **Input** | **Valid Equivalence Classes** |
| **Length of text** | Example: 10 characters  Explanation: An integer in [0,X] where X is the total possible characters that can fit the image according to the specifications. |
| **Position of text in the image** | See image 4.1.p0  Text angled at 0°, characters are full, characters are not overlapping other characters. |
| See image 4.1.p90  Text angled at 90°, characters are full, characters are not overlapping other characters. |
| See image 4.1.p180  Text angled at 180°, characters are full, characters are not overlapping other characters. |
| See image 4.1.p270  Text angled at 270°, characters are full, characters are not overlapping other characters. |
| See image 4.1.p270+0  A mix of 2 to 4 angled text, characters are full, characters are not overlapping other characters. |
| **Language** | 123  No specific language, only numbers + symbols. |
| Abc@10  English + numbers + symbols. |
| אבג#10  Hebrew + numbers + symbols. |
| Añc^10  Spanish + numbers + symbols. |
| åbc\*10  Swedish + numbers + symbols. |
| Añc\_å\_2  A mix of 2 to 4 languages out of English, Hebrew, Spanish and Swedish + numbers + symbols. |
| **Space between words and lines** | ABC  No space/tabs. |
| A BC  Space between words. |
| A  B  C  Space between lines. |
| A A  B B  CC  Space/tab between words and lines. |
| **The content of the text** | Empty. |
| 1 2\*  No language specific characters (numbers and symbols). |
| Ab C-33  Contains alphanumeric and symbol characters. |

**4.2 Test Strategy**

In order to test Tesseract according to the requirements specified in the project description,

while considering test limitations (A team of 4 people, using up to 300 images and less than 15 minutes of PC time) we decided to use different test techniques in order to provide vast coverage of the requirements:

1. There are 42 fonts supported by Tesseract (that we needed to test). In order to check that Tesseract can handle these fonts, we used the boundary values test technique. For each font we generated 4 tests - {English letters, Hebrew letters} X {font size 8, 72}. Notice that some fonts do not support Hebrew letters, so we generated tests only with English letters. Furthermore, note that ‘Marlett’ font does not support English nor Hebrew letters, so we didn’t test it.
2. There are 4 supported languages by Tesseract (that we needed to test). In addition, tesseract should be able to identify numbers and special characters. In order to test this requirement, we created pictures for each subset of [English letters, Hebrew letters, Spanish letters, Swedish letters, Numbers, Special characters] with a generic font (Ariel) and size 20. This test is meant to check that tesseract can identify single as well as multiple languages in a single picture. Notice that since most letters of Spanish, English and Swedish are identical, in order to test other features of Tesseract we chose to make tests only on English and Hebrew letters.
3. There is a range of font sizes which is supported by Tesseract: [8,72] (Integer). So, in order to test, we used boundary values: we generated a total of 10 pictures with generic font Ariel of size 20. The boundary values of font sizes we chose are [8,9,32,71,72] and for each font size we generated 2 pictures, one with Hebrew letters and one with English letters (We didn’t test pictures with Swedish and Spanish letters due to the similarity of its letters with English).
4. Next, we needed to check multiple positioning options of text in the picture. We generated pictures with the generic font Ariel and reasonable size of 40, which has a single English letter in every one of its corners. In addition, we generated a full picture of English letters in order to see if Tesseract is able to correctly identify the text, and on the contrary, we generated an empty image as well to insure that Tesseract doesn’t recognize non-existing characters. Moreover, we generated 3 pictures with a full single line of English letters positioned at the top, middle and bottom of the picture and checked Tesseract performance on it also.
5. According to the requirements, the OCR is limited to text angled at 0, 90, 180, 270 degrees. We generated pictures (generic font Ariel and reasonable size of 40) with English angled text to ensure Tesseract can handle angled text as well.
6. We checked that Tesseract could handle the most popular image formats: JPEG, JPG and PNG. For each format we generated a picture with English letters and an empty picture to ensure Tesseract recognizes correctly (with respect to the project's images number and test running time restrictions).
7. In order to check how Tesseract handles spaces between letters, we generated 5 pictures containing random English characters, spaces and newlines and gave them as input to Tesseract, moreover we generated 4 pictures of a concatenation of English and Hebrew letters and between them: one new line, 2 new lines, 3 new lines, and one with no new lines at all, all of it in order to see if Tesseract performs differently due to new lines between languages.
8. In order to test Tesseract configuration parameters, for each language we generated a picture contains a simple sentence (sentence with 2 words) and used it to test each configuration parameter of Tesseract, as follows:
   1. blackList/unBlackList/whitelist - we created tests with empty list, full list (the whole language alphabet) and partial list (with existing and non-existing letters that appear on the picture). We decided to perform several tests on each picture with respect to equivalence classes testing technique. In addition, we generated tests with combinations of these 3 configurations, to see that there are no contradictions between them and the Tesseract functions as expected.
   2. Page separator - We tested that we can use a unique letter from each of the supported languages as a page separator. Again, we used equivalence classes testing technique in order to minimize the number of tests.

As a group decision we decided to not generate pictures with noise, considering the number of tests limitation and the limited resources. Furthermore, from the same considerations we decided to generate most of the pictures with a generic font type and size (Ariel 20), uniform picture size (1280X1024) and .png picture format, since all of these parameters are checked explicitly in different tests.

**4.3 Strategy implementation**

**4.3.1** Images that fit our strategy - located in the ‘images’ directory in our submission.

**4.3.2** Our test automation - project\_artium.py

**4.3.3** We ran the tests, and the results appears in testResults\_artuim.csv file.

**4.3.4** Bug reports

* Full reports for our most 5 critical bugs are attached to this document as appendix 2.1-2.5.
* The table for the rest of the bugs is attached to this document as appendix 2.6.

**Appendix 1.1: Q2 - Page Separator Bug Report (1)**

**When using the condition without double quotes, the condition page\_separator= is valid.**

**Description:**

Tesseract OCR allows us to choose a custom separator using the **page\_separator=”string”** configuration. The page\_separator can be also used without “”, which leads to an odd looking input:

**tesseract imageFileName outputFileName -c page\_separator= -l eng**

**Steps to reproduce:**

1. Run Tesseract using the following command:

**tesseract imageFileName outputFileName1 -c page\_separator= -l eng**

1. Run Tesseract using the following command:

**tesseract imageFileName outputFileName2 -c page\_separator=”” -l eng**

1. Inspect the output files - see that both are the same.

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameters that were used are page\_separator and the language flag for English.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** Medium

Using the same configuration in multiple ways may cause confusion – especially when there are differences between the two. This is an undocumented feature.

**Appendix 1.2: Q2 - Page Separator Bug Report (2)**

**Using the ^ character as a separator is discarded**

**Description:**

Tesseract OCR allows us to choose a custom separator. While trying to use the ‘^’ character as a separator, the program acts like we choose the ‘’ character as a separator. This happened with multiple images, in english. See the attached images.

**Steps to reproduce:**

1. Run Tesseract using the following command:

*tesseract path.txt output -c page\_separator=^ -l eng*

Or this command:

*tesseract path.txt output -c page\_separator=”^” -l eng*

1. Inspect the output file - see that the “^” character is missing - and instead we find the double quote character (“)

**Setup and Additional Info:**

Version of Tesseract used: tesseract v5.0.0-alpha.20201127

The only parameters that were used are only the page\_separator and english

Language flag.

OS: Windows 10, Version 10.0.18363.1621

Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:**

Medium

**Rationale:**

As long as a human can easily read the image, we expect Tesseract to OCR it correctly. Suggesting a Medium severity due to the specificity of this image format and the combination of the two parameters.

Please see the attached files in the “Question 2 Attachments” folder.

**Appendix 1.3: Q2 - Page Separator Bug Report (3)**

**When using the page\_separator condition with double quotes and/or backlash characters, some of the output might be discarded.**

**Description:**

Tesseract OCR allows us to choose a custom separator using the **page\_separator=”string”** configuration. When the chosen string contains double quotes and/or backlash characters, due to escaping some of the output might be discarded. The lack of consistency is due to the two options of using the page\_separator condition (with or without double quotes).

**Steps to reproduce:**

1. Run Tesseract using the following command:

**tesseract imageFileName outputFileName1 -c page\_separator=”S”**

for S.

1. Run Tesseract using the following command:

**tesseract imageFileName outputFileName2 -c page\_separator=S**

for S.

1. Inspect the output files: Notice that there are differences according to escaping.

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameters that were used are page\_separator and the language flag for English.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** Medium

There is some failure but with a reasonable work around.

**Appendix 1.4: Q2 - Page Separator Bug Report (4)**

**When using the page\_separator condition with a vertical bar and without double quotes, an error occurs.**

**Description:**

Tesseract OCR allows us to choose a custom separator using the **page\_separator=”string”** or **page\_separator=string** configuration. When the chosen string is the vertical bar “|”, there are two different outcomes when using with and without the double quotes. For **page\_separator=|** an error occurs; **page\_separator=”|”** works as expected.

**Steps to reproduce:**

1. Run Tesseract using the following command:

**tesseract imageFileName outputFileName -c page\_separator=|**

1. Inspect the error message .
2. Run Tesseract using the following command:

**tesseract imageFileName outputFileName -c page\_separator=”|”**

1. Inspect the output file: the vertical bar is the separator as expected.

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameters that were used are page\_separator and the language flag for English.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** Medium

There is some failure but with a reasonable work around.

**Appendix 2.1: Q4.3.4 - Bug Report (1)**

**Tesseract couldn’t correctly recognize Spanish characters (Spanish language and characters whitelist parameters are specified).**

**Description:**

Tesseract OCR allows us to choose a language parameter and characters whiteslist parameter using **-l <language>** and **tessedit\_char\_whitelist=<some string>** configuration.

when the chosen language is “spa” (Spanish) and the chosen characters whitelist is “jyRVvN” and Tesseract given an image containing only the string "ñ prueba simple Ñ" (.png format, Arial font, size 40), tesseract characters recognition is badly mistaken and results in an incorrect output.

**Steps to reproduce:**

1. Run Tesseract using the following command:

**Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_311 -l spa -c tessedit\_char\_whitelist=jyRVvN**

1. Inspect the suited incorrect output file.

**Expected results:** empty output file.

**Actual results:** output file containing the string: “ N N j N N N”

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameters that were used are tessedit\_char\_whitelist and the language flag for Spanish.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** High.

Incorrect recognition of spanish characters in a simple sentence with reasonable font size.

**Appendix 2.2: Q4.3.4 - Bug Report (2)**

**Tesseract couldn’t correctly recognize concatenation of special characters.**

**Description:**

When given an image containing only a concatenation of special characters (.png format, Arial font, size 20) with no configuration parameters specified, Tesseract OCR characters recognition is badly mistaken and results in an incorrect output.

**Steps to reproduce:**

1. Run Tesseract using the following command:

**Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_12**

1. Inspect the suited incorrect output file.

**Expected results:** output file containing the string: “~,`!@#$%^&\*()-\_+={}[]|\\/:;"<>.?\” .

**Actual results:** output file containing the string: “~,1@#S%%84()-\_ += GI]Vii"<>.?"”.

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. OS: Windows 10, Version 10.0.18363.1621
3. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** High.

It seems Tesseract OCR is having trouble recognizing special characters correctly (and especially a concations of several of them) , the issue repeats on other tests containing special characters and a concatenation of them as well.

**Appendix 2.3: Q4.3.4 - Bug Report (3)**

**Characters blacklist configuration parameter of Tesseract is not working when used with Hebrew letters.**

**Description:**

Tesseract OCR allows us to choose a language parameter and characters blacklist parameter using **-l <language>** and **tessedit\_char\_blacklist=<some string>** configuration.

when the chosen language is “heb” (Hebrew), the chosen characters blacklist is “טש” and Tesseract given an image containing only the string "טסט פשוט" (.png format, Arial font, size 40), tesseract does not avoid the characters specified on the blacklist and results in an incorrect output.

**Steps to reproduce:**

1. Run Tesseract using the following command:

**Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_240 -l heb -c tessedit\_char\_blacklist=טש**

1. Inspect the suited incorrect output file.

**Expected results:** output file containing the string: “ופ ס” (inverted due to tesseract output of hebrew letters - not considered a bug).

**Actual results:** output file containing the string: “טושפ טסט” (inverted due to tesseract output of hebrew letters - not considered a bug).

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameters that were used are tessedit\_char\_blacklist and the language flag for Hebrew.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** High.

It seems Tesseract OCR is having trouble correctly considering blacklist Hebrew characters; this issue repeats on other tests (like 238 - 242).

**Appendix 2.4: Q4.3.4 - Bug Report (4)**

**when used as page separators, Spanish / Swedish letters are not printed properly by Tesseract OCR.**

**Description:**

Tesseract OCR allows us to choose a language parameter and custom page separator using the **-l <language>** and **page\_separator=”string”** or **page\_separator=string** configurations. When the chosen language is “swe” (Swedish) and the chosen page separator string is the swedish letter “Å” or the chosen language is “spa” (Spanish) and the chosen page separator string is the spanish letter “ñ”, the page separator printed on the output file incorrectly is “?”. We suspect the same for other special letters (different from English letters) of Spanish (Ñ) and Swedish (å ä ö Ä Ö).

**Steps to reproduce:**

1. Run Tesseract using the following command:

**Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_332 -l swe -c page\_separator=Å**

1. Run Tesseract using the following command:

**Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_333 -l spa -c page\_separator=ñ**

1. Inspect the suited incorrect output files.

**Expected results:** output file containing the string: “Simple Test” and the Swedish letter “Å” / the Spanish letter “ñ” as page separator.

**Actual results:** output file containing the string: “Simple Test” and the letter “?” as page separator.

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameters that were used are page\_separator and the language flag for Swedish / Spanish.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU
5. The input image contains only the sentence “Simple Test” written in Arial font, size 40 and .png format.

**Suggested Severity:** High.

It seems Tesseract OCR couldn’t print special letters (different from English letters) of Spanish and Swedish as page separators.

**Appendix 2.5: Q4.3.4 - Bug Report (5)**

**Tesseract OCR couldn’t recognize a single letter positioned on the image’s corners or center.**

**Description:**

when the input image given to tesseract contains only a single English letter (“A”) positioned

at a corner of the picture (.png format, Arial font, size 40), Tesseract OCR doesn’t recognize the letter and outputs an empty output file. This issue repeated on every corner of the image and on it’s center (see tests number 212 - 220).

**Steps to reproduce:**

1. Run Tesseract using a command suited for any test number from 212 to 220, for example test number 212:

**Tesseract images\212\_testType\_position\_Arial\_40.png output/test\_212 -l eng**

1. Inspect the suited incorrect output files.

**Expected results:** output file containing a single character: “A”.

**Actual results:** empty output file.

**Setup and Additional Information:**

1. Version of Tesseract used: tesseract v5.0.0-alpha.20201127
2. The only parameter that was used is the language flag for English.
3. OS: Windows 10, Version 10.0.18363.1621
4. Executed on HP 340s G7 2D220EA, i5-1035G1 CPU

**Suggested Severity:** High.

It seems Tesseract OCR is having trouble recognizing a single character written in a standard font type and size positioned at any of it’s corners or center.

**Appendix 2.6: Q4.3.4 - Bug Table**

|  |  |  |
| --- | --- | --- |
| **Title for the bug** | **Test number** | **Tesseract command to run** |
| Added "." which is not in the text. | 4 | Tesseract images\4\_testType\_file\_format\_Arial\_20.jpeg output/test\_4 -l eng |
| Added "." which is not in the text. | 6 | Tesseract images\6\_testType\_file\_format\_Arial\_20.jpg output/test\_6 -l eng |
| Recognized "ן" as "ו" in Hebrew | 8 | Tesseract images\8\_testType\_sub\_lang\_Arial\_20.png output/test\_8 -l heb |
| Recognized "c" as "e" in swedish | 9 | Tesseract images\9\_testType\_sub\_lang\_Arial\_20.png output/test\_9 -l swe |
| Recognized "ñ" as "R" in Spanish | 10 | Tesseract images\10\_testType\_sub\_lang\_Arial\_20.png output/test\_10 -l spa |
| Can't identify special characters properly. | 12 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_12 |
| Didn't detect Hebrew letters | 13 | Tesseract images\13\_testType\_sub\_lang\_Arial\_20.png output/test\_13 -l eng+heb |
| Didn't detect Swedish letters | 14 | Tesseract images\14\_testType\_sub\_lang\_Arial\_20.png output/test\_14 -l eng+swe |
| Recognized "ñ" as "R" in Spanish | 15 | Tesseract images\15\_testType\_sub\_lang\_Arial\_20.png output/test\_15 -l eng+spa |
| Can't identify special characters properly. | 17 | Tesseract images\17\_testType\_sub\_lang\_Arial\_20.png output/test\_17 -l eng |
| Recognized "ן" as "ו" in Hebrew | 18 | Tesseract images\18\_testType\_sub\_lang\_Arial\_20.png output/test\_18 -l heb+swe |
| Recognized "ן" as "ו" in Hebrew | 19 | Tesseract images\19\_testType\_sub\_lang\_Arial\_20.png output/test\_19 -l heb+spa |
| Recognized "ן" as "ו" in Hebrew | 20 | Tesseract images\20\_testType\_sub\_lang\_Arial\_20.png output/test\_20 -l heb |
| Recognized "ן" as "ו" in Hebrew | 21 | Tesseract images\21\_testType\_sub\_lang\_Arial\_20.png output/test\_21 -l heb |
| Recognized "c" as "e" in swedish | 22 | Tesseract images\22\_testType\_sub\_lang\_Arial\_20.png output/test\_22 -l swe+spa |
| Recognized "c" as "e" in swedish | 23 | Tesseract images\23\_testType\_sub\_lang\_Arial\_20.png output/test\_23 -l swe |
| Recognized "c" as "e" in swedish | 24 | Tesseract images\24\_testType\_sub\_lang\_Arial\_20.png output/test\_24 -l swe |
| Recognized "ñ" as "R" in Spanish | 25 | Tesseract images\25\_testType\_sub\_lang\_Arial\_20.png output/test\_25 -l spa |
| Recognized "ñ" as "R" in Spanish | 26 | Tesseract images\26\_testType\_sub\_lang\_Arial\_20.png output/test\_26 -l spa |
| Can't identify special characters properly. | 27 | Tesseract images\27\_testType\_sub\_lang\_Arial\_20.png output/test\_27 |
| Didn't detect Hebrew letters | 28 | Tesseract images\28\_testType\_sub\_lang\_Arial\_20.png output/test\_28 -l eng+heb+swe |
| Didn't detect Hebrew letters | 29 | Tesseract images\29\_testType\_sub\_lang\_Arial\_20.png output/test\_29 -l eng+heb+spa |
| Didn't detect Hebrew letters | 30 | Tesseract images\30\_testType\_sub\_lang\_Arial\_20.png output/test\_30 -l eng+heb |
| Didn't detect Hebrew letters | 31 | Tesseract images\31\_testType\_sub\_lang\_Arial\_20.png output/test\_31 -l eng+heb |
| Recognized "c" as "e" in swedish | 32 | Tesseract images\32\_testType\_sub\_lang\_Arial\_20.png output/test\_32 -l eng+swe+spa |
| Recognized "c" as "e" in swedish | 33 | Tesseract images\33\_testType\_sub\_lang\_Arial\_20.png output/test\_33 -l eng+swe |
| Didn't detect Swedish letters | 34 | Tesseract images\34\_testType\_sub\_lang\_Arial\_20.png output/test\_34 -l eng+swe |
| Recognized "ñ" as "R" in Spanish | 35 | Tesseract images\35\_testType\_sub\_lang\_Arial\_20.png output/test\_35 -l eng+spa |
| Recognized "ñ" as "R" in Spanish | 36 | Tesseract images\36\_testType\_sub\_lang\_Arial\_20.png output/test\_36 -l eng+spa |
| Can't identify special characters properly. | 37 | Tesseract images\37\_testType\_sub\_lang\_Arial\_20.png output/test\_37 -l eng |
| Recognized "ן" as "ו" in Hebrew | 38 | Tesseract images\38\_testType\_sub\_lang\_Arial\_20.png output/test\_38 -l heb+swe+spa |
| Recognized "ן" as "ו" in Hebrew | 39 | Tesseract images\39\_testType\_sub\_lang\_Arial\_20.png output/test\_39 -l heb+swe |
| Recognized "ן" as "ו" in Hebrew | 40 | Tesseract images\40\_testType\_sub\_lang\_Arial\_20.png output/test\_40 -l heb+swe |
| Recognized "ן" as "ו" in Hebrew | 41 | Tesseract images\41\_testType\_sub\_lang\_Arial\_20.png output/test\_41 -l heb+spa |
| Recognized "ן" as "ו" in Hebrew | 42 | Tesseract images\42\_testType\_sub\_lang\_Arial\_20.png output/test\_42 -l heb+spa |
| Recognized "ן" as "ו" in Hebrew | 43 | Tesseract images\43\_testType\_sub\_lang\_Arial\_20.png output/test\_43 -l heb |
| Recognized "c" as "e" in swedish | 44 | Tesseract images\44\_testType\_sub\_lang\_Arial\_20.png output/test\_44 -l swe+spa |
| Recognized "c" as "e" in swedish | 45 | Tesseract images\45\_testType\_sub\_lang\_Arial\_20.png output/test\_45 -l swe+spa |
| Recognized "c" as "e" in swedish | 46 | Tesseract images\46\_testType\_sub\_lang\_Arial\_20.png output/test\_46 -l swe |
| Recognized "ñ" as "R" in Spanish | 47 | Tesseract images\47\_testType\_sub\_lang\_Arial\_20.png output/test\_47 -l spa |
| Recognized "ñ" as "R" in Spanish | 48 | Tesseract images\48\_testType\_sub\_lang\_Arial\_20.png output/test\_48 -l eng+heb+swe+spa |
| Didn't detect Hebrew letters | 49 | Tesseract images\49\_testType\_sub\_lang\_Arial\_20.png output/test\_49 -l eng+heb+swe |
| Didn't detect Hebrew letters | 50 | Tesseract images\50\_testType\_sub\_lang\_Arial\_20.png output/test\_50 -l eng+heb+swe |
| Didn't detect Hebrew letters | 51 | Tesseract images\51\_testType\_sub\_lang\_Arial\_20.png output/test\_51 -l eng+heb+spa |
| Didn't detect Hebrew letters | 52 | Tesseract images\52\_testType\_sub\_lang\_Arial\_20.png output/test\_52 -l eng+heb+spa |
| Can't identify special characters properly. | 53 | Tesseract images\53\_testType\_sub\_lang\_Arial\_20.png output/test\_53 -l eng+heb |
| Recognized "ñ" as "R" in Spanish | 54 | Tesseract images\54\_testType\_sub\_lang\_Arial\_20.png output/test\_54 -l eng+swe+spa |
| Recognized "c" as "e" in swedish | 55 | Tesseract images\55\_testType\_sub\_lang\_Arial\_20.png output/test\_55 -l eng+swe+spa |
| Recognized "c" as "e" in swedish | 56 | Tesseract images\56\_testType\_sub\_lang\_Arial\_20.png output/test\_56 -l eng+swe |
| Recognized "ñ" as "R" in Spanish | 57 | Tesseract images\57\_testType\_sub\_lang\_Arial\_20.png output/test\_57 -l eng+spa |
| Recognized "ן" as "ו" in Hebrew | 58 | Tesseract images\58\_testType\_sub\_lang\_Arial\_20.png output/test\_58 -l heb+swe+spa |
| Recognized "ן" as "ו" in Hebrew | 59 | Tesseract images\59\_testType\_sub\_lang\_Arial\_20.png output/test\_59 -l heb+swe+spa |
| Recognized "ן" as "ו" in Hebrew | 60 | Tesseract images\60\_testType\_sub\_lang\_Arial\_20.png output/test\_60 -l heb+swe |
| Recognized "ן" as "ו" in Hebrew | 61 | Tesseract images\61\_testType\_sub\_lang\_Arial\_20.png output/test\_61 -l heb+spa |
| Recognized "c" as "e" in Swedish | 62 | Tesseract images\62\_testType\_sub\_lang\_Arial\_20.png output/test\_62 -l swe+spa |
| Recognized "c" as "e" in Swedish | 63 | Tesseract images\63\_testType\_sub\_lang\_Arial\_20.png output/test\_63 -l eng+heb+swe+spa |
| Recognized "c" as "e" in Swedish | 64 | Tesseract images\64\_testType\_sub\_lang\_Arial\_20.png output/test\_64 -l eng+heb+swe+spa |
| Recognized "c" as "e" in Swedish | 65 | Tesseract images\65\_testType\_sub\_lang\_Arial\_20.png output/test\_65 -l eng+heb+swe |
| Didn't detect Hebrew letters | 66 | Tesseract images\66\_testType\_sub\_lang\_Arial\_20.png output/test\_66 -l eng+heb+spa |
| Recognized "ñ" as "R" in Spanish | 67 | Tesseract images\67\_testType\_sub\_lang\_Arial\_20.png output/test\_67 -l eng+swe+spa |
| Recognized "ñ" as "R" in Spanish | 68 | Tesseract images\68\_testType\_sub\_lang\_Arial\_20.png output/test\_68 -l heb+swe+spa |
| Recognized "c" as "e" in Swedish | 69 | Tesseract images\69\_testType\_sub\_lang\_Arial\_20.png output/test\_69 -l eng+heb+swe+spa |
| Didn't recognize letters of size 8 | 70 | Tesseract images\70\_testType\_fonts\_Aharoni Bold\_8.png output/test\_70 -l heb |
| Didn't recognize letters of size 8 | 71 | Tesseract images\71\_testType\_fonts\_Aharoni Bold\_8.png output/test\_71 -l eng |
| Didn't detect Hebrew letters | 72 | Tesseract images\72\_testType\_fonts\_Aharoni Bold\_72.png output/test\_72 -l heb |
| Didn't recognize letters of size 72 | 73 | Tesseract images\73\_testType\_fonts\_Aharoni Bold\_72.png output/test\_73 -l eng |
| Recognized "ן" as "ו" in Hebrew | 74 | Tesseract images\74\_testType\_fonts\_Arial\_8.png output/test\_74 -l heb |
| Didn't recognize letters of size 8 | 75 | Tesseract images\75\_testType\_fonts\_Arial\_8.png output/test\_75 -l eng |
| Recognized "ן" as "ו" in Hebrew | 76 | Tesseract images\76\_testType\_fonts\_Arial\_72.png output/test\_76 -l heb |
| Extra Z recognized | 77 | Tesseract images\77\_testType\_fonts\_Arial\_72.png output/test\_77 -l eng |
| Didn't recognize letters of size 8 | 78 | Tesseract images\78\_testType\_fonts\_Arial Black\_8.png output/test\_78 -l eng |
| Didn't recognize letters of size 8 | 80 | Tesseract images\80\_testType\_fonts\_Bahnschrift\_8.png output/test\_80 -l eng |
| Extra Z recognized | 81 | Tesseract images\81\_testType\_fonts\_Bahnschrift\_72.png output/test\_81 -l eng |
| Didn't recognize letters of size 8 | 82 | Tesseract images\82\_testType\_fonts\_Calibri\_8.png output/test\_82 -l heb |
| Didn't recognize letters of size 8 | 83 | Tesseract images\83\_testType\_fonts\_Calibri\_8.png output/test\_83 -l eng |
| Recognized "ן" as "ו" in Hebrew | 84 | Tesseract images\84\_testType\_fonts\_Calibri\_72.png output/test\_84 -l heb |
| Didn't recognize letters of size 72 | 85 | Tesseract images\85\_testType\_fonts\_Calibri\_72.png output/test\_85 -l eng |
| Didn't recognize letters of size 8 | 86 | Tesseract images\86\_testType\_fonts\_Cambria\_8.png output/test\_86 -l eng |
| Didn't recognize letters of size 8 | 88 | Tesseract images\88\_testType\_fonts\_Cambria Math\_8.png output/test\_88 -l eng |
| Didn't recognize letters of size 72 | 89 | Tesseract images\89\_testType\_fonts\_Cambria Math\_72.png output/test\_89 -l eng |
| Didn't recognize letters of size 8 | 90 | Tesseract images\90\_testType\_fonts\_Candara\_8.png output/test\_90 -l eng |
| Didn't recognize letters of size 72 | 91 | Tesseract images\91\_testType\_fonts\_Candara\_72.png output/test\_91 -l eng |
| Didn't recognize letters of size 8 | 92 | Tesseract images\92\_testType\_fonts\_Comic Sans MS\_8.png output/test\_92 -l eng |
| Extra Z recognized | 93 | Tesseract images\93\_testType\_fonts\_Comic Sans MS\_72.png output/test\_93 -l eng |
| Didn't recognize letters of size 8 | 94 | Tesseract images\94\_testType\_fonts\_Consolas\_8.png output/test\_94 -l eng |
| Didn't recognize letters of size 72 | 95 | Tesseract images\95\_testType\_fonts\_Consolas\_72.png output/test\_95 -l eng |
| Didn't recognize letters of size 8 | 96 | Tesseract images\96\_testType\_fonts\_Constantia\_8.png output/test\_96 -l eng |
| Extra Z recognized | 97 | Tesseract images\97\_testType\_fonts\_Constantia\_72.png output/test\_97 -l eng |
| Didn't recognize letters of size 8 | 98 | Tesseract images\98\_testType\_fonts\_Corbel\_8.png output/test\_98 -l eng |
| Didn't recognize letters of size 72 | 99 | Tesseract images\99\_testType\_fonts\_Corbel\_72.png output/test\_99 -l eng |
| Didn't recognize letters of size 8 | 100 | Tesseract images\100\_testType\_fonts\_Courier New\_8.png output/test\_100 -l heb |
| Didn't recognize letters of size 8 | 101 | Tesseract images\101\_testType\_fonts\_Courier New\_8.png output/test\_101 -l eng |
| Recognized "ן" as "ו" in Hebrew | 102 | Tesseract images\102\_testType\_fonts\_Courier New\_72.png output/test\_102 -l heb |
| Recognized "q" as "g" in English | 103 | Tesseract images\103\_testType\_fonts\_Courier New\_72.png output/test\_103 -l eng |
| Didn't recognize letters of size 8 | 104 | Tesseract images\104\_testType\_fonts\_David\_8.png output/test\_104 -l heb |
| Didn't recognize letters of size 8 | 105 | Tesseract images\105\_testType\_fonts\_David\_8.png output/test\_105 -l eng |
| Recognized "ן" as "ו" in Hebrew | 106 | Tesseract images\106\_testType\_fonts\_David\_72.png output/test\_106 -l heb |
| Recognized "q" as "g" in English | 107 | Tesseract images\107\_testType\_fonts\_David\_72.png output/test\_107 -l eng |
| Didn't recognize letters of size 8 | 108 | Tesseract images\108\_testType\_fonts\_David Bold\_8.png output/test\_108 -l heb |
| Didn't recognize letters of size 8 | 109 | Tesseract images\109\_testType\_fonts\_David Bold\_8.png output/test\_109 -l eng |
| Didn't detect Hebrew letters | 110 | Tesseract images\110\_testType\_fonts\_David Bold\_72.png output/test\_110 -l heb |
| Didn't recognize letters of size 72 | 111 | Tesseract images\111\_testType\_fonts\_David Bold\_72.png output/test\_111 -l eng |
| Didn't recognize letters of size 8 | 112 | Tesseract images\112\_testType\_fonts\_Ebrima\_8.png output/test\_112 -l eng |
| Recognized "q" as "a" in English | 113 | Tesseract images\113\_testType\_fonts\_Ebrima\_72.png output/test\_113 -l eng |
| Didn't recognize letters of size 8 | 114 | Tesseract images\114\_testType\_fonts\_Forte\_8.png output/test\_114 -l eng |
| Recognized "q" as "a" in English | 115 | Tesseract images\115\_testType\_fonts\_Forte\_72.png output/test\_115 -l eng |
| Didn't recognize letters of size 8 | 116 | Tesseract images\116\_testType\_fonts\_Franklin Gothic Medium\_8.png output/test\_116 -l eng |
| Didn't recognize letters of size 8 | 118 | Tesseract images\118\_testType\_fonts\_FrankRuehl\_8.png output/test\_118 -l heb |
| Didn't recognize letters of size 8 | 119 | Tesseract images\119\_testType\_fonts\_FrankRuehl\_8.png output/test\_119 -l eng |
| Recognized "ן" as "ו" in Hebrew | 120 | Tesseract images\120\_testType\_fonts\_FrankRuehl\_72.png output/test\_120 -l heb |
| Recognized "l" as "!" in English | 121 | Tesseract images\121\_testType\_fonts\_FrankRuehl\_72.png output/test\_121 -l eng |
| Didn't recognize letters of size 8 | 122 | Tesseract images\122\_testType\_fonts\_Gabriola\_8.png output/test\_122 -l eng |
| Didn't recognize letters of size 72 | 123 | Tesseract images\123\_testType\_fonts\_Gabriola\_72.png output/test\_123 -l eng |
| Didn't recognize letters of size 8 | 124 | Tesseract images\124\_testType\_fonts\_Gadugi\_8.png output/test\_124 -l eng |
| Recognized "q" as "a" in English | 125 | Tesseract images\125\_testType\_fonts\_Gadugi\_72.png output/test\_125 -l eng |
| Didn't recognize letters of size 8 | 126 | Tesseract images\126\_testType\_fonts\_Georgia\_8.png output/test\_126 -l eng |
| Didn't recognize letters of size 8 | 128 | Tesseract images\128\_testType\_fonts\_Gisha\_8.png output/test\_128 -l heb |
| Didn't recognize letters of size 8 | 129 | Tesseract images\129\_testType\_fonts\_Gisha\_8.png output/test\_129 -l eng |
| Recognized "ן" as "ו" in Hebrew | 130 | Tesseract images\130\_testType\_fonts\_Gisha\_72.png output/test\_130 -l heb |
| Recognized "q" as "a" in English | 131 | Tesseract images\131\_testType\_fonts\_Gisha\_72.png output/test\_131 -l eng |
| Didn't recognize letters of size 8 | 132 | Tesseract images\132\_testType\_fonts\_Gisha Bold\_8.png output/test\_132 -l heb |
| Didn't recognize letters of size 8 | 133 | Tesseract images\133\_testType\_fonts\_Gisha Bold\_8.png output/test\_133 -l eng |
| Didn't detect Hebrew letters | 134 | Tesseract images\134\_testType\_fonts\_Gisha Bold\_72.png output/test\_134 -l heb |
| Recognized "q" as "a" in English | 135 | Tesseract images\135\_testType\_fonts\_Gisha Bold\_72.png output/test\_135 -l eng |
| Didn't recognize letters of size 8 | 136 | Tesseract images\136\_testType\_fonts\_Impact\_8.png output/test\_136 -l eng |
| Didn't recognize letters of size 72 | 137 | Tesseract images\137\_testType\_fonts\_Impact\_72.png output/test\_137 -l eng |
| Didn't recognize letters of size 8 | 138 | Tesseract images\138\_testType\_fonts\_Ink Free\_8.png output/test\_138 -l eng |
| Didn't recognize letters of size 72 | 139 | Tesseract images\139\_testType\_fonts\_Ink Free\_72.png output/test\_139 -l eng |
| Didn't recognize letters of size 8 | 140 | Tesseract images\140\_testType\_fonts\_Levenim MT\_8.png output/test\_140 -l heb |
| Didn't recognize letters of size 8 | 141 | Tesseract images\141\_testType\_fonts\_Levenim MT\_8.png output/test\_141 -l eng |
| Didn't recognize letters of size 72 | 142 | Tesseract images\142\_testType\_fonts\_Levenim MT\_72.png output/test\_142 -l heb |
| Recognized "q" as "o" in English | 143 | Tesseract images\143\_testType\_fonts\_Levenim MT\_72.png output/test\_143 -l eng |
| Didn't recognize letters of size 8 | 144 | Tesseract images\144\_testType\_fonts\_Levenim MT Bold\_8.png output/test\_144 -l heb |
| Didn't recognize letters of size 8 | 145 | Tesseract images\145\_testType\_fonts\_Levenim MT Bold\_8.png output/test\_145 -l eng |
| Recognized "ן" as "ו" in Hebrew | 146 | Tesseract images\146\_testType\_fonts\_Levenim MT Bold\_72.png output/test\_146 -l heb |
| Extra Z recognized | 147 | Tesseract images\147\_testType\_fonts\_Levenim MT Bold\_72.png output/test\_147 -l eng |
| Didn't recognize letters of size 8 | 148 | Tesseract images\148\_testType\_fonts\_Lucida Console\_8.png output/test\_148 -l eng |
| Didn't recognize letters of size 72 | 149 | Tesseract images\149\_testType\_fonts\_Lucida Console\_72.png output/test\_149 -l eng |
| Didn't recognize letters of size 8 | 150 | Tesseract images\150\_testType\_fonts\_Lucida Sans Unicode\_8.png output/test\_150 -l heb |
| Didn't recognize letters of size 8 | 151 | Tesseract images\151\_testType\_fonts\_Lucida Sans Unicode\_8.png output/test\_151 -l eng |
| Recognized "ן" as "ו" in Hebrew | 152 | Tesseract images\152\_testType\_fonts\_Lucida Sans Unicode\_72.png output/test\_152 -l heb |
| Didn't recognize letters of size 72 | 153 | Tesseract images\153\_testType\_fonts\_Lucida Sans Unicode\_72.png output/test\_153 -l eng |
| Recognized "ן" as "ו" in Hebrew | 154 | Tesseract images\154\_testType\_fonts\_Microsoft Sans Serif\_8.png output/test\_154 -l heb |
| Didn't recognize letters of size 8 | 155 | Tesseract images\155\_testType\_fonts\_Microsoft Sans Serif\_8.png output/test\_155 -l eng |
| Didn't detect Hebrew letters | 156 | Tesseract images\156\_testType\_fonts\_Microsoft Sans Serif\_72.png output/test\_156 -l heb |
| Didn't recognize letters of size 8 | 158 | Tesseract images\158\_testType\_fonts\_Miriam\_8.png output/test\_158 -l heb |
| Didn't recognize letters of size 8 | 159 | Tesseract images\159\_testType\_fonts\_Miriam\_8.png output/test\_159 -l eng |
| Recognized "ן" as "ו" in Hebrew | 160 | Tesseract images\160\_testType\_fonts\_Miriam\_72.png output/test\_160 -l heb |
| Recognized "q" as "a" in English | 161 | Tesseract images\161\_testType\_fonts\_Miriam\_72.png output/test\_161 -l eng |
| Didn't recognize letters of size 8 | 162 | Tesseract images\162\_testType\_fonts\_Miriam Fixed\_8.png output/test\_162 -l heb |
| Didn't recognize letters of size 8 | 163 | Tesseract images\163\_testType\_fonts\_Miriam Fixed\_8.png output/test\_163 -l eng |
| Didn't recognize letters of size 72 | 164 | Tesseract images\164\_testType\_fonts\_Miriam Fixed\_72.png output/test\_164 -l heb |
| Recognized "q" as "g" in English | 165 | Tesseract images\165\_testType\_fonts\_Miriam Fixed\_72.png output/test\_165 -l eng |
| Didn't recognize letters of size 8 | 166 | Tesseract images\166\_testType\_fonts\_MS Gothic\_8.png output/test\_166 -l eng |
| Didn't recognize letters of size 72 | 167 | Tesseract images\167\_testType\_fonts\_MS Gothic\_72.png output/test\_167 -l eng |
| Didn't recognize letters of size 8 | 168 | Tesseract images\168\_testType\_fonts\_Narkisim\_8.png output/test\_168 -l heb |
| Didn't recognize letters of size 8 | 169 | Tesseract images\169\_testType\_fonts\_Narkisim\_8.png output/test\_169 -l eng |
| Recognized "ן" as "ו" in Hebrew | 170 | Tesseract images\170\_testType\_fonts\_Narkisim\_72.png output/test\_170 -l heb |
| Recognized "q" as "a" in English | 171 | Tesseract images\171\_testType\_fonts\_Narkisim\_72.png output/test\_171 -l eng |
| Didn't recognize letters of size 8 | 172 | Tesseract images\172\_testType\_fonts\_Rod\_8.png output/test\_172 -l heb |
| Didn't recognize letters of size 8 | 173 | Tesseract images\173\_testType\_fonts\_Rod\_8.png output/test\_173 -l eng |
| Didn't recognize letters of size 72 | 174 | Tesseract images\174\_testType\_fonts\_Rod\_72.png output/test\_174 -l heb |
| Recognized "q" as "g" in English | 175 | Tesseract images\175\_testType\_fonts\_Rod\_72.png output/test\_175 -l eng |
| Didn't recognize letters of size 8 | 176 | Tesseract images\176\_testType\_fonts\_Segoe Print\_8.png output/test\_176 -l eng |
| Extra Z recognized | 177 | Tesseract images\177\_testType\_fonts\_Segoe Print\_72.png output/test\_177 -l eng |
| Didn't recognize letters of size 8 | 178 | Tesseract images\178\_testType\_fonts\_Segoe Script\_8.png output/test\_178 -l eng |
| Didn't recognize letters of size 72 | 179 | Tesseract images\179\_testType\_fonts\_Segoe Script\_72.png output/test\_179 -l eng |
| Recognized "ן" as "ו" in Hebrew | 180 | Tesseract images\180\_testType\_fonts\_Segoe UI\_8.png output/test\_180 -l heb |
| Didn't recognize letters of size 8 | 181 | Tesseract images\181\_testType\_fonts\_Segoe UI\_8.png output/test\_181 -l eng |
| Recognized "ן" as "ו" in Hebrew | 182 | Tesseract images\182\_testType\_fonts\_Segoe UI\_72.png output/test\_182 -l heb |
| Recognized "q" as "a" in English | 183 | Tesseract images\183\_testType\_fonts\_Segoe UI\_72.png output/test\_183 -l eng |
| Didn't recognize letters of size 8 | 184 | Tesseract images\184\_testType\_fonts\_Tahoma\_8.png output/test\_184 -l heb |
| Didn't recognize letters of size 8 | 185 | Tesseract images\185\_testType\_fonts\_Tahoma\_8.png output/test\_185 -l eng |
| Recognized "ן" as "ו" in Hebrew | 186 | Tesseract images\186\_testType\_fonts\_Tahoma\_72.png output/test\_186 -l heb |
| Didn't recognize letters of size 8 | 188 | Tesseract images\188\_testType\_fonts\_Times New Roman\_8.png output/test\_188 -l heb |
| Didn't recognize letters of size 8 | 189 | Tesseract images\189\_testType\_fonts\_Times New Roman\_8.png output/test\_189 -l eng |
| Recognized "ן" as "ז" in Hebrew | 190 | Tesseract images\190\_testType\_fonts\_Times New Roman\_72.png output/test\_190 -l heb |
| Recognized "j" as "y" in English | 191 | Tesseract images\191\_testType\_fonts\_Times New Roman\_72.png output/test\_191 -l eng |
| Recognized "ן" as "ו" in Hebrew | 192 | Tesseract images\192\_testType\_font\_size\_Arial\_8.png output/test\_192 -l heb |
| Didn't recognize letters of size 8 | 193 | Tesseract images\193\_testType\_font\_size\_Arial\_8.png output/test\_193 -l eng |
| Recognized "ן" as "ו" in Hebrew | 194 | Tesseract images\194\_testType\_font\_size\_Arial\_9.png output/test\_194 -l heb |
| Recognized "l" as "!" in English | 195 | Tesseract images\195\_testType\_font\_size\_Arial\_9.png output/test\_195 -l eng |
| Recognized "ן" as "ו" in Hebrew | 196 | Tesseract images\196\_testType\_font\_size\_Arial\_32.png output/test\_196 -l heb |
| Recognized "q" as "a" in English | 197 | Tesseract images\197\_testType\_font\_size\_Arial\_32.png output/test\_197 -l eng |
| Recognized "ן" as "ו" in Hebrew | 198 | Tesseract images\198\_testType\_font\_size\_Arial\_71.png output/test\_198 -l heb |
| Recognized "q" as "a" in English | 199 | Tesseract images\199\_testType\_font\_size\_Arial\_71.png output/test\_199 -l eng |
| Recognized "ן" as "ו" in Hebrew | 200 | Tesseract images\200\_testType\_font\_size\_Arial\_72.png output/test\_200 -l heb |
| Extra Z recognized | 201 | Tesseract images\201\_testType\_font\_size\_Arial\_72.png output/test\_201 -l eng |
| Didn't Recognize single character at corner | 202 | Tesseract images\202\_testType\_length\_of\_text\_Arial\_20.png output/test\_202 -l eng |
| Didn't recognize picture that is full of letters | 205 | Tesseract images\205\_testType\_length\_of\_text\_Arial\_20.png output/test\_205 -l eng |
| Can't identify letters separated with spaces | 207 | Tesseract images\207\_testType\_space\_Arial\_20.png output/test\_207 -l eng |
| Can't identify letters separated with spaces | 208 | Tesseract images\208\_testType\_space\_Arial\_20.png output/test\_208 -l eng |
| Can't identify letters separated with spaces | 209 | Tesseract images\209\_testType\_space\_Arial\_20.png output/test\_209 -l eng |
| Can't identify letters separated with spaces | 210 | Tesseract images\210\_testType\_space\_Arial\_20.png output/test\_210 -l eng |
| Can't identify letters separated with spaces | 211 | Tesseract images\211\_testType\_space\_Arial\_20.png output/test\_211 -l eng |
| Didn't Recognize single character at corner | 212 | Tesseract images\212\_testType\_position\_Arial\_40.png output/test\_212 -l eng |
| Didn't Recognize single character at corner | 213 | Tesseract images\213\_testType\_position\_Arial\_40.png output/test\_213 -l eng |
| Didn't Recognize single character at corner | 214 | Tesseract images\214\_testType\_position\_Arial\_40.png output/test\_214 -l eng |
| Didn't Recognize single character at corner | 215 | Tesseract images\215\_testType\_position\_Arial\_40.png output/test\_215 -l eng |
| Didn't Recognize single character at corner | 216 | Tesseract images\216\_testType\_position\_Arial\_40.png output/test\_216 -l eng |
| Didn't Recognize single character at corner | 217 | Tesseract images\217\_testType\_position\_Arial\_40.png output/test\_217 -l eng |
| Didn't Recognize single character at corner | 218 | Tesseract images\218\_testType\_position\_Arial\_40.png output/test\_218 -l eng |
| Didn't Recognize single character at corner | 219 | Tesseract images\219\_testType\_position\_Arial\_40.png output/test\_219 -l eng |
| Didn't Recognize single character at corner | 220 | Tesseract images\220\_testType\_position\_Arial\_40.png output/test\_220 -l eng |
| Can't identify letters which is rotated | 221 | Tesseract images\221\_testType\_position\_Arial\_40.png output/test\_221 -l eng |
| Can't identify letters which is rotated | 222 | Tesseract images\222\_testType\_position\_Arial\_40.png output/test\_222 -l eng |
| Can't identify letters which is rotated | 223 | Tesseract images\223\_testType\_position\_Arial\_40.png output/test\_223 -l eng |
| Can't identify letters which is rotated | 224 | Tesseract images\224\_testType\_position\_Arial\_40.png output/test\_224 -l eng |
| Didn't detect Hebrew letters | 225 | Tesseract images\225\_testType\_line\_spaces\_Arial\_20.png output/test\_225 -l eng+heb |
| Didn't detect Hebrew letters | 226 | Tesseract images\226\_testType\_line\_spaces\_Arial\_20.png output/test\_226 -l eng+heb |
| Recognized "ן" as "ו" in Hebrew | 227 | Tesseract images\227\_testType\_line\_spaces\_Arial\_20.png output/test\_227 -l eng+heb |
| Recognized "ן" as "ו" in Hebrew | 228 | Tesseract images\228\_testType\_line\_spaces\_Arial\_20.png output/test\_228 -l eng+heb |
| Recognized "ñ" as "Á" in Spanish | 232 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_232 -l spa |
| Blacklist of Hebrew letters doesn't work | 238 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_238 -l heb -c tessedit\_char\_blacklist=אבגדהוזחטיכלמנסעפצקרשתםףץן |
| Blacklist of Hebrew letters doesn't work | 240 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_240 -l heb -c tessedit\_char\_blacklist=טש |
| Blacklist of Hebrew letters doesn't work | 242 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_242 -l heb -c tessedit\_char\_blacklist=אצקרתםףץןטש |
| Blacklist of Swedish letters doesn't work | 243 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_243 -l swe -c tessedit\_char\_blacklist=bcdfghjklmnpqrstvwxzaeiouyåäöÅÄÖABCDEFGHIJKLMNOPQRSTUVWXYZ |
| Blacklist of Swedish letters doesn't work | 245 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_245 -l swe -c tessedit\_char\_blacklist=etäöÅ |
| Blacklist of Swedish letters doesn't work | 247 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_247 -l swe -c tessedit\_char\_blacklist=etäöÅAfoPg |
| Blacklist of Spanish letters doesn't work | 248 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_248 -l spa -c tessedit\_char\_blacklist=ñabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZÑ |
| Blacklist of Spanish letters doesn't work | 249 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_249 -l spa -c tessedit\_char\_blacklist= |
| Blacklist of Spanish letters doesn't work | 250 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_250 -l spa -c tessedit\_char\_blacklist=ñeip |
| Blacklist of Spanish letters doesn't work | 251 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_251 -l spa -c tessedit\_char\_blacklist=jyRVvN |
| Blacklist of Spanish letters doesn't work | 252 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_252 -l spa -c tessedit\_char\_blacklist=jyRVvNñeip |
| Blacklist of digits doesn't work | 253 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_253 -c tessedit\_char\_blacklist=0123456789 |
| Blacklist of digits doesn't work | 255 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_255 -c tessedit\_char\_blacklist=25689 |
| Blacklist of digits doesn't work | 257 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_257 -c tessedit\_char\_blacklist=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ25689 |
| Blacklist of special characters doesn't work | 258 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_258 -c tessedit\_char\_blacklist=~,`!@#$%^&\*()-\_+={}[]|\\/:;"<>.?\ |
| Blacklist of special characters doesn't work | 259 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_259 -c tessedit\_char\_blacklist= |
| Blacklist of special characters doesn't work | 260 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_260 -c tessedit\_char\_blacklist=~!$%^&(+=]? |
| Blacklist of special characters doesn't work | 261 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_261 -c tessedit\_char\_blacklist=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ25689 |
| Blacklist of special characters doesn't work | 262 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_262 -c tessedit\_char\_blacklist=~!$%^&(+=]?abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ |
| Blacklist of Hebrew letters doesn't work | 270 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_270 -l heb -c tessedit\_char\_blacklist=פשוט -c tessedit\_char\_unblacklist=טש |
| Blacklist of Swedish letters doesn't work | 275 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_275 -l swe -c tessedit\_char\_blacklist=ÄöåetÖ -c tessedit\_char\_unblacklist=eÄ |
| Blacklist of Swedish letters doesn't work | 277 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_277 -l swe -c tessedit\_char\_blacklist=ÄöåetÖ -c tessedit\_char\_unblacklist=eÄvmbHL |
| Blacklist of Spanish letters doesn't work | 278 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_278 -l spa -c tessedit\_char\_blacklist= -c tessedit\_char\_unblacklist=ñabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZÑ |
| Blacklist of Spanish letters doesn't work | 279 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_279 -l spa -c tessedit\_char\_blacklist=ñabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZÑ -c tessedit\_char\_unblacklist=ñabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZÑ |
| Blacklist of Spanish letters doesn't work | 280 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_280 -l spa -c tessedit\_char\_blacklist=Ñeiupb -c tessedit\_char\_unblacklist=ep |
| Blacklist of Spanish letters doesn't work | 281 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_281 -l spa -c tessedit\_char\_blacklist=ep -c tessedit\_char\_unblacklist=Ñeiupb |
| Blacklist of Spanish letters doesn't work | 282 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_282 -l spa -c tessedit\_char\_blacklist=Ñeiupb -c tessedit\_char\_unblacklist=eplQvMH |
| Blacklist of digits doesn't work | 285 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_285 -c tessedit\_char\_blacklist=20578 -c tessedit\_char\_unblacklist=27 |
| Blacklist of digits doesn't work | 287 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_287 -c tessedit\_char\_blacklist=20578 -c tessedit\_char\_unblacklist=abgHk27 |
| Blacklist of special characters doesn't work | 288 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_288 -c tessedit\_char\_blacklist= -c tessedit\_char\_unblacklist=~,`!@#$%^&\*()-\_+={}[]|\\/:;"<>.?\ |
| Blacklist of special characters doesn't work | 289 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_289 -c tessedit\_char\_blacklist=~,`!@#$%^&\*()-\_+={}[]|\\/:;"<>.?\  -c tessedit\_char\_unblacklist=~,`!@#$%^&\*()-\_+={}[]|\\/:;"<>.?\ |
| Blacklist of special characters doesn't work | 290 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_290 -c tessedit\_char\_blacklist=!#%)\_}[<? -c tessedit\_char\_unblacklist=!%}[? |
| Blacklist of special characters doesn't work | 291 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_291 -c tessedit\_char\_blacklist=!%}[? -c tessedit\_char\_unblacklist=!#%)\_}[<? |
| Blacklist of special characters doesn't work | 292 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_292 -c tessedit\_char\_blacklist=!#%)\_}[<? -c tessedit\_char\_unblacklist=!%}[?fghjkHGV |
| Didn't Recognize space | 293 | Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_293 -l eng -c tessedit\_char\_whitelist=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ |
| Empty whitelist bug | 294 | Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_294 -l eng -c tessedit\_char\_whitelist= |
| Didn't Recognize space | 295 | Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_295 -l eng -c tessedit\_char\_whitelist=etTi |
| Didn't Recognize space | 297 | Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_297 -l eng -c tessedit\_char\_whitelist=etTikPjGa |
| Whitelist of Hebrew letters doesn't work | 298 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_298 -l heb -c tessedit\_char\_whitelist=אבגדהוזחטיכלמנסעפצקרשתםףץן |
| Empty whitelist bug | 299 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_299 -l heb -c tessedit\_char\_whitelist= |
| Whitelist of Hebrew letters doesn't work | 300 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_300 -l heb -c tessedit\_char\_whitelist=טש |
| Whitelist of Hebrew letters doesn't work | 302 | Tesseract images\230\_testType\_simple\_tests\_Arial\_40.png output/test\_302 -l heb -c tessedit\_char\_whitelist=אצקרתםףץןטש |
| Whitelist of Swedish letters doesn't work | 303 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_303 -l swe -c tessedit\_char\_whitelist=bcdfghjklmnpqrstvwxzaeiouyåäöÅÄÖABCDEFGHIJKLMNOPQRSTUVWXYZ |
| Whitelist of Swedish letters doesn't work | 304 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_304 -l swe -c tessedit\_char\_whitelist= |
| Whitelist of Swedish letters doesn't work | 305 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_305 -l swe -c tessedit\_char\_whitelist=etäöÅ |
| Whitelist of Swedish letters doesn't work | 306 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_306 -l swe -c tessedit\_char\_whitelist=AfoPg |
| Whitelist of Swedish letters doesn't work | 307 | Tesseract images\231\_testType\_simple\_tests\_Arial\_40.png output/test\_307 -l swe -c tessedit\_char\_whitelist=etäöÅAfoPg |
| Whitelist of Spanish letters doesn't work | 308 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_308 -l spa -c tessedit\_char\_whitelist=ñabcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZÑ |
| Whitelist of Spanish letters doesn't work | 309 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_309 -l spa -c tessedit\_char\_whitelist= |
| Whitelist of Spanish letters doesn't work | 310 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_310 -l spa -c tessedit\_char\_whitelist=ñeip |
| Whitelist in Spanish recognize letters in an empty pic | 311 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_311 -l spa -c tessedit\_char\_whitelist=jyRVvN |
| Whitelist of Spanish letters doesn't work | 312 | Tesseract images\232\_testType\_simple\_tests\_Arial\_40.png output/test\_312 -l spa -c tessedit\_char\_whitelist=jyRVvNñeip |
| Whitelist of digits doesn't work | 314 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_314 -c tessedit\_char\_whitelist= |
| Whitelist of digits doesn't work | 315 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_315 -c tessedit\_char\_whitelist=25689 |
| Whitelist of digits doesn't work | 316 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_316 -c tessedit\_char\_whitelist=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ |
| Whitelist of digits doesn't work | 317 | Tesseract images\11\_testType\_sub\_lang\_Arial\_20.png output/test\_317 -c tessedit\_char\_whitelist=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ25689 |
| Whitelist of special characters doesn't work | 318 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_318 -c tessedit\_char\_whitelist=~,`!@#$%^&\*()-\_+={}[]|\\/:;"<>.?\ |
| Whitelist of special characters doesn't work | 319 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_319 -c tessedit\_char\_whitelist= |
| Whitelist of special characters doesn't work | 320 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_320 -c tessedit\_char\_whitelist=~!$%^&(+=]? |
| Whitelist of special characters doesn't work | 321 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_321 -c tessedit\_char\_whitelist=abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ25689 |
| Whitelist of special characters doesn't work | 322 | Tesseract images\12\_testType\_sub\_lang\_Arial\_20.png output/test\_322 -c tessedit\_char\_blacklist= -c tessedit\_char\_unblacklist= -c tessedit\_char\_whitelist=~!$%^&(+=]?abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ  -c page\_separator= |
| Page Separator in Swedish doesn't work | 325 | Tesseract images\1\_testType\_file\_format\_Arial\_20.png output/test\_325 -l swe -c page\_separator=Å |
| Page Separator in Spanish doesn't work | 326 | Tesseract images\1\_testType\_file\_format\_Arial\_20.png output/test\_326 -l spa -c page\_separator=ñ |
| Page Separator in Swedish doesn't work | 332 | Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_332 -l swe -c page\_separator=Å |
| Page Separator in Spanish doesn't work | 333 | Tesseract images\229\_testType\_simple\_tests\_Arial\_40.png output/test\_333 -l spa -c page\_separator=ñ |