**קורס עיבוד תמונה פרוייקט סוף**

**מגישים:**

**שם:** דן אברין **ת.ז.** 204358394

**שם:** דניאל קוצ'ין **ת.ז.** 314256827

בפרוייקט זה בחרנו לעבוד על תמונה מקורית שלנו:

****

אותה תמונה בשחור לבן:



1. **Pre-Processing:**

בשלב הpre processing העברנו על התמונה על פי הצורך פילטרים של חידוד או החלקה:

תוצאת פילטר חידוד:



תוצאת פילטר החלקה:



1. **Geometric Operators:**

בשלב זה, הרצנו אופרטורים מורפולוגים שונים על התמונה:

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [1, 0, 0] [0, 1, 0] [0, 0, 1] | transformation\_identity |
| [2, 0, 0] [0, 2, 0] [0, 0, 1] | transformation\_scale\_filter\_2x |
| [0.5, 0, 0] [0, 0.5, 0] [0, 0, 1] | transformation\_scale\_filter\_halfx |
| [1, 1, 0] [0, 1, 0] [0, 0, 1] | transformation\_scale\_horizontal\_shear |
| [1, 0, 0] [1, 1, 0] [0, 0, 1] | transformation\_scale\_vertical\_shear |
| [1, 0, 100] [0, 1, 100] [0, 0, 1 ] | transformation\_translation |
| [0.9, -0.7, 0] [0.7, 0.9, 0] [0, 0, 1] | transformation\_rotation |

**2.1 טרנפורמציית הזהות:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [1, 0, 0] [0, 1, 0] [0, 0, 1] | transformation\_identity |

**תוצאה:**

****

בתמונה זו היה שימוש באופרטור המורפולוגי של זהות, ולכן קיבלנו אותה תמונת מקור שאיתה עבדנו.

**2.2 טרנפורמציית scaling - במקרה זה הגדלה פי 2:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [2, 0, 0] [0, 2, 0] [0, 0, 1] | transformation\_scale\_filter\_2x |

**תוצאה:**

****

בתמונה הנ"ל קיבלנו תמונה כביכול יותר כהה, אך בפועל מדובר בתמונה גדולה 2X מהתמונה המקורית, ועקב מרווח הפיקסלים במתיחה התמונה נראית יותר כהה,

**ניתן לראות זאות בעזרת ZOOM IN לחלק מן התמונה:**



**2.3 טרנפורמציית scaling - במקרה זה הקטנה פי 0.5:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [0.5, 0, 0] [0, 0.5, 0] [0, 0, 1] | transformation\_scale\_filter\_halfx |

**תוצאה:**



קשה להבין, אך אם נעשה ZOOM IN נוכל לראות את ההבדלים,

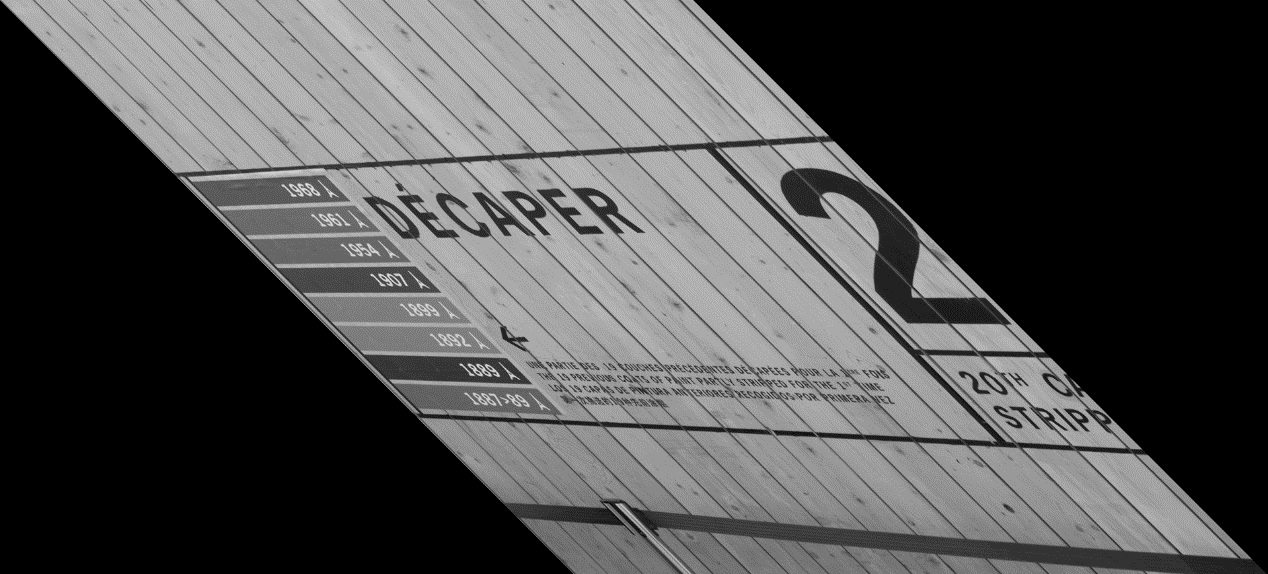
**מימין, התמונה המקורית, משמאל התמונה שעברה הקטנה פי 0.5:**



**2.4 טרנפורמציית Horizontal Shear - מתיחה לרוחב:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [1, 1, 0] [0, 1, 0] [0, 0, 1] | transformation\_scale\_horizontal\_shear |

**התוצאה:**

****

**2.5 טרנפורמציית Vertical Shear - מתיחה לאורך:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [1, 0, 0] [1, 1, 0] [0, 0, 1] | transformation\_scale\_vertical\_shear |

****

**2.6 טרנפורמציית Translation - הזזה – במקרה זה הזזה ימינה ומטה ב100 פיקסלים:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [1, 0, 100] [0, 1, 100] [0, 0, 1 ] | transformation\_translation |

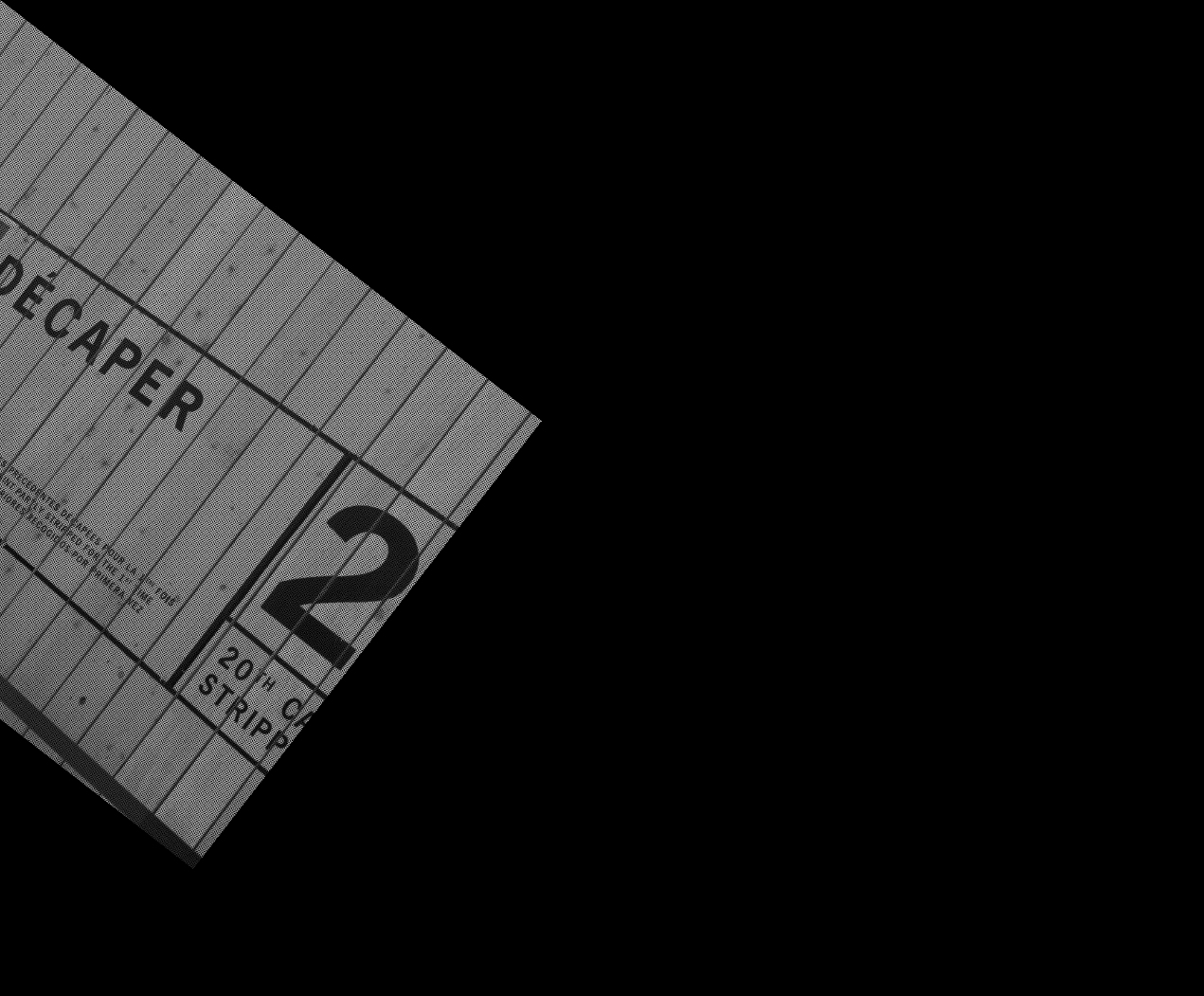
**תוצאה:**



**2.7 טרנפורמציית Rotation – סיבוב – במקרה זה הזזה ימינה ומטה ב100 פיקסלים:**

|  |  |
| --- | --- |
| Transformation | Transformation Name |
| [0.9, -0.7, 0] [0.7, 0.9, 0] [0, 0, 1] | transformation\_rotation |

**תוצאה:**

****

1. **Morpological Operators:**

**3.1 מימוש HIT AND MISS:**

ביצענו תהליך HIT AND MISS כאשר עבדנו על חתיכה מן התמונה שהצגנו למעלה:



* + 1. **מציאת הסיפרה 1:**

יצרנו Structure Element עם הצורה המאונכת הבאה:

[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]  
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

והשתמשנו בStructure Eelement הבא על מנת לבודד אותיות, בתהליך הmiss:

[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]  
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]

תחילה ביצענו תהליך Threshold עם סף 0.7:

def get\_thershold(image, thrshold=0.7):  
 # blur the image to denoise  
 blurred\_image = skimage.filters.gaussian(image, sigma=1.0)  
 binary\_mask = blurred\_image > thrshold  
 binary\_mask = binary\_mask.astype(int)  
 return binary\_mask

וקיבלנו את התוצאה הבאה:

