

הנדסת חלל – מטלה 1

חלק א:

הסבר מילולי- יש לנו 2 תמונות של כוכבים, אחת גדולה המכילה כ-200-100 כוכבים ואחת קטנה המכילה כ-20-10 כוכבים ואנחנו צריכים למצוא התאמה.

האלגוריתם שלנו יפעל על פי כמה שלבים וייעזר באלגוריתם ראנזק שנלמד בהרצאה ובאופן כללי האלגוריתם מנסה למצוא את הזוגות המתאימים ביותר של כוכבים בשתי התמונות על ידי השוואת המאפיינים שלהם כדי לאפשר חישוב מיקום וזווית תנועה של המצלמה בין שתי התמונות:

שלב ראשון – למצוא 2 כוכבים מרכזיים בכל אחת מהתמונות באמצעות אלגוריתם ראנזק.

שלב שני – נשתמש באלגוריתם ראנזק כדי לחשב את מערכת הקואורדינטות המשותפת בין הנקודות המתאימות, האלגוריתם יחשב את זה על ידי בחירת קבוצה של נקודות מתאימות וחישוב ההעברה המתאימה להם.

שלב שלישי – נשתמש במערכת הקואורדינטות המשותפת כדי להעביר את הנקודות מהתמונה הראשונה למערכת הקואורדינטות בתמונה השנייה.

שלב רביעי – נחשב את ההפרש במיקום של כל זוג נקודות מהתמונות ונשתמש בזה לצורך תיקון עיוותים וכדי למצוא את ההתאמה הטובה ביותר בין שתי התמונות.

האלגוריתם מצפה למצוא קבוצות של כוכבים בשתי התמונות שנמצאים בקרבה זה לזה ושמרחקים ומסתיימים באותו זמן ובאותו מרחב. על מנת למצוא התאמה בין שתי תמונות כוכבים יש להתייחס למיקום היחסי של הכוכבים בין שתי התמונות.

חיפשנו קוד המתאים ומצאנו משהו בסגנון הבא בפייתון:

```
import cv2
```

```
big_image = cv2.imread('big_image.jpg')
```

```
zoomed_image = cv2.imread('zoomed_image.jpg')
```

```
gray_big = cv2.cvtColor(big_image, cv2.COLOR_BGR2GRAY)
```

```
gray_zoomed = cv2.cvtColor(zoomed_image, cv2.COLOR_BGR2GRAY)
```

```
detector = cv2.ORB_create()
```

```
kp1, des1 = detector.detectAndCompute(gray_big, None)
```

```
kp2, des2 = detector.detectAndCompute(gray_zoomed, None)
```

```
pts1 = np.float32([kp1[m.queryIdx].pt for m in matches]).reshape(-1, 1, 2)
```

```
pts2 = np.float32([kp2[m.trainIdx].pt for m in matches]).reshape(-1, 1, 2)
```

```
H, _ = cv2.findHomography(pts2, pts1, cv2.RANSAC)
```

```
h, w = big_image.shape[: 2]
```

```
aligned_image = cv2.warpPerspective(zoomed_image, H, (w, h))
```

הסבר OpenCV - cv2 - ספרייה המשמשת למניפולציות על תמונות, נטען את 2 התמונות שלנו

לאחר מכן נוסיף פילטר שהופך את התמונה לשחור לבן - לבן יהיה לנו כוכבים ושחור יהיה לנו הרקע הכללי.

לאחר מכן נזהה את כל הכוכבים על ידי שימוש באלגוריתמים שמזהים שינויים בבהירות של פיקסלים מסוימים לדוגמא ORB.

בשלב הזה יש לנו את כל הכוכבים שלנו ממופים.

לאחר מכן ננסה בברוט פורס (כל האפשרויות הקיימות) למצוא התאמה כלשהי .

בשלב הזה יש לנו מבנה נתונים המכיל את כל ההתאמות, ונשאר רק למצוא את ההתאמה הקרובה ביותר.