|  |  |
| --- | --- |
| **הצעת פרויקט-**  **תש"ף**  **חלק א' - פרטי הפרויקט** | |
| **שם: איתי מזרחי**  **Name: Itay Mizrahi**  **טלפון: 0546445586**  [**itaymizrahi@gmail.com**](mailto:itaymizrahi@gmail.com) **דוא"ל:** | **שם: אפרים סטוקמן**  **Name: Efrayim Sztokman**  **טלפון : 0526695920**  **effiszt@gmail.com דוא"ל:** |
| **שם המנחה : מאעווייה עקש** | |
| **שם הפרויקט : Photo Organizer** | |
| **מקום ביצוע הפרויקט : מכללת הדסה** | |

# תיאור משוער של הפרויקט

* **מבוא –**

לעיתים קורה שאנו מצלמים מספר רב של תמונות השוכנות במחשב ושוכחים מהם, בעת שאנו רוצים לראות את התמונות האלו שוב אנו צריכים לעבור קודם כל מספר מכשולים:

1) כפילויות של תמונות או תמונות זהות – כמו תמונות אשר צולמו ברצף ואנו מעוניינם רק ב"טובות" מביניהם.

2) תמונות אשר סובבנו או ערכנו ולא מחקנו אחת מהתמונות (שאולי אנו לא רוצים).

3) התמונות מפוזרות אקראית בדיסק ללא שום סדר ואנו נדרשים לחפש זמן רב במערכת

הקבצים.

* **סקירת רקע –** כיום קיימים בשוק מספר רב של תוכנות אשר מבטיחות מציאת תמונות כפולות כמו [Duplicate Photos Fixer Pro](https://www.duplicatephotosfixer.com/lp/duplicate-photos-finder/?useparams=1&utm_source=googleadw&utm_campaign=dpf_gads_int_lp5&utm_medium=GADS_DPFW_LP5&utm_adgroup=duplicatephotosfixer&btnclicktrackappend=dpf_gads_int_lp5&pxl=GADS_DPFW_LP5&adwp_campaignid=1062967636&adwp_adgroupid=53223748758&adwp_feeditemid=&adwp_targetid=kwd-380262707237&adwp_loc_interest_ms=&adwp_loc_physical_ms=1007978&adwp_matchtype=e&adwp_network=g&adwp_device=c&adwp_devicemodel=&adwp_gclid=CjwKCAiAh5_uBRA5EiwASW3IanwKMp0Ob77yQOQILJi6KryPkNMh0u0Tesk8dy_4BaXK13Vwu9bi9RoCiVQQAvD_BwE&adwp_ifmobile_value=&adwp_ifnotmobile_value=computer_tablet&adwp_ifsearch_value=google_search&adwp_ifcontent_value=&adwp_creative=305712576172&adwp_keyword=duplicate%20photos%20fixer%20pro&adwp_placement=&adwp_target=&adwp_param1=&adwp_param2=&adwp_random=4127910006721257567&adwp_adposition=1t1&adwp_escapedlpurl=https%3A%2F%2Fwww.duplicatephotosfixer.com%2Flp%2Fduplicate-photos-finder%2F%3Fuseparams%3D1%26utm_source%3Dgoogleadw%26utm_campaign%3Ddpf_gads_int_lp5%26utm_medium%3DGADS_DPFW_LP5%26utm_adgroup%3Dduplicatephotosfixer%26btnclicktrackappend%3Ddpf_gads_int_lp5%26pxl%3DGADS_DPFW_LP5&adwp=1&gclid=CjwKCAiAh5_uBRA5EiwASW3IanwKMp0Ob77yQOQILJi6KryPkNMh0u0Tesk8dy_4BaXK13Vwu9bi9RoCiVQQAvD_BwE), [DupeGuru](https://dupeguru.voltaicideas.net/), [Awesome Duplicate Photo Finder](https://www.portablefreeware.com/index.php?id=1807) ועוד.

Duplicate Photos Fixer: היא הקרובה ביותר לפתרון שאנחנו מציעים, מוצאת תמונות שאינן 100% זהות

כולל תמונות בקנה מידה שונה אך לא מתמודדת עם תמונות מסובבות. ובנוסף דורשת תשלום על מנת לאפשר פיצ'רים כגון מחיקת כל התמונות בלחיצת כפתור אחד.

:Awesome Duplicate Photo Finder כמו Duplicate Photos Fixer ואף מתאימה בין תמונה עם רקע transparent לאותה התמונה עם רקע קיים אך לא מוצאת תמונות בקנה מידה שונה או מסובבות.

DupeGuru: מוצאת תמונות שדומות ב 100% בלבד ומציעה ממשק משתמש פחות נוח.

* **מוטיבציה –**

הקלה על ארגון תמונות המשתמש ופינוי של מקום בדיסק.

* **מטרת הפרויקט –**

התכנה שלנו תסרוק עבור המשתמש את כל התמונות הנמצאות במחשב, תמצא כפילויות של תמונות או תמונות דומות כגון 2 תמונות של אדם שנלקחו אחת אחרי השנייה: באחת העיניים שלו פקוחות ובשניה הן סגורות. כל זה על מנת לאפשר למשתמש למחוק את התמונות אשר הוא לא מעוניין בהם.

* [ אם נספיק: התכנה גם תקטרג את התמונות באופן שיהיה יותר נוח למשתמש למצוא תמונה שהוא רוצה, דוגמא אפשרית : התכנה תשים בקבוצה את כל התמונות שצולמו בחוף ים]
* **מפרט דרישות –**

1) התכנה תפעל על מערכת הפעלה Windows.

2) דרוש מעבר מהיר על כל מערכת הקבצים כדי שהמשתמש לא ימתין זמן רב, ולכן יש דגש רב על יעילות.

מבחן היעילות תהיה בדיקה כי זמן לביצוע התכנה לא יעלה על חצי שעה עבור 2000 תמונות (בערך 70 תמונות בדקה ), פירוט הדרישה בכיצד תבחן הצלחה של הפרויקט (סעיף 3) .

3)התכנה לא יכולה לצרוך הרבה זכרון כי המשתמש ירצה להשתמש בתכנה גם על מערכת אשר אין בה הרבה זכרון פנוי, נרצה שהזכרון שהתכנה מקצה לטובת ריצתה לא תעלה על 10% של הזכרון (עת אין מספיק זיכרון פנוי יודיע למשתמש כדי שיפנה מקום).

4)יש צורך בבטחון בזיהוי- נרצה לאפשר זיהוי שגוי, כלומר על סרק, לא יותר מ5% מהזמן, אחרת תרד רמת האמינות בתכנה.

5) יש צורך בכך שהמערכת לא תפספס זיהויים של תמונות זהות יותר מ 15% מהזמן.

6) ניתן יהיה לבדוק זיהוי תמונות ב3 רמות דיוק (העולות בחומרתן): א) רק תמונות זהות לחלוטין ב)גם תמונה עם סיבוב/ הגדה / הקטנה ג) גם עם שינוי בתמונה עצמה (כגון – מבט שונה של אדם בתמונה, תזוזה של המצלם המשנה את זווית הצילום, אדם בתמונה עצם עיניים וכו') [ ד) אם נספיק: על פי סוג התמונה ע"י זיהוי האוביקטים בתמונה ותיוגם]

7) דרוש ממשק נוח למשתמש אשר תהיה קלה להבנה ולתפעול.

8) יכולת שליטה של המשתמש על הקבצים אשר המערכת מוצאת הכוללת אפשרות למחיקה ו/או העבר התמונה לתיקיות אחרות.

* **תכנון כללי –**

תכנון ראשוני לפני פיתוח:

1. קריאה ולמידה.
2. על סמך שלב 1 קבלת החלטה סופית לגבי השימוש בלמידת מכונה.
3. באם הוחלט על שימוש בלמידת מכונה – הכנת מאגר תמונות מגוון כדי להזין את המכונה.
4. בדיקת זמני ריצה של אלגוריתמים לזיהוי תמונה קיימים כדי לחשב זמן ריצה משוער של התוכנה שלנו ובמידת הצורך חשיבה על פתרון שאינו הטריוויאלי כדי להמנע מהשוואה של כל תמונה עם כל תמונה אחרת.
5. החלטה על שפות התכנות / ספריות שנעשה בהם שימוש בפרויקט.
6. חלוקת משימות לקראת הפיתוח.

פיתוח:

1( כתיבת אלגוריתם או אלגוריתמי זיהוי, בהתאם להחלטות שקבלנו בשלבים הקודמים.

2) בדיקות לאלגוריתם הזיהוי ושיפורים כדי להגיע לרמת הדיוק הרצוייה.

3) סריקה יעילה של מערכת הקבצים ומציאת כל התמונות.

4) שילוב השלבים הקודמים וישום הפתרון משלב 4 (תכנון ראשוני) לבניית מבנה הנתונים שיכיל את כל התמונות הדומות שנמצאו בסריקה.

5) בדיקות שאכן מבנה הנתונים מכיל תמונות ברמת הדיוק הרצויה. ושפורים במידת הצורך.

5) פיתוח ממשק המשתמש.

6) בדיקות לממשק המשתמש ושיפורים במידת הצורך.

* **תחומים אשר הפרויקט משתייך**

1) עיבוד תמונה- הפרוייקט עוסק בסוגים רבים של תמונות, יש צורך בשיטות ניהול וזיהוי תמונות.

2)למידת מכונה- הכלי המרכזי זיהוי תמונות דומות יהיה מערכת לומדת אשר תעשה הבחנות אלו.

3) מבני נתונים - עלינו לשמור מידע על התמונות בצורה שתיעל את מהירות ההשוואות.

* **מורכבות הפרוייקט –**

1)סריקת מערכת הקבצים : בזמן סביר, מציאת כל התמונות השוכנות באחסון ללא טלות בסוג התמונות (JPEG, GIF וכו').

2) שליטה בזיכרון - אנו צריכים לשמור כמות מידע להשוואת תמונות מינימלי אך מספיק על מנת לבצע הבדיקות.

3)עיבוד תמונה –דרוש אלגוריתם קליטת מידע מתמונה לצורך השוואה מדוייקת ויעילה עם תמונות אחרות.

4)ממשק נוח למשתמש- יש צורך בבניית ממשק נוח וידידותי למשתמשים בתכנה.

5) דיוק הבדיקה לעומת מהירות הבדיקה – הזמן לבדיקה בודדת חייבת להיות מהירה כדי שזמן ריצת התוכנית לא תהיה גדולה מדי אך דרוש בדיקה יסודית על מנת

6) יש צורך בהורדת כמות הבדיקות ממספר הטריוויאלי O(N2) אשר על קלט בגודל 2000 תדרוש 4 מיליון השוואות. יש חשיבות להתקרב ל - . O(NlogN)

* **כלים בהם יעשה שימוש -** Phash , machine learning .
* **כלים בהם אולי יעשה שימוש-** Keypoint matching using randomized trees, SIFT .
* **כיצד תבחן הצלחה של הפרויקט.**

1)היבט הזיהוי- ייבדק על ידי 1000 תמונות אשר מתוכם חלק זהים וחלק שונים מעט. נקבע אחוז הצלחה רצוי להיות 85% לזיהוי מוצלח מבין כל התמונות הכפולות 75% מבין התמונות השונות מעט . ופחות מ5% זיהוי תמונות ככפולות שאינם כפולות (false-positive ).

2)נוחות המשתמש – ניתן ל20 משתמשים להשתמש בתוכנה ולענות על סקר שביעות רצון בה נשאל לגבי מהירות התכנה ונוחות הממשק הגרפי.

3) מבחן היעילות תהיה בדיקה כי זמן לביצוע התכנה לא יעלה על חצי שעה עבור 2000 תמונות (בערך 70 תמונות בדקה ).

דרישה זו תיבדק על מחשב עם המפרט הבא:

System: Intel® Core™ i7-8700 CPU @ 3.20GHz

16GB RAM

OS: Windows 10 64-bit

Disk space: 250GB

**חלק ב' – תכנון השלבים ולוח זמנים משוער של הפרוייקט**