

ReadMe

Περιγραφή και σχολιασμός κώδικα και εν γένει παραδοτέων αρχείων

Ρέππα Σμαράγδα – 1115201600143

Ακαδημαϊκό έτος: 2020-2021

Σχολιασμός Παραδοτέων

Στα παραδοτέα αρχεία βρίσκονται 4 αρχεία κώδικα Python και 5 φακέλων, απαραίτητων για τη λειτουργία του προγράμματος.

- `functions.py`: Αρχείο κώδικα, λειτουργεί σαν βιβλιοθήκη συναρτήσεων
- `img_proc.py`: Αρχείο κώδικα, λειτουργεί σαν βιβλιοθήκη συναρτήσεων
- `sift.py`: Εκτελέσιμο αρχείο κώδικα
- `compare.py`: Εκτελέσιμο αρχείο κώδικα
- `figures`: Φάκελος στον οποίο αποθηκεύονται τα διαγράμματα που εκτυπώνονται στο τέλος της εκτέλεσης του `sift.py`, ανάλογα με τη βούληση του χρήστη
- `img_keypoints`: Φάκελος στον οποίο αποθηκεύονται οι φωτογραφίες μαζί με τα keypoints τους. Αποθηκεύονται και για τα δύο εκτελέσιμα, οι φωτογραφίες της τελευταίας εκτέλεσης ή επανάληψης, όχι όλων.
- `matches`: Φάκελος στον οποίο αποθηκεύονται οι ζεύγη φωτογραφιών με τα matches τους. Αποθηκεύονται και για τα δύο εκτελέσιμα, οι φωτογραφίες της τελευταίας εκτέλεσης ή επανάληψης, όχι όλων.
- `training_images`: Περιέχονται 4 φωτογραφίες από τις οποίες ο χρήστης, σε κάθε εκτελέσιμο, θα πρέπει να επιλέξει μια από αυτές για να τρέξει το πρόγραμμα.
- `transformed_images`: Αποθηκεύονται από το πρόγραμμα που εκτελείται εκείνη τη στιγμή. Είναι η φωτογραφία που επιλέχθηκε με το φίλτρο που της εφαρμόστηκε.

Ο σκοπός που όλες αυτές οι φωτογραφίες αποθηκεύονται είναι επειδή αν εκτυπωθούν, ο χρήστης θα τις δει για λίγα μόνο δευτερόλεπτα. Γι' αυτό αν αποθηκευθούν δίνεται η δυνατότητα στον χρήστη να δει κάποια αποτελέσματα του, έχοντας χρόνο και χωρίς βιασύνη. Οι φάκελοι και τα αρχεία κώδικα πρέπει να βρίσκονται στο ίδιο directory.

Σχολιασμός Αρχείων Κώδικα

Θα σχολιαστεί τι χρησιμοποιεί το πρόγραμμα, θα γίνει περιγραφή κάποιων σημαντικών διαδικασιών, ενώ υπάρχει και σχολιασμός σ εμορφή σχολίων στα αρχεία κώδικα για περισσότερες πληροφορίες.

- `img_proc.py`

Δεν είναι προς εκτέλεση, λειτουργεί ως βιβλιοθήκη. Περιέχει όλες τις συναρτήσεις που εφαρμόζουν φίλτρο πάνω στην αρχική εικόνα και μια ακόμη συνάρτηση.

Η `process_img()` παίρνει την αρχική εικόνα και της εφαρμόζει το κάθε φίλτρο μια φορά και ύστερα 2 φορές. Δηλαδή δημιουργεί ένα query set που περιέχει την αρχική εικόνα, μια φωτογραφία που έχει περάσει μια φορά από gaussian blur, 2 φορές από gaussian blur, μια φορά από median blur, δύο φορές από median blur, κ.ο.κ. Επιστρέφει μια λίστα από αυτές τις φωτογραφίες και τις αποθηκεύει όλες στον φάκελο `transformed images`. Οι αριθμοί που δίνονται στις εικόνες, γράφονται στα σχόλια της συνάρτησης. Με αυτούς του αριθμούς

αποθηκεύονται, αποκαλούνται κατά τη διάρκεια της εκτέλεσης του κώδικα. Δίνονται και παρακάτω στην περιγραφή του User Interface.

- `functions.py`

Δεν είναι προς εκτέλεση, λειτουργεί ως βιβλιοθήκη. Περιέχει συναρτήσεις απαραίτητες και για τα 2 εκτελέσιμα.

Η `load_images()` χρησιμεύει στο να ανοίγει τον φάκελο `training_images` και να διαβάζει τις 4 εικόνες. Επιστρέφει λίστα με αυτές τις εικόνες και ετικέτα για την κάθε μια.

Η `apply_sift_algo()` για κάθε φωτογραφία του `query_set` βρίσκει τα `keypoints` και τους αντίστοιχους `descriptors` με τη βοήθεια της `detectAndCompute()` της βιβλιοθήκης OpenCV για το μοντέλο που περάστηκε σαν παράμετρος. Το πλήθος των `keypoints` που βρέθηκε σε κάθε φωτογραφία εκτυπώνεται. Η `drawKeypoints()` του OpenCV επιστρέφει την εκάστοτε εικόνα με τα `keypoints` της ζωγραφισμένα. Αυτές αποθηκεύονται στον φάκελο `img_keypoints` αν το επιθυμεί ο χρήστης. Τα `keypoints` και `descriptors` που επιστρέφονται από την `detectAndCompute()` αποθηκεύονται σε μια λίστα, η οποία θα περαστεί στην `matcher_function`.

Η `matcher_function()` χρειάζεται τα `keypoints` και τα `descriptors` της κάθε εικόνας. Χρησιμοποιείται ο Brute Force Matcher του OpenCV, ο οποίος καλεί συνάρτηση πλησιέστερων γειτόνων για το ταιρίασμα εικόνων. Τα μόνα ταιριάσματα που θα βρεθούν είναι αυτά της αρχικής εικόνας με όλες τις επεξεργασμένες. Δεν θα διερευνηθούν τα ταιριάσματα όλων των φωτογραφιών με όλες. Με το ratio test του D. Lowe θα κρατηθούν μόνο τα καλά `matches`. Οι φωτογραφίες των ζευγών και των κοινών τους σημείων αποθηκεύονται, αν το θέλει ο χρήστης, στον φάκελο `matches` και ο αριθμός των `matches` που βρέθηκαν για κάθε ζεύγος εκτυπώνεται μαζί με το συνολικό success rate του μοντέλου.

Η `save_graphs()` αποθηκεύει με την βοήθεια της βιβλιοθήκης `matplotlib` σχεδιάζει τα γραφήματα του πλήθους χαρακτηριστικών και του ποσοστού επιτυχίας που ορίστηκε, συναρτήσει της μεταβλητής με την οποία επέλεξε να πειραματιστεί ο χρήστης.

- `sift.py`

Βρίσκεται η `main` συνάρτηση του εκτελέσιμου. Οι συναρτήσεις της βρίσκονται στα `functions.py`, `img_proc.py`.

Διαβάζει από τον αντίστοιχο φάκελο τις αρχικές 4 εικόνες και ρωτάει τον χρήστη ποιά από τις φωτογραφίες θέλει να επιλέξει για να δημιουργηθεί το `query set` για το μοντέλο. Το δημιουργεί μέσω του `process_img()`.

Στη συνέχεια, δίνει στον χρήστη τη δυνατότητα να επιλέξει αν θέλει να τρέξει πειράματα ή όχι. Αν όχι, οι φωτογραφίες θα περαστούν σε ένα μοντέλο με default παραμέτρους και θα κληθεί η συνάρτηση `apply_sift_algo()`. Την διαδικασία αυτής της συνάρτησης την αναλύσαμε πιο πάνω. Αν επιλέξει κάθε φορά που θα του ζητηθεί να αποθηκευθούν οι αντίστοιχες φωτογραφίες, κοιτώντας στους αντίστοιχους φακέλους για φωτογραφίες με `keypoints` και `matches`, θα δει τα αποτελέσματα του αλγορίθμου. Αν ναι, τότε το πρόγραμμα μπαίνει σε μια δομή επανάληψης, η οποία επιτρέπει στον χρήστη να τρέξει πολλαπλά πειράματα. Σε κάθε επανάληψη ζητείται από τον χρήστη να δώσει τις τιμές των παραμέτρων που θα ήθελε να είναι στο μοντέλο και καλείται η `apply_sift_algo()`.

Στο τέλος κάθε επανάληψης δίνεται στον χρήστη η ευκαιρία να τερματίσει το πρόγραμμα. Αν επιλέξει όχι, του ζητούνται εκ νέου παράμετροι για το μοντέλο. Αν ναι, το πρόγραμμα ρωτάει τον χρήστη αν θέλει διαγράμματα για τα πειράματα του. Αν επιλέξει όχι το πρόγραμμα τερματίζει. Αν επιλέξει ναι, του ζητείται η παράμετρος για την οποία θέλει να γίνουν διαγράμματα, καλείται η `save_graphs()` και το πρόγραμμα τερματίζει.

- `compare.py`

Βρίσκεται η main συνάρτηση του εκτελέσιμου. Οι συναρτήσεις της βρίσκονται στα functions.py, img_proc.py.

Διαβάζει από τον αντίστοιχο φάκελο τις αρχικές 4 εικόνες και ρωτάει τον χρήστη ποιά από τις φωτογραφίες θέλει να επιλέξει για να δημιουργηθεί το query set για το μοντέλο. Το δημιουργεί μέσω του process_img().

Η σύγκριση αποτελεσμάτων πρόκειται να γίνει ανάμεσα στα αποτελέσματα ενός μοντέλου με τις default τιμές των παραμέτρων και σε ένα μοντέλο που θα έχει παραμέτρους αυτές που θα δοθούν απο τον χρήστη. Στο πρώτο μοντέλο θα δοθεί για επεξεργασία η αρχική εικόνα από τις 4 στο training_images, ενώ στο δεύτερο θα δοθεί κάποια από το υπόλοιπο query set, κάποια που έχει περάσει απο φίλτρο.

Αφου δημιουργηθούν τα δυο μοντέλα και ο χρήστης επιλέξει φωτογραφία, τότε με την detectAndCompute() θα υπολογιστούν τα keypoints και οι descriptors για την κάθε φωτογραφία και με την drawKeypoints() θα εκτυπωθούν τα αποτελέσματα στις φωτογραφίες αλλά και θα αποθηκευθούν στον φάκελο img_keypoints.

Ύστερα, θα βρεθούν τα matches των 2 εικόνων τα οποία επίσης θα εκτυπωθούν αλλά και θα αποθηκευθούν στον φάκελο matches.

System Requirements

Python 3.6
Pip 21.1.2
Numpy 1.19.5
Matplotlib 3.3.4
CV2 4.5.2

Περιγραφή του User Interface

- Εκτελώντας python sift.py

1. “Which photo would you like to choose for the experiment?”

Το πρόγραμμα ξεκινάει φορτώνοντας τις φωτογραφίες από τον φάκελο training images. Ο χρήστης καλείται να επιλέξει ποιά απο τις 4 θέλει για να τρέξει το μοντέλο.

```
Which photo would you like to choose for the experiment? cat
```

Στη συγκεκριμένη φωτογραφία, σαν απάντηση στην ερώτηση δίνεται η ‘cat’. Σε αυτή την ερώτηση οι πιθανές απαντήσεις είναι οι εξης: ‘cat’ για τη φωτογραφία με τη γάτα, ‘dog’ για αυτή με τον σκύλο, ‘tree’ για το δέντρο και ‘wheel’ για τη ρόδα. Αν η απάντηση δεν είναι αποδεκτή, δηλαδή όχι κάποια από τις παραπάνω, δίνεται στον χρήστη μια ακόμα ευκαιρία να δώσει τη σωστή συμβολοσειρά.

```
Which photo would you like to choose for the experiment? unacceptable  
This photo doesn't exist. Try again.  
Which photo would you like to choose for the experiment?
```

2. Αυτόματη αποθήκευση query set εικόνων

Όταν επιλεγεί εικόνα, στην `process_img()` περνάει από φίλτρα. Οι εικόνες που προκύπτουν αποτελούν το query set που θα περάσει από το μοντέλο SIFT και αποθηκεύονται αυτόματα στον φάκελο `transformed_images`. Τα ονόματα αρχείων είναι αριθμοί, οι οποίοι αντιστοιχούν στην εκάστοτε επεξεργασία της εικόνας. Δίνονται οι αντιστοιχίες:

1. Filtered once
2. Filtered twice
3. Blurred once
4. Blurred twice
5. Gaussian blurred once
6. Gaussian blurred twice
7. Median blurred once
8. Median blurred twice
9. Bilateral blurred once
10. Bilateral blurred twice

3. “Do you want to experiment with parameters or not? (yes or no)”

Αμέσως μετά, ο χρήστης ρωτάται αν θέλει να πειραματιστεί με τις παραμέτρους.

```
Do you want to experiment with parameters or not? (yes or no) no
```

Οι αποδεκτές απαντήσεις είναι ‘yes’, ‘no’. Στη συγκεκριμένη φωτογραφία, έχει απαντηθεί ‘no’ ενδεικτικά. Αν η απάντηση δεν είναι αποδεκτή, το πρόγραμμα τερματίζει επιτόπου.

```
Do you want to experiment with parameters or not? (yes or no) unacceptable
Not acceptable answer. Exiting...

Finished! Exiting...
```

- (a) Σε περίπτωση που στην ερώτηση “Do you want to experiment with parameters or not?” ο χρήστης απαντήσει αρνητικά, τότε το πρόγραμμα θα τρέξει μια φορά με το μοντέλο με default παραμέτρους.
- (b) Αν απαντήσει θετικά, τότε το πρόγραμμα μπαίνει σε δομή επανάληψης και καλείται να δίνει τιμές παραμέτρων μοντέλου σε κάθε επανάληψη. Προτείνεται, αν θέλετε να πειραματιστείτε με τη συμπεριφορά του μοντέλου, να μεταβάλλετε την ίδια παράμετρο σε κάθε επανάληψη, κρατώντας τις άλλες σταθερές. Επίσης, οι τιμές που πρόκειται να δώσετε, να τις δώσετε με αύξουσα σειρά.

4. Παράμετροι μοντέλου

```
Give number of features: 0
Give number of octave layers: 3
Give contrast threshold: 0.04
Give edge threshold: 10
Give Gaussian sigma: 1.6
```

Κάθε πρόταση εμφανίζεται ξεχωριστά, μόλις απαντηθεί η πρώτη, εμφανίζεται η επόμενη κ.ο.κ. Με τη σειρά που βλέπουμε μας ζητείται να δώσουμε μέγιστο πλήθος χαρακτηριστικών, αριθμό στρωμάτων ανα οκτάβα, contrast threshold, edge threshold, gaussian sigma. Ενδεικτικά, στη φωτογραφία έχουν δοθεί οι default τιμές του μοντέλου. Τα χαρακτηριστικά πρέπει να δοθούν σαν ακέραιοι, όπως και τα στρώματα ανα οκτάβα. Όλα τα υπόλοιπα πρέπει να δοθούν σαν δεκαδικοί.

5. “Do you want pictures with keypoints to be saved? (yes or no)”

Το πρόγραμμα ζητάει από τον χρήστη να του πει αν θέλει να αποθηκεύσει τις φωτογραφίες με σχεδιασμένα τα keypoints. Οι αποδεκτές απαντήσεις είναι ‘yes’, ‘no’.

```
Do you want pictures with keypoints to be saved? (yes or no) yes
```

Αν δοθεί κάποια άλλη απάντηση, το πρόγραμμα θα συνεχίσει αλλά δεν θα αποθηκεύσει τις φωτογραφίες. Επίσης, κάθε φορά που γίνεται save, οι ήδη υπάρχουσες φωτογραφίες του φακέλου από προηγούμενη επανάληψη ή εκτέλεση του κώδικα, χάνονται και αντικαθίστανται από τις νέες.

Οι φωτογραφίες θα αποθηκευθούν με όνομα κάποιον αριθμό. Ο αριθμός αυτός αντιστοιχεί σε κάποια εικόνα του transformed_images και θα αποθηκευθεί στα img_keypoints. Δίνονται οι αντιστοιχίες:

Initial. Τα keypoints στην αρχική εικόνα

1. Τα keypoints στην εικόνα 1 του transformed_images
2. Τα keypoints στην εικόνα 2 του transformed_images
3. Τα keypoints στην εικόνα 3 του transformed_images
4. Τα keypoints στην εικόνα 4 του transformed_images
5. Τα keypoints στην εικόνα 5 του transformed_images
6. Τα keypoints στην εικόνα 6 του transformed_images
7. Τα keypoints στην εικόνα 7 του transformed_images
8. Τα keypoints στην εικόνα 8 του transformed_images
9. Τα keypoints στην εικόνα 9 του transformed_images
10. Τα keypoints στην εικόνα 10 του transformed_images

6. “Do you want to save pictures of matches? (yes or no)”

Το πρόγραμμα ζητάει από τον χρήστη να του πει αν θέλει να αποθηκεύσει στον φάκελο τις φωτογραφίες των ζευγαριών με σχεδιασμένα τα matches που εντοπίστηκαν. Οι αποδεκτές απαντήσεις είναι ‘yes’, ‘no’.

Do you want to save pictures of matches? (yes or no) yes

Αν δοθεί κάποια άλλη απάντηση, το πρόγραμμα θα συνεχίσει αλλά δεν θα αποθηκεύσει τις φωτογραφίες. Επίσης, κάθε φορά που γίνεται save, οι ήδη υπάρχουσες φωτογραφίες του φακέλου από προηγούμενη επανάληψη ή εκτέλεση του κώδικα, χάνονται και αντικαθίστανται από τις νέες.

Οι φωτογραφίες θα αποθηκευθούν με όνομα κάποιον αριθμό. Ο αριθμός αυτός αντιστοιχεί σε κάποια εικόνα του transformed_images και θα αποθηκευθεί στα matches. Δίνονται οι αντιστοιχίες:

1. Τα matches μεταξύ αρχικής και εικόνας 1 του transformed_images
2. Τα matches μεταξύ αρχικής και εικόνας 2 του transformed_images
3. Τα matches μεταξύ αρχικής και εικόνας 3 του transformed_images
4. Τα matches μεταξύ αρχικής και εικόνας 4 του transformed_images
5. Τα matches μεταξύ αρχικής και εικόνας 5 του transformed_images
6. Τα matches μεταξύ αρχικής και εικόνας 6 του transformed_images
7. Τα matches μεταξύ αρχικής και εικόνας 7 του transformed_images
8. Τα matches μεταξύ αρχικής και εικόνας 8 του transformed_images
9. Τα matches μεταξύ αρχικής και εικόνας 9 του transformed_images
10. Τα matches μεταξύ αρχικής και εικόνας 10 του transformed_images

7. “Would you like to repeat the experiment? (yes or no)”

Το πρόγραμμα ζητάει από τον χρήστη να του πει αν θέλει να επαναλάβει τη διαδικασία ώστε να συνεχίσει να πειραματίζεται. Οι αποδεκτές απαντήσεις είναι ‘yes’, ‘no’. Αν δοθεί μη αποδεκτή απάντηση το πρόγραμμα τερματίζει επιτόπου.

Would you like to repeat the experiment? (yes or no) yes

(a) Αν απαντηθεί ναι, ζητούνται νέες παράμετροι από το χρήστη και επαναλαμβάνεται το πείραμα για άλλη μια φορά.

(b) Αν απαντηθεί όχι, τότε ο χρήστης θα ερωτηθεί για το αν θέλει διαγράμματα.

8. “Would you like to have diagrams of your experiments? (yes or no)”

Οι αποδεκτές απαντήσεις είναι ‘yes’, ‘no’. Αν δοθεί μη αποδεκτή απάντηση το πρόγραμμα δεν επιστρέφει διαγράμματα.

Would you like to have diagrams of your experiments? (yes or no) yes

(a) Αν απαντηθεί όχι, το πρόγραμμα τερματίζει.

(b) Αν απαντηθεί ναι, γίνεται μια ακόμα ερώτηση στον χρήστη. Πρέπει να δώσει την παράμετρο με την οποία έκανε πειράματα ώστε τα διαγράμματα να γίνουν συναρτήσει αυτής.

Give the parameter you have been experimenting with: 1

Η απάντηση δίνεται με ακεραίους από 1 έως 5, με τον κάθε ένα να αντιστοιχεί σε μια παράμετρο.

- Εκτελώντας `python compare.py`

Τα βήματα 1, 2 για το `sift.py` ισχύουν και εδώ.

1. Παράμετροι δεύτερου μοντέλου

Το βήμα 4 ισχύει μόνο που οι παράμετροι που θα δοθούν εδώ θα προορίζονται για το δεύτερο απο τα δύο μοντέλα που θα φτιαχτούν. Όπως έχουμε πει η σύγκριση θα γίνει ανάμεσα στα αποτελέσματα ενός μοντέλου με τις default τιμές των παραμέτρων και σε ένα μοντέλο που θα έχει παραμέτρους αυτές που θα δοθούν απο τον χρήστη.

2. “Select the number of the image you want to compare with the initial:”

Έτσι επιλέγεται η φωτογραφία που θα περάσει από το δεύτερο μοντέλο με τις παραμέτρους του χρήστη και θα συγκριθεί με την αρχική φωτογραφία που θα περάσει απο μοντέλο με default τιμές παραμέτρων.

Αποδεκτές απαντήσεις είναι ακέραιος από το 1 έως το 10. Οι αντιστοιχήσεις εικόνων- αριθμών δίνονται στο βήμα 2 για το `sift.py`.

```
Select the number of the image you want to compare with the initial: 7
```

3. Υποχρεωτική αποθήκευση εικόνων με keypoints και matches

Οι δύο φωτογραφίες με τα keypoints τους θα αποθηκευθούν στον φάκελο `img_keypoints` ονόματα `comp_photo1.jpg`, `comp_photo2.jpg`. Η πρώτη είναι η αρχική φωτογραφία με τα keypoints, η δεύτερη είναι η φωτογραφία που επιλέχθηκε απο τον χρήστη, επίσης με τα keypoints.

Η εικόνα με τις 2 φωτογραφίες με τα matches που βρέθηκαν μεταξύ τους θα αποθηκευθεί στον φάκελο `matches` με το όνομα `comp_matches.jpg`.

Και για τις δυο περιπτώσεις ισχύει ότι κάθε εκτέλεση του αλγορίθμου, αντικαθιστά τις προηγούμενες εικόνες που είχαν αποθηκευθεί στον εκάστοτε φάκελο με νέες.