

Γραφική με Υπολογιστές 2018

Εργασία #2: Μετασχηματισμοί και Προβολές

Ζητούμενα

A. Έστω \mathbf{u} στήλη 3×1 με τις συντεταγμένες ενός μοναδιαίου διανύσματος ως προς κάποιο σύστημα συντεταγμένων και θ γωνία σε rad. Να υλοποιήσετε τη συνάρτηση

$$R = \text{rotmat}(\theta, \mathbf{u})$$

που υπολογίζει τον πίνακα περιστροφής κατά θ περί άξονα που διέρχεται από την αρχή του συστήματος συντεταγμένων και είναι παράλληλος προς το \mathbf{u} .

B. Έστω $c_p \in \mathbb{R}^3$ η 3×1 στήλη με τις συντεταγμένες ενός σημείου p ως προς ένα σύστημα συντεταγμένων. Στο ίδιο σύστημα συντεταγμένων, έστω: R ένας 3×3 πίνακας περιστροφής και $c_t \in \mathbb{R}^3$ στήλη 3×1 με τις συντεταγμένες ενός διανύσματος μετατόπισης t . Να υλοποιήσετε τη συνάρτηση:

$$c_q = \text{affinetrans}(c_p, R, c_t)$$

που υλοποιεί σημειακό μετασχηματισμό affine εφαρμόζοντας την περιστροφή του p κατά R και έπειτα την μετατόπιση του κατά t . Φροντίστε η `pointtrans` να δουλεύει σωστά και στην περίπτωση που τα c_p, c_q είναι $3 \times n$ πίνακες με τις συντεταγμένες n σημείων.

Γ. Έστω $c_p \in \mathbb{R}^3$ η 3×1 στήλη με τις συντεταγμένες ενός σημείου p ως προς σύστημα συντεταγμένων με αρχή o και άξονες (b'_1, b'_2, b'_3) . Έστω $d_p \in \mathbb{R}^3$ οι συντεταγμένες του ίδιου σημείου ως προς σύστημα συντεταγμένων με αρχή $o \oplus \mathbf{v}_0$ και άξονες $(b_1, b_2, b_3) = (\mathcal{L}(b'_1), \mathcal{L}(b'_2), \mathcal{L}(b'_3))$, όπου \mathcal{L} μετασχηματισμός περιστροφής όπως περιγράφηκε παραπάνω. Αν $c_o \in \mathbb{R}^3$ η 3×1 στήλη με τις συντεταγμένες του διανύσματος \mathbf{v}_0 ως προς το αρχικό σύστημα αξόνων, να υλοποιήσετε τη συνάρτηση:

$$d_p = \text{systemtrans}(c_p, b_1, b_2, b_3, c_o)$$

Φροντίστε η `systemtrans` να δουλεύει σωστά και στην περίπτωση που τα c_p, d_p είναι $3 \times n$ πίνακες με τις συντεταγμένες n σημείων.

Δ. Έστω $c_p \in \mathbb{R}^3$ η 3×1 στήλη με τις συντεταγμένες ενός σημείου ως προς το WCS, $\{o, \mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0\}$. Έστω ότι μία προοπτική κάμερα έχει κέντρο $c = o \oplus \mathbf{v}_c$ και μοναδιαία διανύσματα $\{\mathbf{x}_c, \mathbf{y}_c, \mathbf{z}_c\}$. Έστω

ότι τα $\mathbf{v}_c, \mathbf{x}_c, \mathbf{y}_c, \mathbf{z}_c$ έχουν συντεταγμένες c_v, c_x, c_y, c_z αντίστοιχα ως προς το WCS και w είναι η απόσταση του πετάσματος από το κέντρο (μετρημένη στις μονάδες που χρησιμοποιεί το σύστημα συντεταγμένων της κάμερας). Να υλοποιήσετε τη συνάρτηση:

$$[P, D] = \text{projectCamera}(w, c_v, c_x, c_y, p)$$

που παράγει τις προοπτικές προβολές των τρισδιάστατων σημείων και τις επιστρέφει στον πίνακα P διάστασης $2 \times n$. Ακόμη, υπολογίζει το βάθος κάθε σημείου πριν την προβολή του στις 2 διαστάσεις και το επιστρέφει στον πίνακα D διάστασης $n \times 1$. Φροντίστε η `projectCamera` να δουλεύει σωστά και στην περίπτωση που το p είναι $3 \times n$ πίνακας με τις συντεταγμένες n σημείων.

E. Να υλοποιήσετε τη συνάρτηση

$$[P, D] = \text{projectKu}(w, c_v, c_K, c_u, p)$$

που παράγει τις προοπτικές προβολές και το βάθος των τρισδιάστατων σημείων του p όπως και η προηγούμενη αλλά δέχεται ως είσοδο τις συντεταγμένες c_K και c_u (σε μη ομογενή μορφή) του σημείου στόχου K και του μονοδιαίου u vector \mathbf{u} αντίστοιχα. Το c_v περιέχει όπως και πριν τις συντεταγμένες του κέντρου της κάμερας ως προς το WCS.

ΣΤ. Να υλοποιήσετε τη συνάρτηση

$$P_{rast} = \text{rasterize}(P, M, N, H, W)$$

που απεικονίζει συντεταγμένες σημείων από το σύστημα μιας κάμερας με πέτασμα διάστασης $H \times W$ (σε ίντσες), σε ακέραιες θέσεις (pixels) της εικόνας, διάστασης $M \times N$, που παράγεται σαν έξοδος από την κάμερα κατά τη φωτογράφιση.

Z. Να υλοποιήσετε τη συνάρτηση

$$P_{2d} = \text{photographObject}(p, M, N, H, W, w, c_v, c_K, c_u)$$

που να χρησιμοποιεί κατάλληλα τις παραπάνω συναρτήσεις για να υλοποιήσει όλο το pipeline της φωτογράφισης ενός αντικειμένου.

Παραδοτέα

1. Τα προγράμματα σε μορφή **σχολιασμένου** πηγαίου κώδικα με σχόλια γραμμένα στα **αγγλικά ή greeklish**.
2. Script επίδειξης με όνομα `demo.m`. Το script αυτό θα πρέπει να καλείται χωρίς εξωτερικά ορίσματα, να διαβάζει το αντικείμενο από το αρχείο `hw2.mat` που σας δίνεται και να εκτελεί ένα προκαθορισμένο σύνολο μετασχηματισμών που περιγράφεται παρακάτω. Ως είσοδο χρησιμοποιείτε τον πίνακα V , διάστασης $K \times 3$, που περιέχει τις τρισδιάστατες συντεταγμένες των K κορυφών των τριγώνων που αποτελούν το αντικείμενο. Δοθέντων των σημείων του πίνακα V , το script σας θα πρέπει να εκτελεί σειριακά τα παρακάτω βήματα:

- (i) Τα μετατοπίζει κατά t_1 .

- (ii) Τα περιστρέφει κατά γωνία ϕ rad περί άξονα που διέρχεται από σημείο K και έχει κατεύθυνση παράλληλη προς διάνυσμα g .
- (iii) Τα μετατοπίζει κατά t_2 .

Κάθε βήμα θα δέχεται ως είσοδο την έξοδο του προηγούμενου. Μετά από κάθε βήμα θα πρέπει να φωτογραφίσετε το αντικείμενο, καλώντας τη συνάρτηση `photographObject` με παραμέτρους κάμερας c_v, c_K, c_u και να το χρωματίζετε καλώντας τη συνάρτηση `objectPainter` της πρώτης εργασίας με χρήση Gouraud shading.

Συνολικά, θα πρέπει να παράξετε 4 φωτογραφίες του αντικειμένου, μια στην αρχική του θέση και από μία για τα αποτελέσματα των βημάτων (i) - (iii). Κάθε φωτογραφία να αποθηκεύεται τη χρήση της συνάρτησης `imwrite`, με όνομα αρχείου τον αριθμό του βήματος (θεωρώντας ότι η αρχική θέση είναι το βήμα 0) και επέκταση `.jpg`. Αν δεν είχατε υλοποιήσει την πρώτη εργασία ή αν είχατε κάποιο λάθος στην υλοποίηση, μπορείτε να χρησιμοποιήσετε την `objectPainter` κάποια-ς/κάποιου συναδέλφου αρκεί να το δηλώσετε στην αναφορά.

Το αρχείο `hw2.mat` που σας δίνεται, περιέχει τις παραμέτρους του αντικείμενου (V, C, F) καθώς και τιμές για όλες τις παραμέτρους που θεωρούνται γνωστές (παραμέτρους κάμερας, διανύσματα μετατόπισης, άξονες περιστροφής κλπ.) Για διευκόλυνση μπορείτε επίσης να χρησιμοποιήσετε τον σκελετό του `script` επίδειξης (`demo.m`) που δίνεται.

3. Αναφορά με:

- Περιγραφή της λειτουργίας και του τρόπου κλήσης των προγραμμάτων,
- Περιγραφή των συναρτήσεων και παρουσίαση του αντίστοιχου ψευδοκώδικα,
- Τα ενδεικτικά αποτελέσματα που παράγονται από τα demos.

Παρατηρήσεις

- Μην κάνετε τεχνητές περιστροφές της φωτογραφίας προκειμένου να φαίνεται “ίσιο” το αντικείμενο.
- Οι εργασίες αξιολογούνται με χρήση της base Matlab R2017a. Συνεπώς, μη χρησιμοποιείτε συναρτήσεις (π.χ. `vecnorm`, `parcluster`) που δεν περιέχονται στην έκδοση αυτή.
- Οι εργασίες είναι αυστηρά ατομικές.
- Υποβάλετε ένα και μόνο αρχείο, τύπου `zip`.
- Το όνομα του αρχείου πρέπει να είναι `AEM.zip`, όπου `AEM` είναι τα τέσσερα ψηφία του Α.Ε.Μ. του φοιτητή της ομάδας.
- Το προς υποβολή αρχείο πρέπει να περιέχει τα αρχεία κώδικα Matlab και το αρχείο `report.pdf` το οποίο θα είναι η αναφορά της εργασίας.
- Η αναφορά πρέπει να είναι ένα αρχείο τύπου PDF, και να έχει όνομα `report.pdf`.
- Όλα τα αρχεία κώδικα πρέπει να είναι αρχεία κειμένου τύπου UTF-8, και να έχουν κατάληξη `m`.
- Το αρχείο τύπου `zip` που θα υποβάλετε δεν πρέπει να περιέχει κανένα φάκελο.

- Για την ονομασία των αρχείων που περιέχονται στο προς υποβολή αρχείο, χρησιμοποιείτε μόνο αγγλικούς χαρακτήρες, και όχι ελληνικούς ή άλλα σύμβολα, πχ “#”, “\$”, “%” κλπ.

Προσοχή: Θα αξιολογηθούν μόνο όσες εργασίες έχουν demos που τρέχουν!