

# Γραφική με Υπολογιστές 2018

## Εργασία #2: Μετασχηματισμοί και Προβολές

### Ζητούμενα

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

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

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

**B.** Έστω  $c \in \mathbb{R}^3$  η  $3 \times 1$  στήλη με τις συντεταγμένες ενός διανύσματος  $\mathbf{u}$  ως προς κάποιο σύστημα συντεταγμένων και έστω  $\mathcal{L}$  διανυσματικός μετασχηματισμός περιστροφής με πίνακα μετασχηματισμού  $R$ , διάστασης  $3 \times 3$ . Έστω  $d \in \mathbb{R}^3$  οι μετασχηματισμένες συντεταγμένες του διανύσματος  $\mathbf{u}$  ως προς το ίδιο σύστημα. Να υλοποιήσετε τη συνάρτηση:

$$d = \text{vectrans}(c, R)$$

Φροντίστε η `vectrans` να δουλεύει σωστά και στην περίπτωση που τα  $c, d$  είναι  $3 \times n$  πίνακες με τις συντεταγμένες  $n$  διανυσμάτων.

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

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

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.

**Δ.** Έστω  $c_p \in \mathbb{R}^3$  η  $3 \times 1$  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  $3 \times 1$  στήλη με τις συντεταγμένες του διανύσματος  $\mathbf{v}_0$  ως προς το αρχικό σύστημα αξόνων, να υλοποιήσετε τη συνάρτηση:

$$d_p = \text{systemtrans}(c_p, b_1, b_2, b_3, c_0)$$

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.

**Ε.** Έστω  $c_p \in \mathbb{R}^3$  η  $3 \times 1$  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 (με  $w = 1$ ) 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.

**ΣΤ.** 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 up vector  $\mathbf{u}$  αντίστοιχα. Το  $c_v$  περιέχει όπως και πριν τις συντεταγμένες του κέντρου της κάμερας ως προς το WCS.

## Παραδοτέα

- Τα προγράμματα σε μορφή **σχολιασμένου** πηγαίου κώδικα με σχόλια γραμμένα στα **αγγλικά ή greeklish**.
- Script επίδειξης με όνομα `demo2.m`. Το script αυτό θα πρέπει να καλείται χωρίς εξωτερικά ορίσματα, να διαβάζει τη γάτα από το αρχείο `cat3d.mat` 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 τη γάτα. Δοθέντων των σημείων του πίνακα  $V$ , το script σας θα πρέπει να:

1. Τα μετατοπίζει κατά  $t_1 = [-1, -1, 3]^T$ .
2. Τα περιστρέφει κατά γωνία  $\phi = \pi/4$  rad περί άξονα που διέρχεται από το σημείο με συντεταγμένες  $K = [0, 0, 0]^T$  και έχει κατεύθυνση παράλληλη προς το διάνυσμα  $g = [0, 1, 0]^T$ .
3. Τα μετατοπίζει κατά  $t_2 = [1, 1, -3]^T$ .

4. Χρησιμοποιώντας την έξοδο του βήματος 3, υπολογίζει την προοπτική προβολή των σημείων της γάτας για κάμερα με  $c_v = [0, 0, -5]^T$ ,  $c_x = [1, 0, 0]^T$ ,  $c_y = [0, 1, 0]^T$  και  $w = 1$ .
5. Χρησιμοποιώντας την έξοδο του βήματος 3, υπολογίζει την προοπτική προβολή των σημείων της γάτας για κάμερα με  $c_v = [-1, -1, -3]^T$ ,  $c_K = [5, 5, 5]^T$ ,  $c_u = [0, 0, 1]^T$  και  $w = 1$ .

Για τα βήματα 1 - 3, παρουσιάστε την αρχική γάτα καθώς και το αποτέλεσμα μετά από κάθε μετασχηματισμό στο ίδιο σύστημα αξόνων καλώντας διαδοχικά τη συνάρτηση `objectPainter` της πρώτης εργασίας με χρήση της μεθόδου `Gouraud` για τη χρωματική απόδοση των τριγώνων. Με τον ίδιο τρόπο, προβάλτε τα αποτελέσματα των βημάτων 4 - 5 σε κοινό σύστημα αξόνων, διαφορετικό από των βημάτων 1 - 3. Συνολικά, θα πρέπει να παράξετε 2 figures, ένα με τα αποτελέσματα των βημάτων 1 - 3 και ένα με τα αποτελέσματα των βημάτων 4 - 5. Αν δεν έχετε υλοποιήσει την πρώτη εργασία ή αν είχατε λάθος στην υλοποίηση, μπορείτε να χρησιμοποιήσετε την `objectPainter` κάποιας/κάποιου συναδέλφου αρκεί να το δηλώσετε στην αναφορά.

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

## Υποβολή εργασίας

- 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.

**Θα αξιολογηθούν μόνο όσες εργασίες έχουν demos που τρέχουν!**