



Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης
Πολυτεχνική Σχολή
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών

Γραφική με Υπολογιστές 2023
Εργασία 2: Μετασχηματισμοί και Προβολές

Γαλάνη Δήμητρα
ΑΕΜ 10331

Μάθημα: Γραφική με Υπολογιστές
Διδάσκων: Ντελόπουλος Αναστάσιος
Καθηγητής Α.Π.Θ.

9 Μαΐου 2024

Περιεχόμενα

1	A. Κλάση Μετασχηματισμών	3
1.1	Συνάρτηση υπολογισμού πίνακα περιστροφής	3
1.2	Συνάρτηση υπολογισμού πίνακα μετατόπισης	3
1.3	Συνάρτηση μετασχηματισμού σημείων	3
2	Συναρτήσεις	4
2.1	Γ. Συνάρτηση αλλαγής συστήματος συντεταγμένων	4
2.2	Δ. Συνάρτηση προσανατολισμού κάμερας	4
2.3	Ε. Συνάρτηση προοπτικής προβολής με pinhole κάμερα	5
2.4	ΣΤ. Συνάρτηση απεικόνισης	5
2.5	Ζ. Συνάρτηση φωτογράφισης	6
3	Αποτελέσματα που παράγονται από το demo	7
3.1	Αρχική θέση	7
3.2	Περιστροφή κατά γωνία θ περί άξονα παράλληλο προς το <code>rot_axis</code>	7
3.3	Μετατόπιση κατά <code>t_0</code>	8
3.4	Μετατόπιση κατά <code>t_1</code>	9

Κατάλογος σχημάτων

1	Αρχική θέση αντικειμένου.	7
2	Περιστροφή αντικειμένου κατά γωνία θ	8
3	Μετατόπιση αντικειμένου κατά t_0	8
4	Μετατόπιση αντικειμένου κατά t_1	9

1 Α. Κλάση Μετασχηματισμών

Αρχικά, υλοποιούμε μια κλάση για την πραγματοποίηση affine μετασχηματισμών. Ως attribute έχει έναν πίνακα μετασχηματισμού 4×4 . Επίσης, εντός της κλάσης υλοποιούμε τις παρακάτω συναρτήσεις.

1.1 Συνάρτηση υπολογισμού πίνακα περιστροφής

Η συνάρτηση αυτή υπολογίζει τον πίνακα περιστροφής που αντιστοιχεί σε δεξιόστροφη περιστροφή κατά γωνία θ περί του άξονα με κατεύθυνση που δίνεται από το μοναδιαίο διάνυσμα u . Έχει τα παρακάτω ορίσματα:

- θ : γωνία περιστροφής θ .
- u : μοναδιαίο διάνυσμα u το οποίο προσδιορίζει την κατεύθυνση του άξονα περιστροφής.

Η συνάρτηση αυτή δεν επιστρέφει κάτι αλλά μεταβάλλει τον πίνακα `mat` της κλάσης, σύμφωνα με τον τύπο του Rodrigues. Ο τύπος του Rodrigues σύμφωνα με τις σημειώσεις του μαθήματος (5.45) είναι ο εξής:

$$R(\theta, u) = I \cdot \cos \theta + u \cdot u^T \cdot (1 - \cos \theta) + u \cdot \sin \theta \quad (1)$$

1.2 Συνάρτηση υπολογισμού πίνακα μετατόπισης

Στην συνάρτηση αυτή μετασχηματίζεται κατάλληλα ο πίνακας `mat` προκειμένου να ενσωματωθεί και ο μετασχηματισμός μετατόπισης κατά διάνυσμα t . Για να γίνει αυτό προστίθεται το διάνυσμα t στην τέταρτη στήλη του πίνακα `mat`. Η συνάρτηση έχει ως όρισμα:

- t : διάνυσμα μετατόπισης.

Δεν διευκρινίζεται το μέγεθος που πρέπει να έχει το όρισμα t , οπότε θεωρούμε ότι είναι ένας πίνακας στήλη μεγέθους 3×1 .

1.3 Συνάρτηση μετασχηματισμού σημείων

Η συνάρτηση αυτή πραγματοποιεί τον affine μετασχηματισμό των σημείων πολλαπλασιάζοντας τον πίνακα `mat` με τα σημεία, αφού προσθέσουμε μία γραμμή στον πίνακα `pts` για να μπορέσει να πραγματοποιηθεί ο πολλαπλασιασμός. Έχει ως είσοδο:

- `pts`: πίνακας τρισδιάστατων συντεταγμένων σημείων, μεγέθους $N \times 3$.

Μετά τον πολλαπλασιασμό, αφαιρούμε την τελευταία γραμμή του πίνακα η οποία περιέχει την ομογενή συντεταγμένη. Παρόλο που στην εκφώνηση της εργασίας διευκρινίζεται ότι ο πίνακας εισόδου είναι μεγέθους $N \times 3$ δεν αναφέρεται κάτι για τον πίνακα εξόδου οπότε χάριν ευκολίας επιλέγουμε ότι η συνάρτηση επιστρέφει έναν πίνακα μεγέθους $3 \times N$ με τα μετασχηματισμένα σημεία.

2 Συναρτήσεις

2.1 Γ. Συνάρτηση αλλαγής συστήματος συντεταγμένων

Για την μετάβαση από το WCS στο σύστημα συντεταγμένων της κάμερας βασιστήκαμε στον τύπο (6.13) των σημειώσεων του μαθήματος. Οπότε, υλοποιούμε την συνάρτηση `world2view` που πραγματοποιεί τον μετασχηματισμό και έχει ορίσματα:

- `pts`: πίνακας μεγέθους $3 \times N$ με τις συντεταγμένες σημείων ως προς ένα αρχικό σύστημα συντεταγμένων.
- `R`: πίνακας περιστροφής του συστήματος συντεταγμένων μεγέθους 3×3 .
- `c0`: διάνυσμα μετατόπισης του σημείου αναφοράς νέου συστήματος συντεταγμένων.

Και επιστρέφει έναν πίνακα μεγέθους $N \times 3$ με τα μετασχηματισμένα σημεία ως προς το σύστημα συντεταγμένων της κάμερας (CCS). Η συνάρτηση αυτή μετασχηματίζει τα σημεία εισόδου από το WCS στο σύστημα συντεταγμένων της κάμερας, το οποίο είναι περιστραμμένο και μετατοπισμένο ως προς το αρχικό σύστημα. Για την πραγματοποίηση των μετασχηματισμών αυτών δημιουργούμε ένα αντικείμενο τύπου `Transform` και εκμεταλλευόμαστε τις συναρτήσεις της κλάσης μετασχηματισμών που ορίσαμε παραπάνω. Με αυτών τον τρόπο μεταφέρουμε τα σημεία του αντικειμένου στο σύστημα συντεταγμένων της κάμερας με σκοπό την προβολή τους στον φακό.

```
camera_coords = world2view(pts, R, t.T) # Convert the points to the camera coordinate frame
```

2.2 Δ. Συνάρτηση προσανατολισμού κάμερας

Για την υλοποίηση της συνάρτησης `lookat` έχουμε συμβουλευτεί το κεφάλαιο 6.3.1 καθώς και το κεφάλαιο 6.3.2 των σημειώσεων του μαθήματος και χρησιμοποιήσαμε τους τύπους (6.6), (6.7), (6.8), (6.12) προκειμένου να υπολογίσουμε τα x_c, y_c, z_c, R, t . Στην συνέχεια υλοποιούμε την συνάρτηση `lookat` με ορίσματα:

- `eye`: το σημείο του κέντρου της κάμερας 3×1 .
- `up`: το μοναδιαίο `up` διάνυσμα της κάμερας 3×1 .
- `target`: το σημείο στόχος 3×1 .

Και επιστρέφει τις παραμέτρους που χρειάζονται για τον μετασχηματισμό των σημείων από ένα σύστημα συντεταγμένων σε ένα άλλο δηλαδή:

- `R`: πίνακας περιστροφής του συστήματος συντεταγμένων μεγέθους 3×3 .
- `t`: διάνυσμα μετατόπισης σημείου αναφοράς.

Στην συνάρτηση αυτή υπολογίζονται οι παράμετροι που χρειάζονται για να μετασχηματιστούν τα σημεία από το WCS στο CCS, δεδομένης της θέσης της κάμερας ως προς το WCS και το σημείο που πρέπει να στοχεύσει η κάμερα (πιθανόν το κέντρο του αντικειμένου). Ο προσδιορισμός των παραμέτρων αυτών γίνεται σύμφωνα με το κεφάλαιο 6.3.1 των σημειώσεων του μαθήματος. Δηλαδή:

$$\hat{z}_c = \frac{\overrightarrow{CK}}{|\overrightarrow{CK}|} \quad (2)$$

$$\hat{y}_c = \frac{t}{|t|}, \quad t = u - \langle u, \hat{z}_c \rangle \cdot \hat{z}_c \quad (3)$$

$$\hat{x}_c = \hat{y}_c \times \hat{z}_c \quad (4)$$

Σύμφωνα με την σχέση (6.13) των σημειώσεων η μετατόπιση υπολογίζεται από: $t = -R^T \cdot d$, το οποίο d ισούται με το `eye` από την σχέση (6.12).

```
R, t = lookat(eye, up, target)
```

2.3 Ε. Συνάρτηση προοπτικής προβολής με pinhole κάμερα

Στην συνέχεια έχουμε την συνάρτηση `perspective_project` με ορίσματα:

- `pts`: πίνακας των σημείων εισόδου μεγέθους $3 \times N$.
- `focal`: η απόσταση του πετάσματος της κάμερας από το κέντρο.
- `R`: πίνακας περιστροφής του συστήματος συντεταγμένων μεγέθους 3×3 . (περιστροφή προς το σύστημα συντεταγμένων της κάμερας.)
- `t`: διάνυσμα μετατόπισης συστήματος συντεταγμένων 1×3

Η συνάρτηση αυτή παράγει τις προοπτικές προβολές και το βάθος των σημείων εισόδου. Αρχικά, μεταφέρει τα σημεία εισόδου στο σύστημα συντεταγμένων της κάμερας καλώντας την συνάρτηση `world2view` και ορίζει την τρίτη συντεταγμένη ως το βάθος του σημείου. Έπειτα παράγει τις προοπτικές προβολές σύμφωνα με τον τύπο:

$$A''B'' = \frac{AB \cdot focal}{depth} \quad (5)$$

Ο τύπος αυτός ταυτίζεται με τον τύπο (6.2) των σημειώσεων του μαθήματος. Τελικά επιστρέφει δύο πίνακες:

- `projected_pts`: πίνακας συντεταγμένων των σημείων στο πέτασμα της κάμερας μεγέθους $2 \times N$.
- `t`: διάνυσμα μετατόπισης σημείου αναφοράς.

```
projected_pts, depth_values = perspective_project(v_pos.T, focal, R, t)
```

2.4 ΣΤ. Συνάρτηση απεικόνισης

Στην συνέχεια υλοποιούμε την συνάρτηση `rasterize` η οποία έχει τα παρακάτω ορίσματα:

- `pts_2d`: πίνακας των σημείων εισόδου μεγέθους $2 \times N$.
- `plane_w`: το πλάτος του πλάνου της κάμερας.
- `plane_h`: το μήκος του πλάνου της κάμερας.
- `res_w`: το πλάτος του πλάνου της εικόνας.
- `res_h`: το μήκος του πλάνου της εικόνας.

Σκοπός της συνάρτησης είναι να απεικονίσει τα σημεία από το σύστημα συντεταγμένων του πετάσματος της κάμερας στην εικόνα, δηλαδή να αντιστοιχίσει τα σημεία σε ακέραιες θέσεις `pixel` του πλάνου της εικόνας. Είναι σημαντικό να λάβουμε υπόψη μας ότι ο άξονας της κάμερας περνάει από το κέντρο του πλάνου της κάμερας, ενώ η αρίθμηση του πλάνου της εικόνας ξεκινά από κάτω προς τα πάνω και από αριστερά προς τα δεξιά. Αρχικά, κεντράρουμε τα σημεία μετακινώντας τα κατά το ένα δεύτερο της διάστασης της εικόνας έτσι ώστε να ευθυγραμμιστούν με το κάτω αριστερό άκρο της εικόνας. Στην συνέχεια, πολλαπλασιάζουμε τα σημεία με τον συντελεστή κλιμάκωσης για να συμβαδίσουν με την ανάλυση της εικόνας και τέλος στρογγυλοποιούμε προκειμένου η εκάστοτε συντεταγμένη να αντιστοιχηθεί με το κοντινότερο `pixel`.

```
pixel_coords = rasterize(projected_pts, plane_w, plane_h, res_w, res_h)
```

2.5 Ζ. Συνάρτηση φωτογράφισης

Η συνάρτηση `render_object` υλοποιεί όλο το pipeline της απεικόνισης ενός αντικειμένου και έχει ως ορίσματα

- `v_pos`: πίνακας μεγέθους $N \times 3$ με τις τρισδιάστατες συντεταγμένες σημείων του αντικειμένου.
- `v_clr`: πίνακας μεγέθους $N \times 3$ με τα χρώματα των κορυφών σε RGB τιμές.
- `t_pos_idx`: πίνακας μεγέθους $F \times 3$, κάθε γραμμή περιέχει τους δείκτες που αντιστοιχούν στις τρεις κορυφές ενός τριγώνου. Χρησιμοποιούνται στην προσπέλαση των πινάκων `v_pos` και `v_clr` για τον εντοπισμό των πληροφοριών του κάθε τριγώνου.
- `plane_h`: ύψος του πετάσματος της κάμερας.
- `plane_w`: πλάτος του πετάσματος της κάμερας.
- `res_h`: ύψος του καμβά σε pixel.
- `res_w`: πλάτος του καμβά σε pixel.
- `focal`: απόσταση του πετάσματος από το κέντρο της κάμερας.
- `eye`: κέντρο της κάμερας ως προς το WCS.
- `up`: διάνυσμα της κάμερας.

Αρχικά με την βοήθεια της συνάρτησης `lookat` υπολογίζει τον πίνακα περιστροφής και τον πίνακα μετατόπισης, δεδομένης της θέσης της κάμερας. Έπειτα, παράγονται οι προοπτικές προβολές με την συνάρτηση `perspective_project`, αφότου τα σημεία έχουν μετατραπεί σε σημεία που αντιστοιχούν στο σύστημα συντεταγμένων της κάμερας εσωτερικά της συνάρτησης αυτής. Στην συνέχεια τα σημεία της κάμερας αντιστοιχίζονται σε σημεία που ανήκουν στην εικόνα που πρόκειται να προβληθεί δηλαδή αντιστοιχίζονται σε pixel. Τέλος, αφού γίνουν οι κατάλληλες προσαρμογές στους πίνακες χρησιμοποιούμε την συνάρτηση `render_img` της προηγούμενης εργασίας, η οποία είναι υπεύθυνη για τον χρωματισμό των τριγώνων που δημιουργούν τα σημεία με σειρά προτεραιότητας που καθορίζεται από το βάθος του κάθε τριγώνου. Τα σημεία, τα χρώματα των κορυφών, οι τριάδες σημείων που σχηματίζουν ένα τρίγωνο δίνονται από το αρχείο `hw2.npy`. Επιλέγεται σύμφωνα με την εκφώνηση της εργασίας ο χρωματισμός κατά την μέθοδο Gouraud. Τέλος, επιστρέφεται η εικόνα που είναι ένας τρισδιάστατος πίνακας $res_w \times res_h \times 3$ όπου κάθε στοιχείο του αντιπροσωπεύει μια θέση στον καμβά και το χρώμα που κατέχει.

```
image = t.render_object(v_pos_rotated.T, v_clr, t_pos_idx, plane_h, plane_w, res_h,
                        res_w, focal, eye, up, target)
```

3 Αποτελέσματα που παράγονται από το demo

Το demo δεν απαιτεί εξωτερικά ορίσματα για να τρέξει. Για να κληθεί αρκεί να το τρέξουμε σε ένα command line ή να το τρέξουμε με όποιον άλλο τρόπο θέλουμε, π.χ.

```
python demo.py
```

Στο demo αρχικά φορτώνονται τα δεδομένα από το αρχείο hw2.npy και αφότου αντιστοιχηθούν με τους κατάλληλους πίνακες-ορίσματα της συνάρτησης `render_object` καλούμε την συνάρτηση αυτή. Θεωρούμε ότι τα χρώματα των κορυφών από το αρχείο που μας δίνεται βρίσκεται σε μορφή RGB. Η βιβλιοθήκη `opencv`, η οποία χρησιμοποιείται για να αποθηκευτεί η εικόνα που παράγεται από την συνάρτηση, έχει κωδικοποίηση χρωμάτων BGR, ενώ ο πίνακας `image` που έχει παραχθεί περιέχει τα χρώματα σε μορφή RGB. Οπότε είναι αναγκαίο προτού αποθηκεύσουμε την εικόνα σε κάθε βήμα να κάνουμε την μετατροπή από RGB σε BGR. Αξίζει να σημειωθεί ότι στην βιβλιοθήκη `opencv` στις εικόνες η αρχή των αξόνων βρίσκεται στην πάνω αριστερή γωνία. Το γεγονός αυτό πρέπει να το έχουμε κατά νου εάν θέλουμε να γνωρίζουμε την μορφή του αντικειμένου σε καρτεσιανές συντεταγμένες που ξεκινάνε από την κάτω αριστερή γωνία.

3.1 Αρχική θέση

Ως είσοδο σημείων παίρνουμε αρχικά τον πίνακα `v_pos` που υπάρχει στο αρχείο hw2.npy και καλούμε την συνάρτηση `render_object` με τα κατάλληλα ορίσματα. Αποθηκεύουμε την εικόνα αυτήν ως `0.jpg`.



Σχήμα 1: Αρχική θέση αντικειμένου.

3.2 Περιστροφή κατά γωνία θ περί άξονα παράλληλο προς το `rot_axis`

Στην συνέχεια, δημιουργούμε ένα αντικείμενο της κλάσης `Transform`, προκειμένου να μετασχηματίσουμε τα σημεία. Θέτουμε τον πίνακα μετασχηματισμού, έτσι ώστε τα σημεία να περιστραφούν κατά γωνία θ ως προς τον επιθυμητό άξονα, και μετασχηματίζουμε τα σημεία με τις συναρτήσεις της κλάσης. Καλώντας την συνάρτηση `render_object` έχουμε σαν είσοδο σημείων τα μετασχηματισμένα σημεία `v_pos_rotated`.



Σχήμα 2: Περιστροφή αντικειμένου κατά γωνία θ .

Παρατηρούμε ότι το αντικείμενο έχει περιστραφεί δεξιόστροφα κατά γωνία θ γύρω από τον άξονα περιστροφής $\text{rot_axis_0} = (0, 1, 0)$, δηλαδή κατά τον άξονα y . Η εικόνα αυτή αποθηκεύεται ως 1.jpg.

3.3 Μετατόπιση κατά t_0

Σε αυτό το βήμα αρχικοποιούμε τον πίνακα μετασχηματισμού του αντικειμένου και ύστερα καλώντας την συνάρτηση `transform_obj.translate(t_0)` θέτουμε τον πίνακα, έτσι ώστε τα σημεία να μετατοπιστούν κατά t_0 . Μετασχηματίζουμε τα σημεία `v_pos_rotated`, έτσι ώστε το αντικείμενο μετά την περιστροφή να μετατοπιστεί. Τέλος, καλούμε πάλι την συνάρτηση `render_object` έχοντας σαν είσοδο σημείων τα εκ νέου μετασχηματισμένα σημεία `v_pos_rot_trans1`.



Σχήμα 3: Μετατόπιση αντικειμένου κατά t_0 .

Παρατηρούμε ότι το αντικείμενο έχει μετατοπιστεί προς τα δεξιά το οποίο είναι λογικό δεδομένου ότι το διάνυσμα μετατόπισης είναι το εξής: $t_0 = (1, 0, 0)$. Δηλαδή, οι συντεταγμένες του x άξονα του αντικειμένου θα αυξηθούν κατά μία μονάδα, δηλαδή το αντικείμενο θα μετατοπιστεί προς τα δεξιά, προς τα αυξανόμενα του άξονα. Η εικόνα αυτή αποθηκεύεται ως 2.jpg.

3.4 Μετατόπιση κατά t_1

Τέλος, σε αυτό το βήμα θα κάνουμε πάλι την ίδια διαδικασία με το προηγούμενο. Αρχικοποιούμε τον πίνακα μετασχηματισμού του αντικειμένου και ύστερα καλώντας την συνάρτηση `transform_obj.translate(t_1)` θέτουμε τον πίνακα, έτσι ώστε τα σημεία να μετατοπιστούν κατά t_1 . Μετασχηματίζουμε τα σημεία `v_pos_rot_trans1`, έτσι ώστε το αντικείμενο μετά την περιστροφή και την πρώτη μετατόπιση να μετατοπιστεί ξανά. Τέλος, καλούμε πάλι την συνάρτηση `render_object` έχοντας σαν είσοδο σημείων τα εκ νέου μετασχηματισμένα σημεία `v_pos_rot_trans1_trans2`.



Σχήμα 4: Μετατόπιση αντικειμένου κατά t_1 .

Σε αυτήν την περίπτωση παρατηρούμε ότι το αντικείμενο έχει μετατοπιστεί προς τα εμπρός το οποίο είναι λογικό δεδομένου ότι το διάνυσμα μετατόπισης είναι το εξής: $t_1 = (0, 0, -1)$. Δηλαδή, οι συντεταγμένες του z άξονα του αντικειμένου θα μειωθούν κατά μία μονάδα, δηλαδή το αντικείμενο θα μετατοπιστεί προς τα κάτω στο καρτεσιανό σύστημα συντεταγμένων, προς τα αυξανόμενα του αρνητικού άξονα. Δεδομένου ότι η κάμερα βρίσκεται στο σημείο $(0, 0, -35)$ με την μείωση της z συντεταγμένης, το αντικείμενο θα πλησιάσει την κάμερα με αποτέλεσμα να το βλέπουμε πιο μεγάλο, δηλαδή μετατοπισμένο προς τα εμπρός. Η εικόνα αυτή αποθηκεύεται ως 3.jpg.