

Γραφική με Υπολογιστές

Εργασία #2: Μετασχηματισμοί και Προβολές

ΟΝ/ΕΠ : Νικόλαος Ιστατιάδης

AEM : 9175

Email: nikoista@ece.auth.gr

Ζητούμενα

Το script 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, ο οποίος περιγράφεται παρακάτω:

Ως είσοδο χρησιμοποιείτε τον πίνακα $V_{K \times 3}$, οποίος 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 ακόλουθα βήματα:

(α') Τα μετατοπίζει κατά $t1$.

(β') Τα περιστρέφει κατά γωνία ϕ rad περί άξονα που διέρχεται από το σημείο O και έχει κατεύθυνση παράλληλη προς διάνυσμα g .

(γ') Τα μετατοπίζει κατά $t2$

Κάθε βήμα θα δέχεται ως είσοδο την έξοδο του προηγούμενου. Μετά από κάθε βήμα θα πρέπει να φωτογραφίζετε το αντικείμενο, καλώντας τη συνάρτηση `render_object` με παραμέτρους κάμερας `cn`, `cK`, `cu` και να το χρωματίζετε καλώντας τη συνάρτηση `render` της πρώτης εργασίας με τη χρήση `Gouraud shading`.

A. Πίνακας Μετασχηματισμού

Κλάση transformation matrix

Η οποία υλοποιεί έναν πίνακα μετασχηματισμού τύπου affine. Η κλάση θα έχει ως μέλη έναν πίνακα $T_{4 \times 4}$ και δύο μεθόδους, την `rotate(θ , u)` και `translate(t)`.

A1. rotate(θ , u)

Θα υπολογίζει τον πίνακα περιστροφής Rodrigues κατά γωνία θ και άξονα που διέρχεται από την αρχή του συστήματος συντεταγμένων και είναι παράλληλος προς το u , και θα ενημερώνει τον πίνακα T της κλάσης `transformation_matrix` κατάλληλα. Σύμφωνα με τις παρουσιάσεις του μαθήματος και τις σημειώσεις εφαρμόζουμε τους τύπους και καταλήγουμε στο επιθυμητό αποτέλεσμα.

A2. translate(t)

Η μέθοδος `translate(t)`, θα δέχεται ως όρισμα ένα διάνυσμα μετατόπισης t και θα ενημερώνει κατάλληλα τον πίνακα T της κλάσης `transformation_matrix`.

B. Συνάρτηση μετασχηματισμού τύπου affine**Συνάρτηση affine transform(cp , T)**

Μετασχηματισμός Affine για συντεταγμένες ενός σημείου cp . Πρώτα το διάνυσμα στήλη των συντεταγμένων του cp υφίσταται γραμμικό διανυσματικό μετασχηματισμό $L * c$, ως προς το ίδιο σύστημα συντεταγμένων και μετά μια μετατόπιση κατά διάνυσμα $t1$ (στήλη). Η συνάρτηση θα υλοποιεί σημειακό μετασχηματισμό affine εφαρμόζοντας τον μετασχηματισμό που προκύπτει από τον πίνακα $T_{4 \times 4}$ των αντικειμένων τύπου `transformation_matrix`.

Γ. Συνάρτηση μετασχηματισμού συστήματος συντεταγμένων**Συνάρτηση system transform(cp , T , $c0$)**

Μετασχηματισμός συστήματος συντεταγμένων (μετατόπιση αρχής και μετασχηματισμό διανυσμάτων βάσης) και έκφραση του δοθέντος διανύσματος ως προς το καινούριο σύστημα σύμφωνα με τον γραμμικό μετασχηματισμό $L - 1 * c$.

Δ. Συνάρτηση προοπτικής κάμερας

Συνάρτηση project_cam(w, cv, cx, cy, cz, p)

Υπολογισμός των μοναδιαίων διανυσμάτων της κάμερας (αν και όσα λείπουν), μετασχηματισμός του συστήματος συντεταγμένων ως προς την κάμερα και προβολή όλων των σημείων πάνω στο πέτασμά της, ενώ έχουν δοθεί ως δεδομένα 2 από τα μοναδιαία διανύσματα του CCS (ως προς το WCS) και το κέντρο της κάμερας (ως προς το WCS).

Ε. Συνάρτηση προοπτικής κάμερας

Συνάρτηση project_cam_ku(w, cv, clookat, cup, p)

Υπολογισμός των μοναδιαίων διανυσμάτων της κάμερας (αν και όσα λείπουν), μετασχηματισμός του συστήματος συντεταγμένων ως προς την κάμερα και προβολή όλων των σημείων πάνω στο πέτασμά της, ενώ έχουν δοθεί ως δεδομένα το κέντρο της κάμερας (ως προς το WCS), ο στόχος (ως προς το WCS) και το up vector της (ως προς το WCS) και κάνοντας χρήση της συνάρτησης project() στο σημείο που είναι απαραίτητο.

ΣΤ. Συνάρτηση απεικόνισης

Συνάρτηση rasterize(P,M,N,H,W)

Απεικόνιση των συντεταγμένων των σημείων από το σύστημα της 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. Συνάρτηση φωτογράφισης

Συνάρτηση `render_object(p, F,C,M,N,H,W,w, cv, clookat, cup)`

Εφαρμόζει κατάλληλα τις παραπάνω συναρτήσεις για να υλοποιήσει όλο το pipeline της φωτογράφισης-ρέντερ ενός αντικειμένου.

Στην συνέχεια με κατάλληλο συνδυασμό αυτών έπρεπε να υλοποιήσουμε τα 0,1,2,3 βήματα σύμφωνα με τα σχόλια του `demo_samples.m`.

Μέσα στον κώδικα του `demo_samples.m` γίνεται εφαρμογή των μετασχηματισμών που ζητούνται αλλά και γίνεται μια “φωτογράφιση” του αντικειμένου καλώντας την συνάρτηση `render_object`.

Στο εσωτερικό της `render_object` υπολογίζουμε τις προοπτικές προβολές με την χρήση της συνάρτησης `project_cam_ku` και μας επιστρέφει τους πίνακες `P2d` και `D` όπου είναι τα διανύσματα του αντικειμένου στις 2- διαστάσεις και το βάθος του (δηλαδή η 3-διάσταση). Έστερα δίνουμε την παραγόμενη έξοδο της `project_cam_ku`, την `P2d`, στην συνάρτηση `rasterize` και εκεί γίνεται απεικόνιση των συντεταγμένων των σημείων από το σύστημα της κάμερας με πέτασμα διάστασης `H x W` (σε ίντσες), σε ακέραιες θέσεις (pixels) της εικόνας, διάστασης `M x N`.

Μέσα στην `project_cam_ku` βρίσκεται η `project_cam` και μέσα σε αυτή η `system_transform`. Έτσι επιτυγχάνεται η σωστή εφαρμογή από άποψη σειράς των συναρτήσεων και καταλήγουμε στο επιθυμητό αποτέλεσμα.

Εκτενέστατη περιγραφή υπάρχει σε μορφή σχολίων μέσα σε όλους τους κώδικες των συναρτήσεων.

Αποτελέσματα

Αρχικά δίνεται το αρχείο hw2.mat όπου εισέρχονται όλες οι απαραίτητες πληροφορίες για την εργασία μας.

Βήμα 0:

Εφαρμόζουμε σωστό initialize και αρχικοποιούμε το σύστημα μας για περεταίρω μετασχηματισμούς. Άρα κάνω χρήση της `distance0 = system_transform(V,eye(3),O)`, με ορίσματα V οι συντεταγμένες που έχουμε από το αρχείο hw2.mat ,τον πίνακα I3 , εφόσον η βάση διατηρείται σταθερή και O το κέντρο τον αξόνων. Χρησιμοποιούμε την `render_object` με όρισμα `p=distance0` και δίνουμε τα δεδομένα της στην `paint_triangle_gouraud` της προηγούμενης εργασίας για να ζωγραφίσει το αντικείμενο όπου φαίνεται στην



ΕΙΚΟΝΑ 0.

Βήμα 1:

Στο σημείο αυτό μας ζητείται να μετατοπίσουμε τα σημεία του Βήματος 0 κατά ένα διάνυσμα $t1$ το οποίο είναι και αυτό εκφρασμένο ως προς το WCS. Δηλαδή κάνουμε έναν γραμμικό μετασχηματισμό Affine με πίνακα γραμμικού μετασχηματισμού ίσο με τον μοναδιαίο $I3 = 3 \times 3$ (ώστε να μην έχω μετασχηματισμό του διανύσματος συντεταγμένων του σημείου P αλλά μόνο μετατόπιση) και συντεταγμένες του σταθερού διανύσματος μετατόπισης $t1$ 3×1 , δηλαδή του ct . Ο πίνακας αυτός δημιουργείτε μέσω του constructor που έχει η κλάση `transformation_matrix`, συνεπώς δημιουργώ ένα αντικείμενο AT με πίνακα T που είναι $I4 = 4 \times 4$. Κάνω χρήση της `distance1 = affine_transform(distance0', AT.T, t1)`, με ορίσματα τον $AT.T = 4 \times 4$ και θα χρησιμοποιήσω μόνο τον 3×3 πίνακα του T , εφόσον η βάση διατηρείται σταθερή, `distance0` οι συντεταγμένες που είχαμε από το προηγούμενο ερώτημα ως προς το WCS. Χρησιμοποιούμε την `render_object` με όρισμα $p = distance1$ και δίνουμε τα δεδομένα της στην `paint_triangle_gouraud` της προηγούμενης εργασίας για να ζωγραφίσει το αντικείμενο όπου φαίνεται στην Εικόνα 1.

**ΕΙΚΟΝΑ 1.**

Βήμα 2:

Εδώ ρέπει να περιστρέψουμε τα σημεία περί άξονα που διέρχεται από ένα σημείο K (Αρχή των αξόνων O) και έχει κατεύθυνση παράλληλη προς το διάνυσμα g κατά γωνία θ . Εδώ πρέπει να εφαρμόσω σταδιακά τα εξής βήματα: Αρχικά πρέπει να μετατοπίσω το σύστημα συντεταγμένων (κρατώντας σταθερή την βάση και μετατοπίζοντας το κέντρο) ώστε να βρισκόμαστε στο νέο κέντρο K , για να μπορώ να χρησιμοποιήσω τον τύπο της περιστροφής του Rodrigues. Γι' αυτό χρησιμοποιώ την `new_distance=system_transform(distance1 ,AT.T , K)` με ορίσματα `distance1` οι συντεταγμένες που έχουμε από το βήμα 1, τον πίνακα `AT.T` (4×4) αλλά θα χρησιμοποιήσουμε των `I3` πίνακα του `T` εφόσον η βάση διατηρείται σταθερή και $K=O$ οι συντεταγμένες του κέντρου τον αξόνων. Στην συνέχεια κάνω χρήση της `AT.translate(cz)` και `affine_transform (new_distance', AT.T)` η οποία έχει ως ορίσματα τον πίνακα περιστροφής R που εμπεριέχεται στον 4×4 πίνακα T τον οποίο παίρνουμε καλώντας την `AT.rotate(theta,g)` .Ως g ορίζουμε το μοναδιαίο διάνυσμα παράλληλο προς τον άξονα περιστροφής, τις συντεταγμένες που πήραμε από το προηγούμενο βήμα εκφρασμένες ως προς το νέο σύστημα συντεταγμένων(με κέντρο το $K=O$) και τέλος το διάνυσμα `cz` 3×1 που είναι ένα μηδενικό διάνυσμα. Δηλαδή εφαρμόζω ένα γραμμικό μετασχηματισμό τύπου Affine με πίνακα γραμμικού μετασχηματισμού τον $L=R$ αλλά χωρίς μετατόπιση. Και τέλος εφαρμόζω πάλι την `rotation1 = system_transform(final_distance, eye(3) ,-K)` προκειμένου να επαναφέρω τα σημεία ως προς το αρχικό σύστημα συντεταγμένων. Χρησιμοποιούμε την `render_object` με όρισμα `p=rotation1` και δίνουμε τα δεδομένα της στην `paint_triangle_gouraud` της προηγούμενης εργασίας για να ζωγραφίσει το αντικείμενο όπου φαίνεται στην Εικόνα 2.



ΕΙΚΟΝΑ 2.

Βήμα 3:

Τέλος, μας ζητάει να τα μετατοπίσουμε κατά ένα διάνυσμα $t2$ οπότε πάλι κάνουμε το ίδιο με το βήμα 1 εφόσον από το προηγούμενο ερώτημα έχουμε τις συντεταγμένες εκφρασμένες ως προς την αρχή των αξόνων και εφαρμόζω την `AT.translate(t2)` και `distance2 = affine_transform(rotation1', AT.T)`. Χρησιμοποιούμε την `render_object` με όρισμα `p=distance2` και δίνουμε τα δεδομένα της στην `paint_triangle_gouraud` της προηγούμενης εργασίας για να ζωγραφίσει το αντικείμενο όπου φαίνεται στην Εικόνα 3.



ΕΙΚΟΝΑ 3.