

Γραφική με Υπολογιστές 2022

Εργασία #3: Θέαση

Στην εργασία αυτή καλείστε να χρησιμοποιήσετε τους αλγορίθμους που υλοποιήθηκαν στις δύο προηγούμενες εργασίες με κάποιες αλλαγές προκειμένου να δημιουργηθεί ένα πλήρες πλαίσιο δημιουργίας φωτογραφιών μίας εικονικής σκηνής.

Ζητούμενα

A. Φωτισμός και Υλικό επιφάνειας

Να δημιουργηθεί η συνάρτηση:

$$I = \text{light}(\text{point}, \text{normal}, \text{vcolor}, \text{cam_pos}, k_a, k_d, k_s, n, l_{pos}, l_{int})$$

η οποία υπολογίζει το φωτισμό ενός σημείου *point*, το οποίο ανήκει σε μία επιφάνεια με υλικό τύπου-Phong λόγω του διάχυτου φωτισμού από το περιβάλλον, διάχυτης ανάκλασης, και κατοπτρικής ανάκλασης, με:

- *point* είναι ένα διάνυσμα-στήλη 1×3 με τις συντεταγμένες του σημείου της επιφάνειας.
- *normal* είναι ένα διάνυσμα-στήλη 1×3 με τις συντεταγμένες του κανονικού διανύσματος της επιφάνειας στο σημείο *point* (δηλαδή του κάθετου διανύσματος προς την επιφάνεια). Το διάνυσμα έχει φορά προς το εξωτερικό της επιφάνειας, δηλαδή προς την πλευρά του παρατηρητή.
- $\text{color} = [c_r, c_g, c_b]^T$ είναι το 1×3 διάνυσμα με τις συνιστώσες του χρώματος του σημείου *P*. Κάθε συνιστώσα ανήκει στο διάστημα $[0, 1]$.
- *cam_pos* είναι ένα διάνυσμα-στήλη 1×3 με τις συντεταγμένες του παρατηρητή (δηλαδή της κάμερας).
- k_a είναι ο συντελεστής διάχυτου φωτός από το περιβάλλον.
- k_d είναι ο συντελεστής διάχυτης ανάκλασης.
- k_s είναι ο συντελεστής κατοπτρικής ανάκλασης.
- *n* είναι η σταθερά Phong.
- l_{pos} είναι η θέση μίας (ή πολλών) σημειακών πηγών φωτισμού ($N \times 3$).

- l_{int} είναι η ένταση μίας (ή πολλών) σημειακών πηγών φωτισμού ($N \times 3$).

Η συνάρτηση υπολογίζει την ένταση της τριχρωματικής ακτινοβολίας $I = [I_r, I_g, I_b]^T$, που ανακλάται από το σημείο P . Η ένταση συνεισφέρει αθροιστικά στο χρώμα του εικονοστοιχείου (pixel). Η συνάρτηση θα πρέπει να λειτουργεί σωστά τόσο για μία, όσο και για περισσότερες σημειακές πηγές φωτισμού, αθροίζοντας τη συνεισφορά κάθε πηγής.

B. Shading

B.1 Υπολογισμός κανονικών διανυσμάτων επιφάνειας

Έστω 3D αντικείμενο που αποτελείται αποκλειστικά από N_T τρίγωνα. Να υλοποιηθεί η συνάρτηση:

$$normals = calculate_normals(verts, faces)$$

όπου:

- $verts$ είναι ένας πίνακας $3 \times N_v$ με τις συντεταγμένες των κορυφών του αντικειμένου.
- $faces$ είναι ένας πίνακας $3 \times N_T$ που περιγράφει τα τρίγωνα. Η k -οστή στήλη του $faces$ περιέχει τους αύξοντες αριθμούς των κορυφών του k -οστού τριγώνου του αντικειμένου, $1 \leq k \leq N_T$. Η σειρά παράθεσης των κορυφών σηματοδοτεί με τον κανόνα του δεξιόστροφου κοχλία την κατεύθυνση του κανονικού διανύσματος και άρα και προς τα που είναι η εξωτερική πλευρά του αντικειμένου.

Η συνάρτηση υπολογίζει τον $3 \times N_v$ πίνακα με τις συντεταγμένες των κάθετων διανυσμάτων σε κάθε σημείο (κορυφή) της επιφάνειας που ορίζει το αντικείμενο.

B2. Συνάρτηση φωτογράφισης

Να υλοποιήσετε τη συνάρτηση:

$$img = render_object(shader, \\focal, eye, lookat, up, bg_color, \\M, N, H, W, \\verts, vert_colors, faces, \\k_a, k_d, k_s, n, l_{pos}, l_{int}, l_{amb})$$

η οποία δημιουργεί την έγχρωμη φωτογραφία img ενός 3D αντικειμένου, υπολογίζοντας το χρώμα με βάση τα μοντέλα φωτισμού της ενότητας A. Συγκεκριμένα υλοποιεί την παρακάτω διαδικασία:

1. Υπολογίζει τα κανονικά διανύσματα των κορυφών του αντικειμένου μέσω της συνάρτησης $calculate_normals$.
2. Προβάλλει τις κορυφές των τριγώνων σε ορθογώνιο πέτασμα κάμερας με χρήση της συνάρτησης $perspective_project$ της εργασίας #2. Τα τρίγωνα με κορυφή εκτός του πετάσματος δεν θα χρωματίζονται.
3. Καλεί επανειλημμένα την αντίστοιχη συνάρτηση πλήρωσης που έχει επιλεγεί βάσει της τιμής της μεταβλητής $shader$ για να χρωματίσει κάθε τρίγωνο του αντικειμένου, ξεκινώντας από εκείνα με το μεγαλύτερο βάθος (όπως και στις προηγούμενες εργασίες).

Τα ορίσματα που δέχεται έχουν ως εξής:

- *shader* είναι δυαδική μεταβλητή ελέγχου που χρησιμοποιείται για να επιλέξει τη συνάρτηση (που περιγράφονται στη συνέχεια) που θα χρησιμοποιηθεί για την πλήρωση των τριγώνων. Θεωρείστε ότι *shader* = "gouraud" αντιστοιχεί σε Gouraud shading, και *shader* = "phong" σε Phong shading.
- *focal* είναι η απόσταση του πετάσματος από το κέντρο της κάμερας (στις μονάδες μέτρησης που χρησιμοποιεί το σύστημα συντεταγμένων της κάμερας).
- *eye* το 3×1 διάνυσμα με τις συντεταγμένες του κέντρου της κάμερας.
- *lookat* το 3×1 διάνυσμα με τις συντεταγμένες του σημείου στόχου της κάμερας.
- *up* το 3×1 μοναδιαίο *up* διάνυσμα της κάμερας.
- *bg_color* το 3×1 διάνυσμα με τις χρωματικές συνιστώσες του φόντου.
- *M*, *N* είναι οι διαστάσεις της παραγόμενης εικόνας σε pixel (δηλαδή $M \times N_{pixels}$).
- *H*, *W* περιγράφουν τις φυσικές διαστάσεις του πετάσματος της κάμερας σε μονάδες μήκους ταυτόσημες με αυτές που χρησιμοποιούνται στο σύστημα συντεταγμένων της κάμερας.
- *verts* είναι ένας πίνακας $3 \times N_v$ με τις συντεταγμένες των κορυφών του αντικειμένου.
- *verts_colors* είναι ένας πίνακας $3 \times N_v$ με τις συνιστώσες του χρώματος κάθε κορυφής του αντικειμένου.
- *faces* όπως ορίστηκε για τη συνάρτηση *calculate_normals*.
- k_a είναι ο συντελεστής διάχυτου φωτός από το περιβάλλον.
- k_d είναι ο συντελεστής διάχυτης ανάκλασης.
- k_s είναι ο συντελεστής κατοπτρικής ανάκλασης.
- *n* είναι η σταθερά Phong.
- l_{pos} είναι η θέση μίας (ή πολλών) σημειακών πηγών φωτισμού ($N \times 3$).
- l_{int} είναι η ένταση μίας (ή πολλών) σημειακών πηγών φωτισμού ($N \times 3$).
- $light_amb = [I_r, I_g, I_b]^T$ είναι το 3×1 διάνυσμα με τις συνιστώσες της έντασης της διάχυτης ακτινοβολίας του περιβάλλοντος. Κάθε συνιστώσα ανήκει στο διάστημα $[0, 1]$.

Για δεδομένο τρίγωνο, να υπολογίζετε τα διανύσματα *V* και *L* των μοντέλων φωτισμού (με αναφορά στα σχήματα των σημειώσεων για τα μοντέλα φωτισμού ¹⁾ μία φορά χρησιμοποιώντας ως σημείο *P* το κέντρο βάρους του τριγώνου (πριν την προβολή του) και να τα θεωρείτε σταθερά για όλα τα σημεία του τριγώνου.

¹⁾δηλαδή ως *L* και *V* εδώ εννοούνται τα διανύσματα από ένα σημείο *P* προς την πηγή φωτός και τον παρατηρητή αντίστοιχα, βλέπε σχήμα 8.3 σελίδα 99

B3. Gouraud Shading

Να υλοποιηθεί η συνάρτηση:

$$Y = \text{shade_gouraud}(verts_p, verts_n, verts_c, bcoords, cam_pos, k_a, k_d, k_s, n, l_{pos}, l_{int}, l_{amb}, X)$$

η οποία υπολογίζει το χρώμα στις κορυφές του δοθέντος τριγώνου με βάση το πλήρες μοντέλο φωτισμού (χρησιμοποιώντας δηλαδή τις συναρτήσεις της ενότητας A) και στη συνέχεια χρησιμοποιεί γραμμική παρεμβολή χρώματος για την εύρεση του χρώματος στα εσωτερικά σημεία του τριγώνου, με την *interpolate_color* της εργασίας #1. Πιο αναλυτικά:

- Ο πίνακας $verts_p$, διάστασης 3×3 περιέχει τις συντεταγμένες των κορυφών του τριγώνου μετά την προβολή τους στο πέτασμα της κάμερας.
- Ο πίνακας $verts_n$ διάστασης 3×3 περιέχει στις στήλες του τα κανονικά διανύσματα των κορυφών του τριγώνου.
- Ο πίνακας $verts_c$, διάστασης 3×3 περιέχει τις συνιστώσες χρώματος για κάθε σημείο του τριγώνου.
- Το διάνυσμα $bcoords$, διάστασης 3×1 περιέχει το κέντρο βάρους του τριγώνου πριν την προβολή του.
- cam_pos είναι ένα διάνυσμα-στήλη 3×1 με τις συντεταγμένες του παρατηρητή (δηλαδή της κάμερας).
- k_a είναι ο συντελεστής διάχυτου φωτός από το περιβάλλον.
- k_d είναι ο συντελεστής διάχυτης ανάκλασης.
- k_s είναι ο συντελεστής κατοπτρικής ανάκλασης.
- n είναι η σταθερά Phong.
- l_{pos} είναι η θέση μίας (ή πολλών) σημειακών πηγών φωτισμού ($N \times 3$).
- l_{int} είναι η ένταση μίας (ή πολλών) σημειακών πηγών φωτισμού ($N \times 3$).
- $l_{amb} = [I_r, I_g, I_b]^T$ είναι το 1×3 διάνυσμα με τις συνιστώσες της έντασης της διάχυτης ακτινοβολίας του περιβάλλοντος. Κάθε συνιστώσα ανήκει στο διάστημα $[0, 1]$.
- X είναι εικόνα (πίνακας διάστασης $M \times N \times 3$) με τυχόν προϋπάρχοντα τρίγωνα
- Y είναι πίνακας διάστασης $M \times N \times 3$ που για τα σημεία του τριγώνου θα περιέχει τις αντίστοιχες χρωματικές συνιστώσες (R_i, G_i, B_i) καθώς και τα προϋπάρχοντα τρίγωνα της εισόδου X (επικαλύπτοντας τυχόν κοινά χρωματισμένα σημεία που προϋπήρχαν από άλλα τρίγωνα).

B4. Phong Shading

Να υλοποιηθεί η συνάρτηση:

$$Y = \text{shade_phong}(\text{verts}_p, \text{verts}_n, \text{verts}_c, \text{bcoords}, \text{cam_pos}, k_a, k_d, k_s, n, l_{pos}, l_{int}, l_{amb}, X)$$

η οποία υπολογίζει το χρώμα των σημείων του τριγώνου πραγματοποιώντας παρεμβολή τόσο στα κανονικά διανύσματα όσο και στα χρώματα των κορυφών. Συγκεκριμένα:

- Για το δοθέν τρίγωνο, θα υπολογίζει τα κανονικά διανύσματα των αρχικών σημείων (δηλαδή, πριν την προβολή) κατά μήκος των ενεργών πλευρών εκτελώντας γραμμική παρεμβολή στα κανονικά διανύσματα των κορυφών της πλευράς.
- Για κάθε εσωτερικό σημείο, θα υπολογίζει το κανονικό διάνυσμα κατά μήκος του scan line εκτελώντας γραμμική παρεμβολή στα κανονικά διανύσματα που αντιστοιχούν στα ενεργά σημεία της πλευράς.
- Παρόμοια διαδικασία θα πραγματοποιείται και για τα χρώματα των σημείων.
- Έχοντας υπολογίσει το κανονικό διάνυσμα και το χρώμα για ένα σημείο, το χρώμα του θα προκύπτει χρησιμοποιώντας τη συνάρτηση *light()*

Τα ορίσματα της συνάρτησης είναι όμοια με της συνάρτησης *shade_gouraud()*.

Παραδοτέα

1. Οι παραπάνω συναρτήσεις σε μορφή **σχολιασμένου** πηγαίου κώδικα python (\leq v3.10) με σχόλια γραμμένα στα **αγγλικά** ή **greeklish**. (κοινώς, **μη γράφετε σχόλια με ελληνικούς χαρακτήρες**).
2. Αναφορά με:
 - Περιγραφή της λειτουργίας και του τρόπου κλήσης των προγραμμάτων.
 - Περιγραφή των συναρτήσεων.
 - Τα ενδεικτικά αποτελέσματα που παράγονται από το demo.
3. script επίδειξης με όνομα demo.py. Το script θα πρέπει να καλείται χωρίς εξωτερικά ορίσματα, να διαβάζει το αντικείμενο από το αρχείο hw3.npy που σας δίνεται, και να παράγει ενδεικτικές φωτογραφίες του αντικειμένου. Συγκεκριμένα, για κάθε επιλογή shader (Gouraud ή Phong) να παράγει και να αποθηκεύει 4 φωτογραφίες:
 - 3 με χρήση μόνο ενός είδους φωτισμού (ambient, diffusion, specular) και μία τέταρτη με συνδυασμό όλων των προηγούμενων. Συνολικά, θα πρέπει να παραχθούν 8 φωτογραφίες, 4 για κάθε shader. Οι τιμές όλων των μεταβλητών που θα χρειαστείτε περιέχονται στο αρχείο hw3.npy. Τέλος, θεωρείστε πως οι συντεταγμένες του αντικειμένου (verts) δίνονται ως προς το WCS.
 - Ακόμα, όσον αφορά το φωτισμό, σας δίνονται τρεις (3) σημειακές πηγές φωτισμού με διαφορετική θέση και διαφορετική ένταση. Ζητείται να αποτυπώσετε μία εικόνα για κάθε πηγή φωτισμού, και μία με όλες μαζί.

Έξτρα Παραδοτέο (+ 1 μονάδα)

Στα αρχεία της εργασίας, σας δίνονται ακόμα:

- Μία εικόνα που απεικονίζει το χρώμα της υφής του αντικειμένου σε μορφή texture map.
- Στο αρχείο με τα δεδομένα της εργασίας, εκτός από τη γεωμετρία του αντικειμένου, τις παραμέτρους της υφής, και την πληροφορία φωτισμού σας δίνονται ακόμα:
 - "uvs": πίνακας $2 \times N_{verts}$ που αντιπροσωπεύει τις 2Δ συντ/νες κάθε σημείου του αντικειμένου στην εικόνα (texture map) που σας έχει δοθεί.
 - "face_uv_indices": πίνακας $3 \times N_{faces}$ που αντιστοιχίζει τις κορυφές κάθε τριγώνου σε συντ/νες uv (όπως ακριβώς οι δείκτες του πίνακα faces στον πίνακα των σημείων του αντικειμένου).

Ζητούνται οι ακόλουθες συναρτήσεις:

$$color = bilerp(uv, texture_map)$$

η οποία επιστρέφει το χρώμα που αντιστοιχεί στις δοθείσες 2Δ συντ/νες uv μέσω δι-γραμμικής παρεμβολής στην εικόνα εισόδου texture_map.

- uv : 2Δ συντ/νες 1×2 .
- texture_map: η εικόνα της υφής του αντικειμένου $M \times M \times 3$.

Για το έξτρα παραδοτέο θα πρέπει να εμπλουτίσετε τη συνάρτηση *render_object*, έτσι ώστε να παίρνει ως είσοδο τους πίνακες *uvs*, *face_uv_indices*, και την εικόνα texture_map. Θα πρέπει αντί να γίνεται παρεμβολή στο χρώμα των κορυφών του εκάστοτε τριγώνου κατά τη διάρκεια του shading, να γίνεται παρεμβολή στις συντ/νες uv των κορυφών, και να χρωματίζεται το κάθε σημείο του τριγώνου χρησιμοποιώντας το χρώμα που προκύπτει από τη συνάρτηση *bilerp*.

Παρατηρήσεις

- Οι εργασίες αξιολογούνται με χρήση Python v3.10.
- Για ανοίξετε την εικόνα texture_map, μπορείτε να χρησιμοποιήσετε οποιαδήποτε βιβλιοθήκη της αρεσκείας σας (ενδεικτικά opencv-python, pillow, imageio).
- 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.
- Η αναφορά πρέπει να είναι ένα αρχείο τύπου PDF, και να έχει όνομα report.pdf.
- Το προς υποβολή αρχείο πρέπει να περιέχει τα αρχεία κώδικα Python και το αρχείο report.pdf το οποίο θα είναι η αναφορά της εργασίας.

- Όλα τα αρχεία κώδικα πρέπει να είναι αρχεία κειμένου τύπου UTF-8, και να έχουν κατάληξη py.
- 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!