



Εργασία 3: Θέαση

Παλάσκα Αιμιλία 10453 (aimiliarm@ece.auth.gr)

Επιβλέποντες: Αναστάσιος Ντελόπουλος, Αντώνης Καρακώττας

Λειτουργία και τρόπος κλήσης των προγραμμάτων

Τα προγράμματα μπορούν να κληθούν είτε από κάποιο IDE π.χ. VSCode είτε με την εντολή τερματικού `/bin/python3 demo_g.py` από τον φάκελο που περιέχει το αρχείο `δεδομένων h3.npy`. Να σημειωθεί πως απαιτούνται μερικά λεπτά για να ολοκληρωθεί η αποθήκευση της τελικής εικόνας. Τέλος, ο κώδικας μπορεί να βρεθεί και στο [GitHub](#).

Αρχεία και συναρτήσεις

Στο πρόγραμμα συμπεριλαμβάνονται τα παρακάτω αρχεία και συναρτήσεις, ενώ περισσότερα στοιχεία για την λειτουργία τους μπορούν να βρεθούν στα σχόλια της εκάστοτε συνάρτησης στα αρχεία κώδικα:

Αρχείο `vector_interp.py`

Χρησιμοποιήθηκε το ίδιο αρχείο με την πρώτη εργασία

Αρχείο `g_shading.py`

Χρησιμοποιήθηκε το ίδιο αρχείο με την πρώτη εργασία

Αρχείο `perspective_project.py`

Χρησιμοποιήθηκε το ίδιο αρχείο με την δεύτερη εργασία

Αρχείο `rasterize.py`

Χρησιμοποιήθηκε το ίδιο αρχείο με την δεύτερη εργασία

Αρχείο `lookat.py`

Χρησιμοποιήθηκε το ίδιο αρχείο με την δεύτερη εργασία

Αρχείο `light.py`

- ❖ `light(point: np.ndarray, normal: np.ndarray, vcolor: np.ndarray, cam_pos: np.ndarray, ka: float, kd: float, ks: float, n: int, lpos: np.ndarray, lint: np.ndarray) -> np.ndarray`: υλοποίηση του μοντέλου φωτισμού σε ένα σημείο

Αρχείο `shading.py`

- ❖ `calculate_normals(verts: np.ndarray, faces: np.ndarray) -> np.ndarray`: υπολογισμός των κανονικών διανυσμάτων δοσμένων σημείων και τριγώνων που σχηματίζουν αυτά
- ❖ `shade_gouraud(vertsp: np.ndarray, vertsn: np.ndarray, vertsc: np.ndarray, bcoords: np.ndarray, cam_pos: np.ndarray, ka: float, kd: float, ks: float, n: int, lpos: np.ndarray, lint: np.ndarray, lamb: np.ndarray, X: np.ndarray) -> np.ndarray`: υλοποιεί των χρωματισμό όλων των τριγώνων με το μοντέλο σκίασης Gouraud
- ❖ `shade_phong(vertsp: np.ndarray, vertsn: np.ndarray, vertsc: np.ndarray, bcoords: np.ndarray, cam_pos: np.ndarray, ka: float, kd: float, ks: float, n: int, lpos: np.ndarray,`



lint: np.ndarray, lamb: np.ndarray, X: np.ndarray) -> np.ndarray: υλοποιεί των χρωματισμό όλων των τριγώνων με το μοντέλο σκίασης Phong

- ❖ `render_object(shader: str, focal: float, eye: np.ndarray, target: np.ndarray, up: np.ndarray, bg_color: np.ndarray, M: int, N: int, H: int, W: int, verts: np.ndarray, vert_colors: np.ndarray, faces: np.ndarray, ka: float, kd: float, ks: float, n: int, lpos: np.ndarray, lint: np.ndarray, lamb: np.ndarray)` -> np.ndarray: δημιουργία εικόνας με την απεικόνιση αντικειμένου, δοσμένων των χαρακτηριστικών της κάμερας, το είδος σκίασης, τις συντεταγμένες και τα χρώματα των ακμών

Αρχείο demo.py

Το αρχείο καλείται χωρίς ορίσματα και δεν περιέχει συναρτήσεις. Είναι υπεύθυνο για την διαχείριση των δεδομένων (loading, pre-processing) και για την κλήση της συνάρτησης `render_object` ενώ μετά την ολοκλήρωση της εκτέλεσης επιδεικνύει και αποθηκεύει την εικόνα στον φάκελο από τον οποίο κλήθηκε το πρόγραμμα.

Παραδοχές που χρησιμοποιήθηκαν

- ❖ Στην συνάρτηση φωτισμού προστίθεται σαν τελευταία στήλη του `lint` το `lamb` για διευκόλυνση
- ❖ Γίνεται η παραδοχή πως τα ορίσματα των συναρτήσεων γίνονται με τα προδιαγεγραμμένα στοιχεία και διαστάσεις, επομένως το error handling είναι μηδαμινό

Ενδεικτικά αποτελέσματα από τα demos

Σημείωση: τα αποτελέσματα δεν είναι τέλεια και οι συναρτήσεις πιθανότατα χρειάζονται κάποιες μικρές μεταβολές αλλά το γενικότερο πλαίσιο απεικόνισης της λειτουργίας του μοντέλου φωτισμού επιτυγχάνεται. Παρατίθενται τα αποτελέσματα, πρώτα για μέθοδο σκίασης Gouraud και έπειτα Phong



Πηγές

- ❖ Γλωσσικό μοντέλο [ChatGPT 3.5](#)
- ❖ Επέκταση κειμενογράφου [Code Blocks](#)
- ❖ [Visual Studio Code](#) - Community Edition
- ❖ [Python](#) - Version 3.10.12