

# Γραφική με Υπολογιστές

## -Εργασία #1-

Θωμάς Πλιάκης  
tpliakis@ece.auth.gr  
AEM: 9018

10 Απριλίου 2022

### interpolate\_color

Η συνάρτηση **vector\_interp** υλοποιεί πολύ απλά μια γραμμική παρεμβολή μεταξύ των σημείων που δίνονται ως ορίσματα με τον παρακάτω τρόπο:

$$r1 = ||x2 - x||$$

$$r2 = ||x - x1||$$

$$percent = \frac{r1}{r1 + r2}$$

$$value = percent * C1 + (1 - percent) * C2$$

Όπου  $r1$  και  $r2$  είναι οι απόστασεις των ενεργών σημείων από το σημείο της παρεμβολής,  $r1 + r2$  είναι η απόσταση των σημείων  $x1, x2$ . Με την διαίρεση γίνεται μια κανονικοποίηση για να έχουμε το ποσοστό μεταξύ 0 και 1 που πρέπει να συμμετέχει κάθε σημείο στον χρωματισμό.

Ο κώδικας, λόγω της βιβλιοθήκης **numpy** μπορεί να υλοποιεί την γραμμική παρεμβολή με μεταβλητές οποιονδήποτε διαστάσεων.

### shade\_triangle

Επειδή η υλοποίηση των δύο **mode** ,["flat" , "gouraud"], ξεκίνησε διαφορετικά, η ενοποίηση των 2 διαδικασιών έγινε με ένα **if ... else** ανάλογα το όρισμα **shade\_t**.

Η συνάρτηση **shade\_triangle** με όρισμα **shade\_t = "flat"** αποδίδει ένα χρώμα σε κάθε τρίγωνο, τον μέσο όρο των χρωμάτων των κορυφών του. Τα βήματα του αλγορίθμου περιγράφονται παρακάτω:

1. Υπολογισμός και αποθήκευσή του μέσου όρου των χρωμάτων στον  $1 \times 3$  πίνακα **c**.
2. Υπολογισμός των ελάχιστων και μεγίστων συντεταγμένων της κάθε πλευράς, καθώς και της κλίσης τους, **Ykmin,Xkmin,Ykmax,Xkmax**.
3. Εύρεση των ολικών μεγίστων και ελάχιστων συντεταγμένων του τριγώνου, **Ymin,Xmin,Ymax,Ymin**.
4. Βρίσκουμε ποιες είναι οι αρχικές ενεργές πλευρές(παίρνουν το λογικό 1) και αποθηκεύουμε σε πίνακα τις συντεταγμένες τους, που αποτελούν και τα ενεργά οριακά τους σημεία **ActiveSides,Xk**. Στον πίνακα **Xact** αποθηκεύονται οι τετμημένες των 2 ενεργών σημείων μόνο.
5. Μπαίνουμε στην κυρίως επανάληψη στην οποία:
  - Ελέγχουμε αν βρισκόμαστε σε κορυφή του τριγώνου.
  - Υλοποιείται το σκανάρισμα σε όλες τις τετμημένες για συγκεκριμένη τεταγμένη κατά την οποία αν κάποιο **pixel** βρίσκεται μέσα ή πάνω στο τρίγωνο χρωματίζεται.
  - Ενημερώνεται η λίστα των ενεργών πλευρών.
  - Τέλος ενημερώνεται η λίστα με τα νέα ενεργά σημεία ανάλογα με την κλίση της κάθε πλευράς του τριγώνου.

Η συνάρτηση **shade\_triangle** με όρισμα **shade\_t = "gouraud"** εκτελεί τις ίδιες αρχικοποιήσεις με προηγουμένως αλλά και κάποιες ακόμα. Τα επιπλέον βήματα του αλγορίθμου που γίνονται είναι:

1. Αποθηκεύση των δεικτών των κορυφών κάθε ακμής στις λίστες **minindex**, **maxindex**.
2. Αποθηκεύση των δεικτών των 2 ενεργών σημείων κάθε ακμής στις λίστες **index** για προσέλαση των **colorsAct**, **Xact** πινάκων.
3. Αποθηκεύση του χρώματος των ενεργών οριακών σημείων στην λίστα **colorsAct**.
4. Όταν γίνεται ο χρωματισμός κάποιου σημείου εσωτερικά του τριγώνου, υπολογίζεται το χρώμα του από την **interpolate\_color** και τα 2 ενεργά σημεία.
5. Ο υπολογισμός του χρώματος για τα ενεργά οριακά σημεία γίνεται με χρήση της **interpolate\_color** από τις αντίστοιχες κορυφές κατά την ενημέρωση της λίστας με τα ενεργά σημεία.

Όλα τα παραπάνω γίνονται για τον χρωματισμό ενός αντικειμένου ολόκληρου που αποτελείται από πολλά τρίγωνα. Αυτό γίνεται στην συνάρτηση:

## render

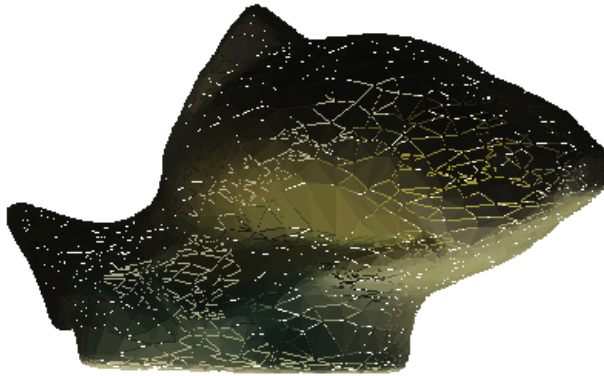
Ο κώδικας στην συνάρτηση **render** περιγράφεται στα παρακάτω βήματα:

1. Γίνεται υπολογισμός του βάθους του κάθε τριγώνου, **Dm**.
2. Ταξινομώ τον πίνακα αυτό κατά φθίνουσα σειρά. Με την ίδια σειρά ταξινομείται και ο πίνακας **faces**, με την βοήθεια του **D\_order** που περιέχει την σειρά που ταξινομήθηκε ο **Dm**.
3. Ανάλογα με ποιο από τα 2 **mode** θέλουμε να χρησιμοποιήσουμε, επιλέγουμε την κατάλληλη τιμή για την μεταβλητή **shade\_t** ,["flat" , "gouraud"].

## Σχόλια

- Τέλος δεν έγινε καμία άλλη παραδοχή πέρα από αυτές της εκφώνησης και των σημειώσεων που περιγράφουν των αλγόριθμο που κληθήκαμε να υλοποιήσουμε.
- Η ποιότητα των εικόνων δείχνει ότι κάποιο λάθος υπάρχει στον κώδικα που δεν μπόρεσα να διορθώσω, αφού εμφανίζονται διάφορα κενά. Υποθέτω ότι δεν πρέπει να χρωματίζονται **pixels** στο περίγραμμα κάποιων τριγώνων.
- Ο χρόνος για να παραχθεί η κάθε εικόνα είναι περίπου 7 δευτερολεπτά, το οποίο μάλλον είναι αρκετός χρόνος αφού ο κώδικας δεν παράγει ακριβώς τον σωστό αποτέλεσμα.

Αποτελέσματα της `shade_t = "flat"`:



Αποτέλεσμα της `shade_t = "gouraud"`:

