

Ειδικά Θέματα Υπολογιστικής Όρασης και Γραφικών

Εργασία Γραφικών 2014-2015

Ανδρέας Καναβός - (kanavos@ceid...)

Γενική Περιγραφή

Στόχος της εργασίας είναι η εξοικείωση με την βιβλιοθήκη OpenGL/freeGlut. Πιο συγκεκριμένα η σύνθεση μίας animated σκηνής με χρήση των βασικών στοιχείων της OpenGL (αντικείμενα, χρωματισμοί, φωτισμοί, σκιάσεις, ανακλάσεις, υφές, κλπ) και στην οποία ο χρήστης θα παρεμβαίνει.

Αναλυτική Περιγραφή

Σκοπός είναι η υλοποίηση ενός αλληλεπιδραστικού viewer μιας 3D κινούμενης σκηνής. Η σκηνή είναι ένας κύβος/κουτί 2x2x2 κεντραρισμένος στην αρχή των αξόνων, ο οποίος περιέχει μία ή περισσότερες σφαίρες που αναπηδούν μέσα σε αυτόν. Ο παρατηρητής τοποθετείται στην θέση (0,0,-3) (θέση κάμερας/ματιού) και παρατηρεί με κατεύθυνση ίδια με τον θετικό Z-άξονα. Η γωνία παρατήρησης είναι 65 μοίρες. Το πρόγραμμα πρέπει να δείχνει την κινούμενη εικόνα και να επιτρέπει στον χρήστη να αλλάζει την θέση του χρησιμοποιώντας το πληκτρολόγιο.

Φάσεις Υλοποίησης

Ξεκινήστε την ανάπτυξη απεικονίζοντας το κουτί και διαδοχικά υλοποιήστε χαρακτηριστικά όπως resize, quit κλπ. Στην συνέχεια, απεικονίστε μία σφαίρα και τους τρόπους φωτισμού και τέλος υλοποιήστε πολλαπλές σφαίρες και συγκρούσεις.

1^ο μέρος

Γραφικά Στοιχεία:

1. Το **κουτί** είναι $2 \times 2 \times 2$ μονάδες σε μέγεθος και είναι κεντραρισμένο στο $(0,0,0)$. Πρέπει να απεικονίσετε 5 πλευρές:
 - a. πίσω μέρος ($z=1$),
 - b. αριστερό μέρος ($x=-1$),
 - c. δεξί μέρος ($x=1$),
 - d. κάτω μέρος ($y=-1$) και
 - e. πάνω μέρος ($y=1$).Μπορείτε να απεικονίσετε τις πλευρές σαν quads, αλλά μία πιο ψηφιδωτή υλοποίηση θα έχει καλύτερα οπτικά αποτελέσματα. Κάθε πλευρά θα έχει και διαφορετικό χρώμα.
2. Οι **σφαίρες** έχουν ακτίνα $\rho=0.05$. Χρησιμοποιείτε την συνάρτηση `glutSolidSphere` για την απεικόνισή τους. Όταν προστίθεται μία σφαίρα στην σκηνή, επιλέγεται με τυχαίο τρόπο η θέση, η ταχύτητα (διάνυσμα) καθώς και το χρώμα. Η αρχική ταχύτητα (μέτρο) θα είναι από 0.5 μέχρι 2. Χρησιμοποιείτε `smooth shading` στην απεικόνιση των σφαιρών.
3. Ο **φωτισμός** θα υλοποιηθεί με διαφορετικούς τρόπους: `ambient only`, `directional light`, `positional light` και `spot light`. Το **πλήκτρο "L"** θα εναλλάσσει τους διαφορετικούς τρόπους. Το **πλήκτρο "P"** θα εναλλάσσει τη θέση των πηγών φωτός (στο κέντρο, σε μία από τις ακμές, σε μία από τις γωνίες, κλπ).

Διεπαφή Χρήστη

1. **L**: εναλλαγή στον τρόπο φωτισμού
2. **P**: εναλλαγή στον τόπο φωτισμού
3. **+**: προσθήκη σφαίρας
4. **-**: προσθήκη σφαίρας (αφαίρεση σφαίρας) - FIFO λογική
5. **S**: ενεργοποίησης/απενεργοποίηση σκιάσεων
6. **q**: τερματισμός εφαρμογής
7. **F**: Το σημείο παρατήρησης κινείται προς την αρχή των αξόνων
8. **B**: Το σημείο παρατήρησης απομακρύνεται από την αρχή των αξόνων
9. **E**: Η κατεύθυνση παρατήρησης κινείται προς τα πάνω
10. **D**: Η κατεύθυνση παρατήρησης κινείται προς τα κάτω
11. **S**: Η κατεύθυνση παρατήρησης κινείται προς τα αριστερά
12. **F**: Η κατεύθυνση παρατήρησης κινείται προς τα δεξιά
13. **T**: Τοποθέτηση υφής πάνω στις σφαίρες (ή επιστροφή στο χρώμα)

2^ο μέρος

Αναπαράσταση Σφαίρας

Η θέση μίας σφαίρας μπορεί να αναπαρασταθεί με 3 στοιχεία: την αρχική τιμή χρόνου t_0 , την αρχική θέση q και το διάνυσμα της ταχύτητας v . Δεδομένων αυτών των στοιχείων, η θέση της σφαίρας μπορεί να δοθεί ως $p(t) = q + (t-t_0)v$ για $t \geq t_0$. Επίσης πρέπει να καταγράφετε το χρώμα της και το χρόνο της επόμενης σύγκρουσης (για να προγραμματίσετε τα γεγονότα!!).

Ουρά Γεγονότων

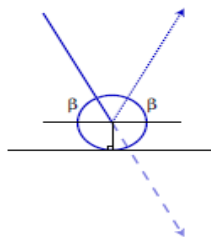
Για να χειριστείτε την προσομοίωση πρέπει να χρησιμοποιήσετε μία ουρά προτεραιότητας (ή κάποια άλλη δομή) για τα γεγονότα (διατεταγμένα ως προς το χρόνο). Υπάρχουν δύο τύποι γεγονότων: συγκρούσεις σφαιρών μεταξύ τους και συγκρούσεις σφαιρών με τα τοιχώματα. Μία σφαίρα κινείται μέχρι ένα από τα δύο γεγονότα να συμβεί. Συνεπώς ο χρόνος που πρέπει να ενημερώνεται η κατάσταση του συστήματος είναι όταν ένα συμβάν λάβει μέρος. Επειδή η κίνηση των σφαιρών είναι απλή, μπορεί να προβλεφθεί το επόμενο γεγονός.

Αν είναι να απεικονίζουμε την σκηνή στον χρόνο t , τότε πρέπει να επεξεργαστούν όλα τα γεγονότα μέχρι την χρονική στιγμή αυτή. Η επεξεργασία σύγκρουσης με τοίχο γίνεται με τον υπολογισμό ενός νέου διανύσματος ταχύτητας v' , την αρχικοποίηση του αρχικού του χρόνου (t_0) στο χρόνο σύγκρουσης και την αρχική θέση (q) στη θέση σύγκρουσης. Στη συνέχεια, υπολογίζεται η επόμενη σύγκρουση είτε με τους τοίχους είτε με τις άλλες σφαίρες. Αν μία σφαίρα συγκρουστεί με μία άλλη τότε μπορεί να προκαλούνται και νέες συγκρούσεις.

Συγκρούσεις με τον τοίχο

Έστω σφαίρα με θέση $p(t) = q + t \cdot v$. Ο χρόνος σύγκρουσης μπορεί να ορισθεί από την προβολή της συνιστώσας του διανύσματος της ταχύτητας v , π.χ. αν $v_x > 0$ τότε η σφαίρα θα χτυπήσει το δεξί μέρος του κουτιού ($x=1$) σε χρόνο t σύμφωνα με την $(1-p) = p(t)_x = q_x + t \cdot v_x$. Άρα ο χρόνος της σύγκρουσης με τον τοίχο είναι $t = (1 - q_x) / v_x$. Αν $v_x < 0$ κάνουμε το αντίστοιχο για την αριστερή πλευρά. Γενικά γίνεται η ίδια διαδικασία για όλες τις πλευρές και σαν γεγονός σύγκρουσης λαμβάνεται ο μικρότερος χρόνος.

Όταν η σφαίρα συγκρουστεί με τον τοίχο το νέο διάνυσμα ταχύτητας είναι η ανάκλαση του αρχικού.

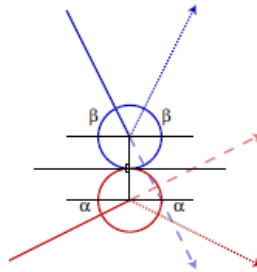


3^ο μέρος

Συγκρούσεις με άλλες σφαίρες

Έστω δύο σφαίρες $p_1(t) = q_1 + tv_1$ και $p_2(t) = q_2 + tv_2$, τότε η απόσταση των κέντρων τους δίνεται από την $d(t) = \sqrt{(p_1(t) - p_2(t))^2}$. Όταν το $d(t) \leq 2r$, τότε έχουμε σύγκρουση και μπορούμε να βρούμε το σημείο της σύγκρουσης από τις ρίζες της εξίσωσης. Αν δεν υπάρχουν πραγματικές ρίζες, τότε δεν υπάρχει σύγκρουση.

Όταν δύο σφαίρες συγκρούονται ανακλώνται ως προς την εφαπτομένη σύγκρουσης



Επέκταση

Θα πρέπει να δημιουργήσετε έναν programmable shader για τον φωτισμό. Το πρόγραμμα φωτισμού θα πρέπει να υλοποιεί τους ίδιους τρόπους φωτισμού όπως η fixed pipeline της βιβλιοθήκης αλλά με σκίαση per pixel. Η μεταβλητή `gl_LightSource[]` διατηρεί την κατάσταση του φωτισμού που θα πρέπει το πρόγραμμά σας να χρησιμοποιεί για να καθορίζει τον τρόπο φωτισμού. Το πρόγραμμα φωτισμού θα πρέπει να υποστηρίζει ambient και diffuse lighting. Τέλος αντί για χρώμα επεκτείνετε την βασική υλοποίηση ώστε να χρησιμοποιείτε στις σφαίρες κάποια υφή.

Αναφορές - Υλικό

- Freeglut <http://freeglut.sourceforge.net/>
- GLTools Library from OpenGL® SuperBible <http://www.starstonesoftware.com/OpenGL/> and <http://www.starstonesoftware.com/files/SB5.zip>
- SuperBible: Comprehensive Tutorial and Reference, Fourth Edition Richard S. Wright Jr.; Benjamin Lipchak; Nicholas Haemel, Addison-Wesley Professional, 2009
- Dave Shreiner. 2009. *OpenGL Programming Guide: The Official Guide to Learning OpenGL, Versions 3.0 and 3.1* (7th ed.). Addison-Wesley Professional
- <http://www.khronos.org/files/opengl-quick-reference-card.pdf>