Πανεπιστήμιο Πατρών

Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών

3Δ Υπολογιστική Γεωμετρία και Όραση

UAV Collision Detection and Path Planning

Τσάμπρας Κωνσταντίνος

up1083865

Περίληψη

Η εργασία αυτή έχει ως σκοπό την απεικόνηση ενός αριθμού UAVs στον τρισδιάστατο χώρο, την δημιουργία περιβαλλόντων όγκων για τα αυτά, τον εντοπισμό συγκρούσεων σε κάθε στιγμή, αλλά και την μοντελοποίηση της κίνησης των UAVs, τον εντοπισμό συγκρούσεων σε χρονικά διαστήματα, και τέλος, την δημιουργία πρωτοκόλλων κίνησης των UAVs με στόχο την αποφυγή συγκρούσεων και την επίτευξη ενός στόχου (προσγείωση/απογείωση).

Προσέγγιση

Για την επίτευξη των παραπάνω γίνεται χρήση της γλώσσας Python, της βιβλιοθήκης Open3D, καθώς και πληθώρας άλλων βοηθητικών βιβλιοθηκών.

Η μοντελοποίηση του περιβάλλοντος, των UAVs και των περιβαλλόντων όγκων γίνεται με την χρήση αντικειμενοστραφούς προγραμματισμού, με κλάσεις όπως:

* Airspace (Ο εναέριος χώρος, υπεύθυνος για την δημιουργία του περιβάλλοντος, των UAVs, τον ορισμό των πρωτοκόλλων και την αποφυγή συγκρούσεων)
* LandingPad (Ο χώρος προσγείωσης, δεν έχει ιδιαίτερες λειτουργίες)
* UAV (Η κλάση που υλοποιεί τα UAVs, υπεύθυνη για την δημιουργία του κάθε UAV, την εμφάνιση και μετακίνησή του, την δημιουργία περιβάλλοντων όγκων, τον εντοπισμό στατικών/χρονικών συγκρούσεων με άλλα UAVs)

Καθώς και με έναν αριθμό βοηθητικών κλάσεων για την περιγραφή περιβαλλόντων όγκων και, γενικότερα, σχημάτων στον τρισδιάστατο χώρο:

* Point3D, Line3D, Cuboid3D, Sphere3D, Mesh3D (Χρησιμοποιούνται για την εμφάνιση απλών έως και σύνθετων αντικειμένων στον τρισδιάστατο χώρο)
* Triangle3D (Μια κλάση που κληρονομεί από την Mesh3D και διευκολύνει τον χειρισμό απλών τριγώνων στον χώρο)
* ConvexPolygon3D (Επίσης μια κλάση που κληρονομεί από την Mesh3D με σκοπό την αναπαράσταση κυρτών πολυγώνων στον τρισδιάστατο χώρο)
* Polyhedron3D (Μια χρήσιμη κλάση για αναπαράσταση περιβαλλόντων όγκων -όπως το kdop- που αποτελείται από έναν αριθμό κυρτών πολυγώνων)
* AabbNode (Μια κλάση η οποία δημιουργεί ένα δέντρο με σκοπό την εύρεση συγκρούσεων, ακολουθώντας την λογική του «διαίρει και βασίλευε»)
* Kdop (Η κλάση η οποία αναπαριστά το Kdop ενός UAV, κληρονομεί από την κλάση Polyhedron3D)

Ταυτόχρονα με την επεξήγηση της επίλυσης του κάθε ερωτήματος, οι κύριες μέθοδοι και λειτουργίες των παραπάνω κλάσεων θα αναφέρονται με περισσότερη λεπτομέρεια.

Ερωτήματα

1. «Επιλέξτε κάποια 3D μοντέλα drone και οπτικοποιήστε τα σε τυχαίες

θέσεις. Επίσης μοντελοποιήστε μία επιφάνεια προσγείωσης

απογείωσης με NxN θέσεις drones»

Για το ερώτημα αυτό, καθώς και για τα επόμενα, συλλέχθηκαν διάφορα μοντέλα UAVs, και γενικότερα ιπτάμενων οχημάτων. Αυτά είναι τα:

- v22\_osprey

- twin\_copter

- F52

- Helicopter

Με αριθμό vertices από 4.000 έως 50.000. Επιπλέον υπάρχει και το μοντέλο quadcopter\_scifi το οποίο έχει 200.000 vertices, αλλά δεν χρησιμοποιείται στις περισσότερες περιπτώσεις λόγω της καθυστέρησης που προκαλεί ένα τόσο μεγάλο αντικείμενο όταν μετακινείται στην σκηνή της open3d (ειδικά στα ερωτήματα Ζ και έπειτα). Παρόλα αυτά μπορεί να χρησιμοποιηθεί κάνοντας uncomment το όνομά του στον ορισμό των μοντέλων.

Το πρώτο ερώτημα δεν είναι σύνθετο, οπότε θα αναφερθούν οι μέθοδοι της κλάσης Airspace υπεύθυνες για την δημιουργία του περιβάλλοντος.

Οι μέθοδοι αυτοί είναι:

1. create\_uavs(): Δημιουργεί στοιχισμένα NxN UAVs πάνω στην επιφάνεια προσγείωσης/απογείωσης. Χρήσιμη μέθοδος για την επόπτευση των περιβαλλόντων όγκων των μοντέλων.
2. create\_colliding\_uavs(): Δημιουργεί 3 UAVs τα οποία συγκρούονται. Χρήσιμη μέθοδος για οπτικοποίηση των μεθόδων ανίχνευσης σύγκρουσης.
3. create\_random \_uavs(): Δημιουργεί UAVs NxN με τυχαία θέση και τυχαίο προσανατολισμό πάνω σε ένα επίπεδο. Χρησιμοποιείται για την ανίχνευση συγκρούσεων σε μια τυχαία τοποθέτηση UAVs.
4. create\_random\_uavs\_non\_colliding(): Δημιουργεί τυχαία NxN UAVs πάνω σε ένα επίπεδο, με κριτήριο να μην συγκρούονται. Χρησιμοποιείται για την αρχικοποίηση καταστάσεων όπου θέλουμε να επιδείξουμε το πρωτόκολλο αποφυγής συγκρούσεων.
5. create\_time\_colliding\_uavs(): Δημιουργεί 2 UAVs με θέσεις και ταχύτητες τέτοιες ώστε να συγκρούονται στο επόμενο χρονικό διάστημα. Χρήσιμο για την δοκιμή ανίχνευσης συγκρούσεων σε ένα χρονικό διάστημα.
6. create\_taking\_off\_uavs(): Δημιουργεί NxN UAVs πάνω στην επιφάνεια προσγείωσης με τυχαίο προσανατολισμό. Αυτή η αρχικοποίηση χρησιμοποιείται όταν ορίζουμε το πρωτόκολλο target\_beacon (Ερώτημα Η. επιλογή 1.). Παράλληλα, δημιουργεί και NxN στόχους πάνω στο dome, έναν για κάθε UAV.
7. create\_landing\_uavs (): Δημιουργεί NxN UAVs πάνω σε σφαίρα (dome) με την δοθείσα ακτίνα γύρω από την περιοχή προσγείωσης/ απογείωσης. Τα UAVs βρίσκονται στο πάνω ημισφαίριο, και όχι πολύ κοντά στο έδαφος. Αυτή η αρχικοποίηση χρησιμοποιείται όταν ορίσουμε το πρωτόκολλο landing (Ερώτημα Η. επιλογή 2.).
8. create\_landing\_uavs\_time(): Δημιουργεί UAVs εντός του dome με τυχαία θέση και προσανατολισμό. Χρησιμοποιείται όταν θέσουμε το πρωτόκολλο landing\_time (Ερώτημα Η επιλογή 3.). Τα UAVs δεν δημιουργούνται όλα αρχικά (όπως στις προηγούμενες μεθόδους): ένας αριθμός δημιουργείται στην εκκίνηση της προσομοίωσης και στην συνέχεια, σε κάθε frame γίνεται μια «κλήρωση» για το αν θα δημιουργηθεί ένα νέο UAV. Η πιθανότητες αυτής της κλήρωσης ορίζονται στο όρισμα flow (flow = 1 σημαίνει ότι σε κάθε frame θα προστίθεται ένα νέο UAV στον εναέριο χώρο). Η δημιουργία του UAV γίνεται με την create\_new\_landing\_uav().

Παρακάτω φαίνεται η εκτέλεση της μεθόδου create\_random\_uavs() για το ερώτημα 1:

Εικόνα που περιέχει γραφιστική, ζωγραφιά, γραφικά, εικονογράφηση

Περιγραφή που δημιουργήθηκε αυτόματα

Εικόνα 1. Airspace.create\_random\_uavs()

1. «Υπολογίστε το κυρτό περίβλημα, ΑΑΒΒ, και k-DOP»

Αρχικά τα UAVs τοποθετούνται στοιχισμένα στις θέσεις απογείωσης.

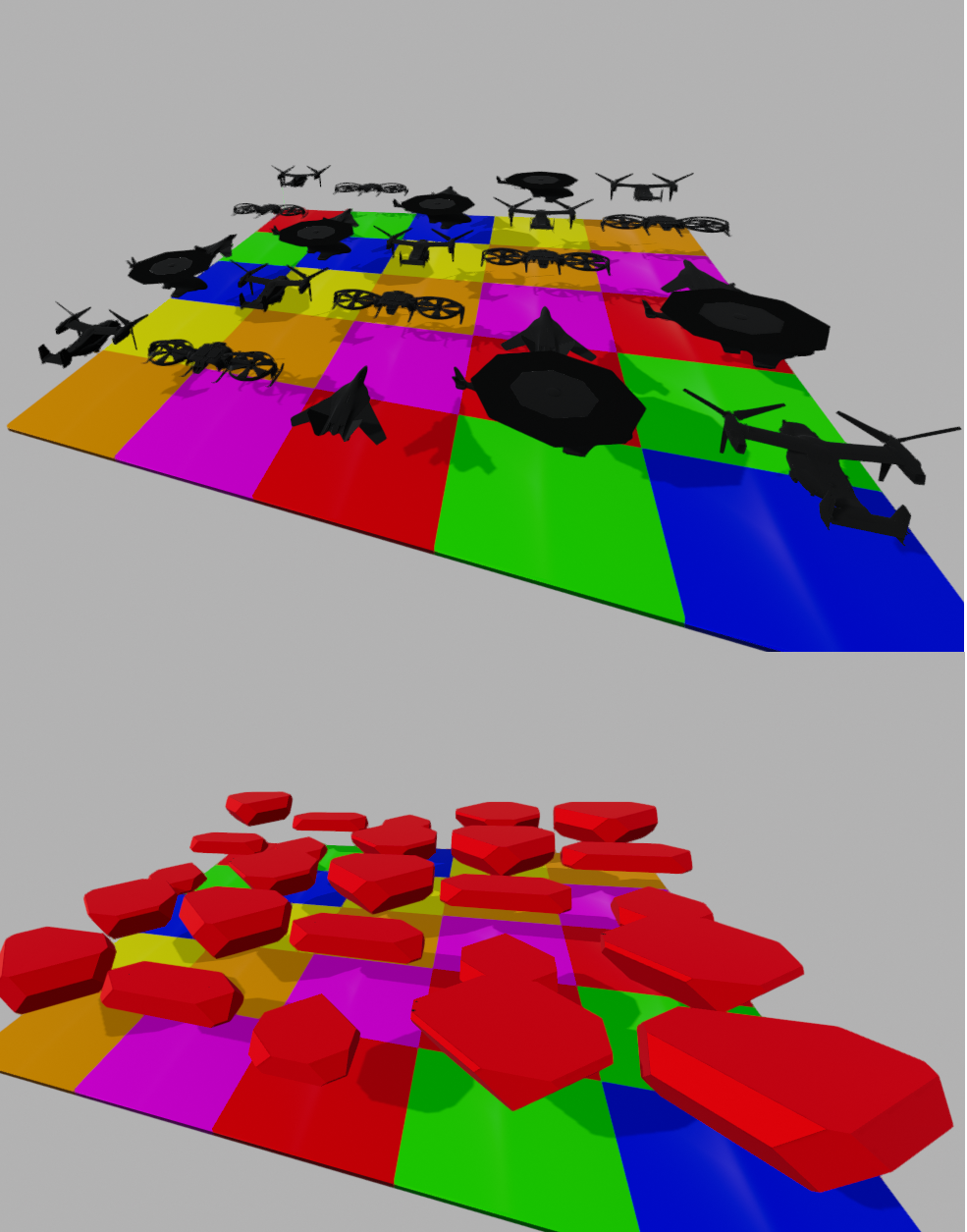
Υπάρχουν οι εξής δυνατότητες για το κάθε UAV που ενεργοποιούνται με την χρήση των παρακάτω πλήκτρων:

* K. Εμφάνιση kdop (το k ορίζεται σύμφωνα με την μεταβλητή kdop\_number, έχουν υλοποιηθεί οι περιπτώσεις για k=6 και k=14, default k = 14).
* C. Εμφάνιση κυρτού περιβλήματος.
* U. Εμφάνιση μοναδιαίας σφαίρας γύρω από το UAV.
* B. Εμφάνιση axis aligned bounding box.

(Ο «περιβάλλων όγκος» Aabb node έχει υλοποιηθεί αλλά δεν υπάρχει κάποιος καλός τρόπος για να οπτικοποιηθεί)

Ακολουθούν παραδείγματα για τον κάθε όγκο, καθώς και η διαδικασία υπολογισμού του:

1. Kdop:



Εικόνα 2. Δημιουργία 14Dops

Η περίπτωση των 6Dops είναι απλή καθώς αφορά την ουσιαστικά την εύρεση του Aabb που θα εξεταστεί παρακάτω.

Η περίπτωση του 14Dop είναι αρκετά πιο σύνθετη.

Η μέθοδος που υλοποιήθηκε έχει πολύ καλά αποτελέσματα σε UAV με φυσιολογικές αναλογίες (όχι πολύ επίπεδο ή λεπτά).

Η δημιουργία του 14Dop γίνεται στην μέθοδο της κλάσης Kdop create\_14dop(). Ακολουθεί την λογική της «αφαίρεσης» των γωνιών ενός Aabb. Τα βήματα που ακολουθούνται περιγράφονται παρακάτω:

1. Ορισμός χρήσιμων συναρτήσεων:

- find\_intersection(plane1, plane2, plane2). Βρίσκει το σημείο τομής των 3 επιπέδων.

- check\_if\_neighbor(corner\_dir, face\_dir). Ελέγχει εάν δύο κατευθύνσεις (γωνιακή και πλευράς) «συνορεύουν». Για παράδειγμα η πλευρά της οποία η κατεύθυνση είναι η [1, 0, 0], είναι γείτονας της γωνίας [1, 1, 1].

- plane\_equation\_from\_point\_normal(point, normal). Εξάγει την εξίσωση του επιπέδου. Δοθέντος ενός σημείου του και του normal του επιπέδου.

1. Εύρεση ενός vertex για κάθε κατεύθυνση, του οποίου η προβολή πάνω στην κατεύθυνση είναι η μέγιστη για αυτήν την κατεύθυνση. Δηλαδή βρίσκουμε το σημείο που ορίζει το επίπεδο που περιορίζει το kdop προς κάθε κατεύθυνση (ένα επίπεδο για κάθε κατεύθυνση).
2. Στην συνέχεια, για κάθε κατεύθυνση γωνίας (πχ [1,-1,1]) βρίσκουμε την εξίσωση του επιπέδου της και τις γειτονικές της κατευθύνσεις που αντιστοιχούν σε πλευρές. Για κάθε τέτοια γειτονική κατεύθυνση, βρίσκουμε