

*Αναγνώριση Προτύπων*

*Ανάλυση Εικόνας*

**Χρήση ροπών στην Ταξινόμηση**

**Δισδιάστατων Εικόνων**

# Εισαγωγή

Η παρούσα εργασία υλοποιεί την ταξινόμηση δυσδιάστατων ασπρόμαυρων εικόνων που περιέχουν 1 από τις 3 δεδομένες μορφές αεροσκαφών-κλάσεων, έχοντας υποστεί τυχαία αλλαγή μεγέθους, αρχικής θέσης και περιστροφής, στην κατάλληλη κλάση.

Παρακάτω φαίνονται οι 3 βασικές **κλάσεις**:

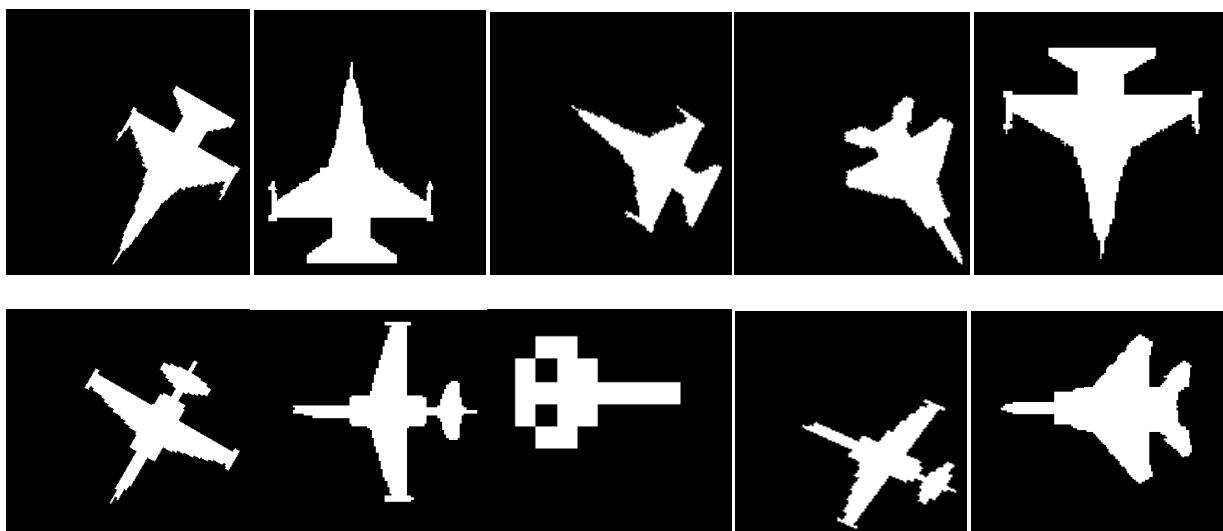


Κλάση 1

Κλάση 2

Κλάση 3

Παρακάτω φαίνεται κάποιες **τυχαίες μορφές**:



## Διάνυσμα Χαρακτηριστικών – Ροπές Hue

Για να επιτευχθεί μια επιτυχής ταξινόμηση, ως διάνυσμα χαρακτηριστικών χρησιμοποιήθηκαν οι 3 πρώτες **ροπές Hue** ( $\varphi_1, \varphi_2, \varphi_3$ ), διότι παραμένουν σχεδόν αμετάβλητες στις τυχαίες αλλαγές που αναφέρθηκαν παραπάνω.

Παρακάτω δίνονται οι μαθηματικοί τύποι για τον υπολογισμό των ροπών Hue:

$$\varphi_1 = n_{20} + n_{02}$$

$$\varphi_2 = (n_{20} - n_{02})^2 + 4n_{11}^2$$

$$\varphi_3 = (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2$$

Όπου

$$n_{ij} = \frac{\mu_{ij}}{m_{00}^{\frac{i+j}{2} + 1}} \quad \text{Κεντρική Κανονικοποιημένη Ροπή}$$

$$\mu_{ij} = \sum_x \sum_y (x - x_c)^i (y - y_c)^j I(x, y) \quad \text{Κεντρική Ροπή}$$

$$m_{ij} = \sum_x \sum_y x^i y^j I(x, y) \quad \text{Ροπή}$$

$$x_c = \frac{m_{10}}{m_{00}}, \quad y_c = \frac{m_{01}}{m_{00}} \quad \text{Κεντροειδή περιοχή}$$

$$I(x, y) \quad \text{Τιμή pixel στη θέση } (x, y)$$

Αναπτύχθηκαν 2 είδη ταξινομητών και συστήματα εκμάθησης αυτών:

- Βάσει της ελάχιστης καρτεσιανής απόστασης (*Euclidean Classifier*)
- Βάσει των 3 πλησιέστερων γειτόνων (*NN Classifier*)

# Σύστημα Ταξινόμησης

## Βάσει της ελάχιστης καρτεσιανής απόστασης

Επιλέγεται η κλάση, της οποίας η μέση τιμή απέχει την ελάχιστη ευκλείδεια απόσταση από το διάνυσμα χαρακτηριστικών της υπό-εξέταση μορφής.

## Βάσει των 3 πλησιέστερων γειτόνων

Επιλέγεται η κλάση, στην οποία ανήκουν οι περισσότεροι από τους 3 πλησιέστερους γείτονες στο διάνυσμα χαρακτηριστικών της υπό-εξέταση μορφής, βάσει του τετραγώνου της ευκλείδειας απόστασης.

\*\* Στη υλοποίηση χρησιμοποιήθηκαν ως υπο-ρουτίνες οι *Matlab* συναρτήσεις *euclidean\_classifier.m* και *k\_nn\_classifier.m* που παρέχονται από το σύγγραμμα του κ. Θεοδωρίδη στο <http://www.elsevierdirect.com/v2/companion.jsp?ISBN=9780123744869>.

# Σύστημα Εκμάθησης

Το σύνολο εκπαίδευσης βρίσκεται στο φάκελο */samples/* και περιέχει 15 αναγνωρισμένες εικόνες αεροσκαφών, 5 για κάθε κλάση.

## Βάσει της ελάχιστης καρτεσιανής απόστασης

Υπολογίζεται η μέση τιμή της κάθε κλάσης από το σύνολο εκπαίδευσης, έτσι ώστε να χρησιμοποιηθεί αργότερα από τον ΕΥ-ταξινομητή για να βρεθεί η κοντινότερη μέση τιμή.

## Βάσει των 3 πλησιέστερων γειτόνων

Υπολογίζεται τα διανύσματα χαρακτηριστικών του συνόλου εκπαίδευσης, έτσι ώστε να χρησιμοποιηθούν αργότερα από τον ΝΝ-ταξινομητή για να βρεθούν οι πλησιέστεροι γείτονες.

# Δομή Κώδικα

## /hue\_moments/

Υπολογισμός των ροπών Hue.

- ***HueMoments.m***

Παίρνει ως είσοδο μια ασπρόμαυρη εικόνα και επιστρέφει τις 7 πρώτες ροπές Hue.

- ***TestHue.m***

Test script για επαλήθευση των αποτελεσμάτων της συνάρτησης *HueMoments.m*. Τα αποτελέσματα καταγράφονται στο κεντρικό φάκελο στο αρχείο κειμένου *moments.txt*.

## /euclidean\_classifier/

Σύστημα ταξινόμησης βάσει της ελάχιστης καρτεσιανής απόστασης.

- ***EuclideanClassifier.m***

Παίρνει ως είσοδο μια ασπρόμαυρη εικόνα και τις μέσες τιμές των κλάσεων και επιστρέφει την κλάση που ταξινομήθηκε.

- ***EuclideanTraining.m***

Υπολογίζει τις μέσες τιμές των διανυσμάτων χαρακτηριστικών του συνόλου εκπαίδευσης.

- ***ClassifyRandoms\_Euclidean.m***

Ταξινομεί 10 τυχαίες εικόνες και επιστρέφει το συνολικό αριθμό λανθασμένων απαντήσεων.

- ***TestEuclidean.m***

Test script για την ταξινόμηση του συνόλου εκπαίδευσης.

## **/nn\_classifier /**

Σύστημα ταξινόμησης βάσει των 3 πλησιέστερων γειτόνων.

- ***NNClassifier.m***

Παίρνει ως είσοδο μια ασπρόμαυρη εικόνα και το σύνολο των χαρακτηριστικών του συνόλου εκπαίδευσης και επιστρέφει την κλάση που ταξινομήθηκε.

- ***NNTraining.m***

Υπολογίζει τα διανύσματα των χαρακτηριστικών του συνόλου εκπαίδευσης.

- ***ClassifyRandoms\_NN.m***

Ταξινομεί 10 τυχαίες εικόνες και επιστρέφει το συνολικό αριθμό λανθασμένων απαντήσεων.

- ***TestNN.m***

Test script για την ταξινόμηση του συνόλου εκπαίδευσης.

## **/evaluation/**

Αξιολόγηση των δύο συστημάτων ταξινόμησης.

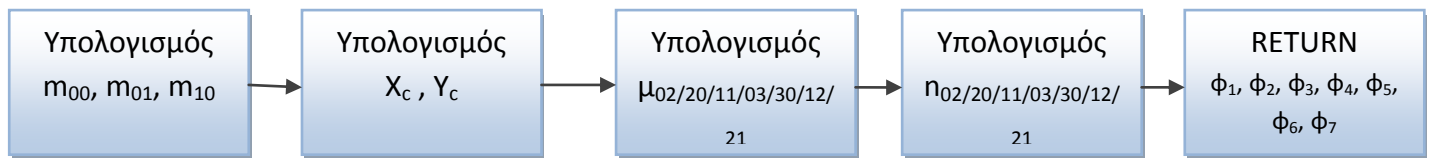
- ***Evaluator.m***

Παίρνει ως είσοδο τον αριθμό επαναλήψεων  $N$  της ταξινόμησης 10 τυχαίων εικόνων και από τους 2 ταξινομητές (συνολικά ταξινομούνται  $10N$  τυχαίες εικόνες).

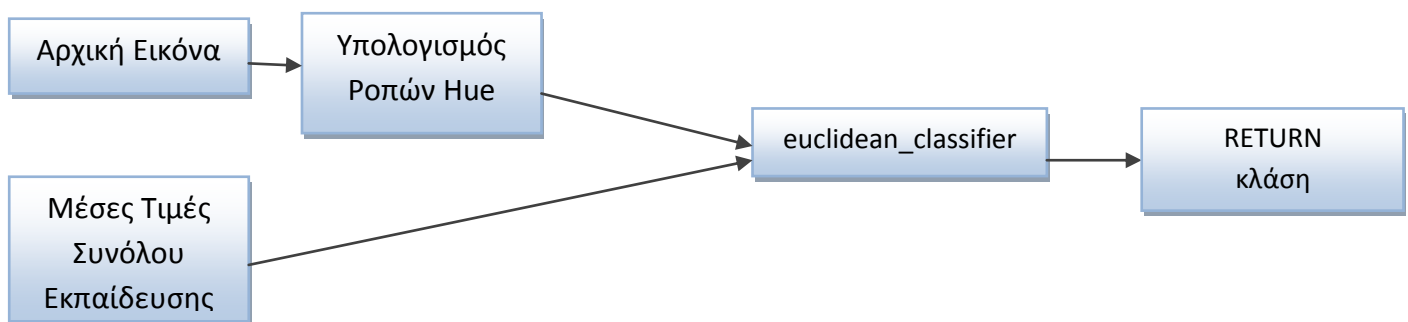
Στο τέλος εμφανίζεται διάγραμμα-πίτα για την αξιολόγηση των συστημάτων σύμφωνα με το ποσοστό αποτυχίας.

# Διαγράμματα Ροής

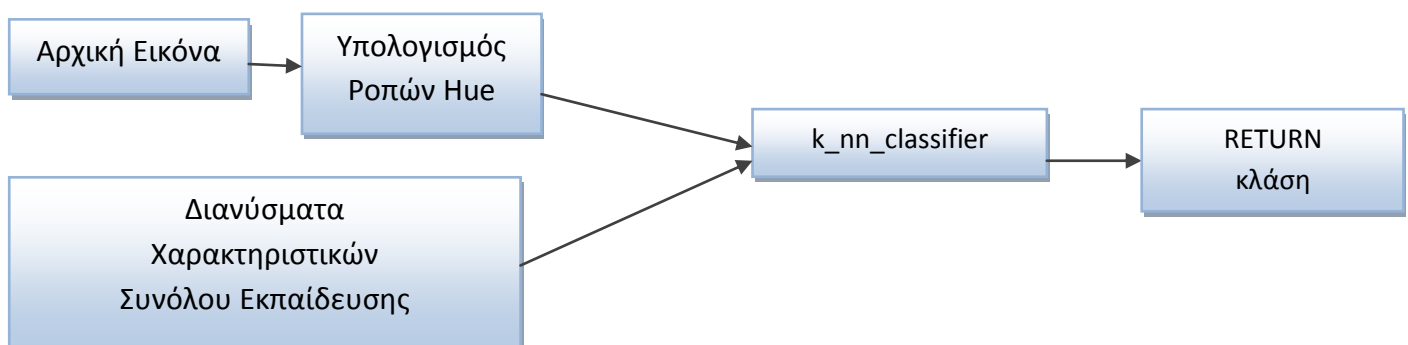
## *HueMoments.m*



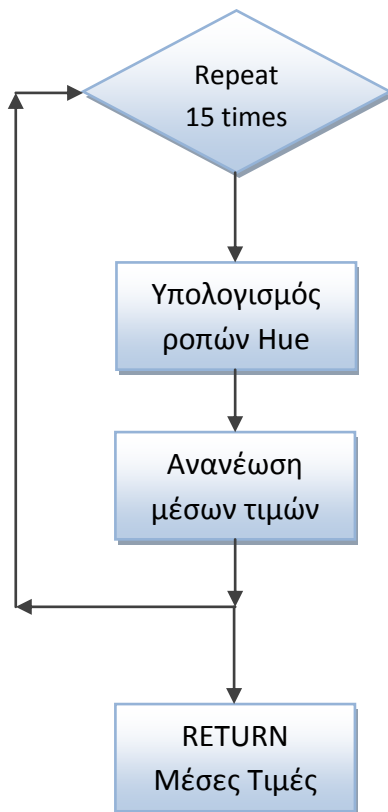
## *EuclideanClassifier.m*



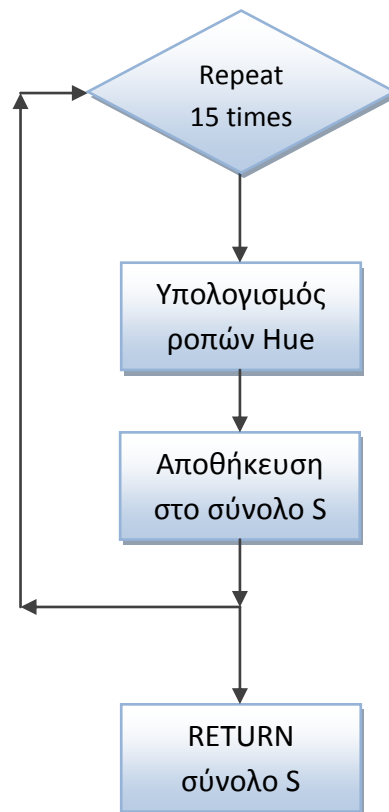
## *NNClassifier.m*



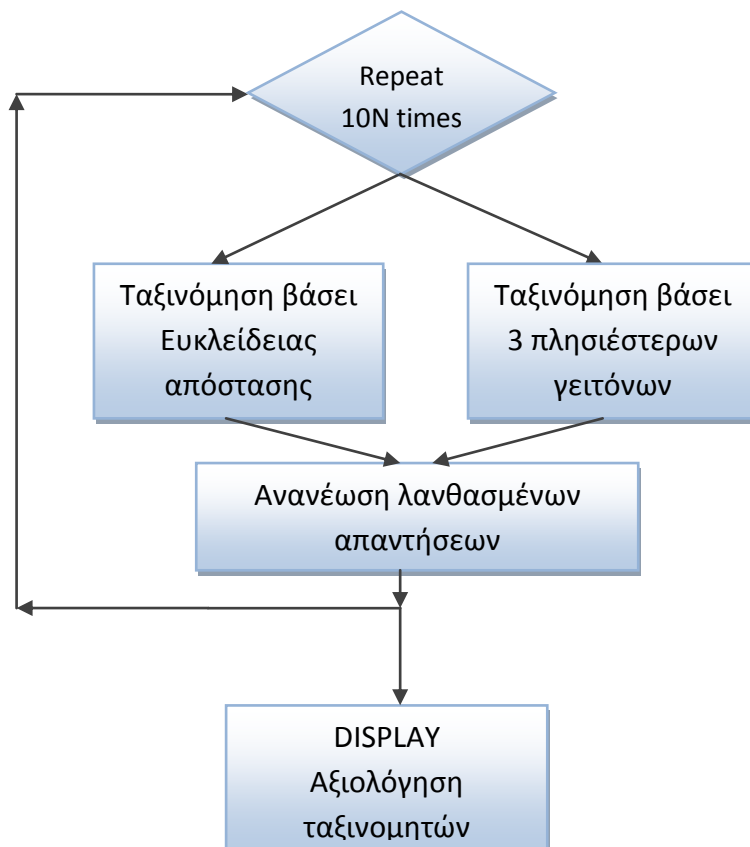
### ***EuclideanTraining.m***



### ***NNTraining.m***



### ***Evaluator.m***





## Vectorization

Όλα τα τμήματα του κώδικα έχουν γίνει *vectorized*, δηλαδή οι *for*-επαναλήψεις έχουν αντικατασταθεί με πράξεις επί πινάκων, για ταχύτερη εκτέλεση.

Παρακάτω φαίνεται ένα τέτοιο παράδειγμα:

*% Υπολογισμός  $m_{10}$ ,  $m_{01}$*

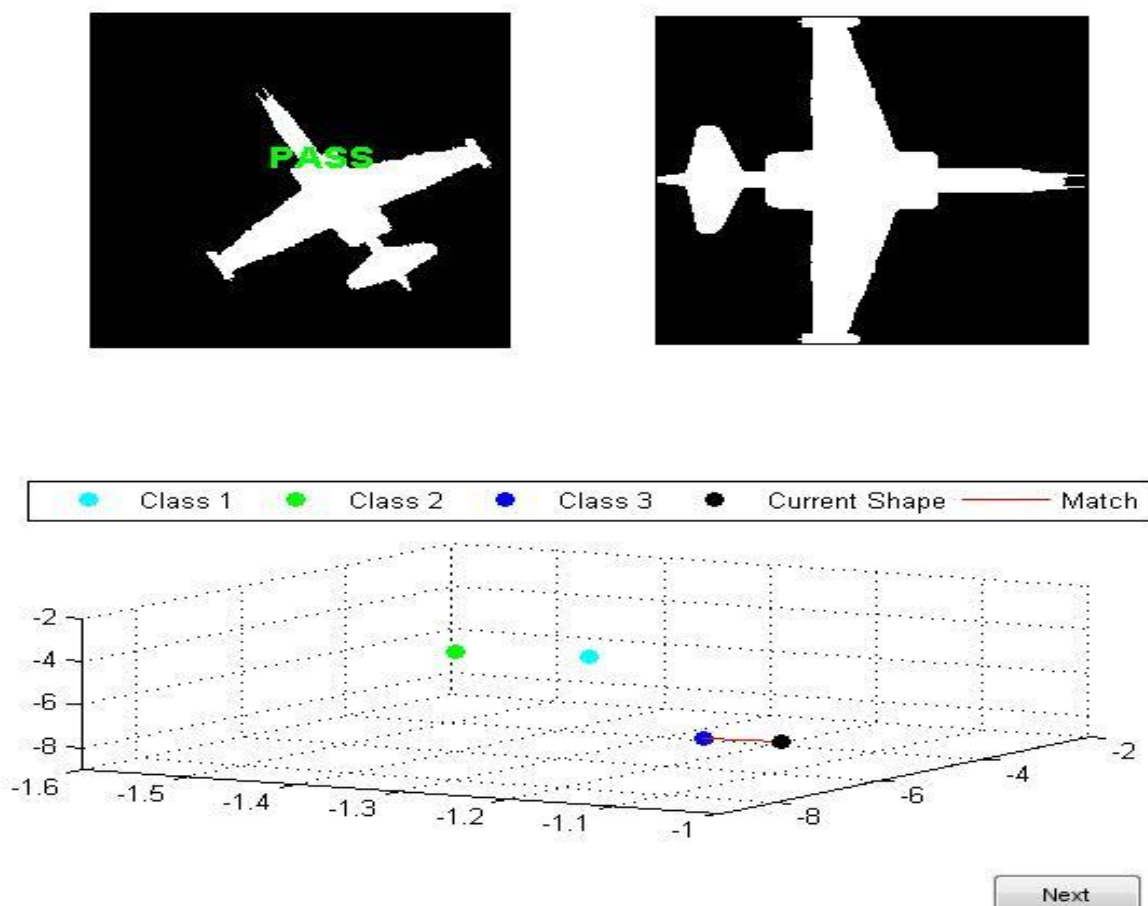
```
 $m_{10} = \text{sum}(\text{sum}(\text{image} .* \text{repmat}((1:M)', [1 \ N])));$ 
```

```
 $m_{01} = \text{sum}(\text{sum}(\text{image} .* \text{repmat}(1:N, [M \ 1])));$ 
```

## Graphic Display

Δίνεται η δυνατότητα δια-δραστικής εκτέλεσης του script *Evaluator.m* με την ανάθεση της μεταβλητής DISPLAY σε 1. Με αυτόν τον τρόπο, το αποτέλεσμα κάθε ταξινόμησης εμφανίζεται στην οθόνη μαζί με μια απεικόνιση των σημείων-διανυσμάτων των υπο-εξέταση χαρακτηριστικών στο τρισδιάστατο χώρο. Για την συνέχιση της εκτέλεσης ο χρήστης πατάει το κουμπί 'Next'.

Παρακάτω φαίνεται ένα στιγμιότυπο της εκτέλεσης:

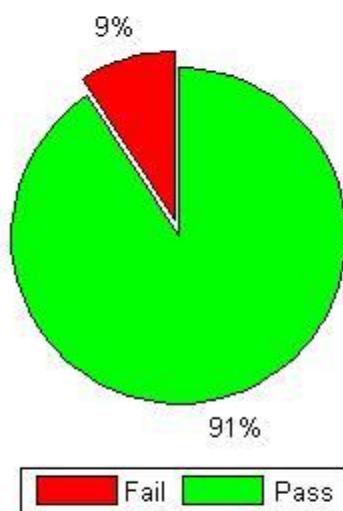


# Αξιολόγηση

Η αξιολόγηση των συστημάτων ταξινόμησης έγινε με βάση το ποσοστό των λανθασμένων απαντήσεων, μιας και το script *shape\_generator2.m* μας δίνει και τα σωστά αποτελέσματα για να επαληθεύσουμε.

Παρακάτω φαίνεται η τελική παρουσίαση της αξιολόγησης των συστημάτων ταξινόμησης του script *Evaluator.m* για 10000 τυχαίες εικόνες ( $N = 1000$ ).

**Euclidean Classifier**



**Nearest-neighbor Classifier**



## Συμπεράσματα

Η πλειοψηφία των λανθασμένων απαντήσεων οφείλονται στην παραγωγή εικόνων πολύ χαμηλής ανάλυσης, τις οποίες ούτε ένας άνθρωπος δεν μπορεί να ταξινομήσει με σιγουριά.

Οι υπόλοιπες λανθασμένες απαντήσεις οφείλονται στο γεγονός ότι οι τιμές των διανυσμάτων χαρακτηριστικών (ροπές Hue) της 1<sup>ης</sup> και της 2<sup>ης</sup> κλάσης είναι αρκετά παρόμοιες.

Οι δύο ταξινομητές έχουν σχεδόν ισοδύναμη αξιοπιστία, όπως φαίνεται και στα παραπάνω διαγράμματα. Επίσης, το ποσοστό αξιοπιστίας είναι πολύ καλό (91%, 90%). Εάν εξαιρέσουμε και τις περιπτώσεις εικόνων χαμηλής ποιότητας, το ποσοστό αξιοπιστίας είναι σχεδόν άριστο.

Στην περίπτωση του ταξινομητή βάσει της ευκλείδειας απόστασης, τα αποτελέσματα είναι αξιόπιστα διότι τα 5 δείγματα από το σύνολο εκπαίδευσης είναι αρκετά για να έχουμε αντιπροσωπευτικά κέντρα των κλάσεων.

Στην περίπτωση του ταξινομητή βάσει των 3 πλησιέστερων γειτόνων, τα αποτελέσματα είναι αξιόπιστα διότι τα 5 δείγματα από το σύνολο εκπαίδευσης είναι αρκετά για να έχουμε ομοιόμορφα κατανεμημένα σημεία που αντιπροσωπεύουν την εκάστοτε κλάση.