

# ΠΛΗΡΟΦΟΡΙΚΗ ΓΙΑ ΤΗΝ ΤΕΧΝΗ Η' ΕΞΑΜΗΝΟ

ΑΜΑΛΙΑ ΦΩΚΑ  
ΕΠΙΚΟΥΡΗ ΚΑΘΗΓΗΤΡΙΑ

Τμήμα Πλαστικών Τεχνών & Επιστημών της Τέχνης

# Περιεχόμενο Μαθήματος

2

- Αλληλεπίδραση

# Ζωντανές Εικόνες – Living Images

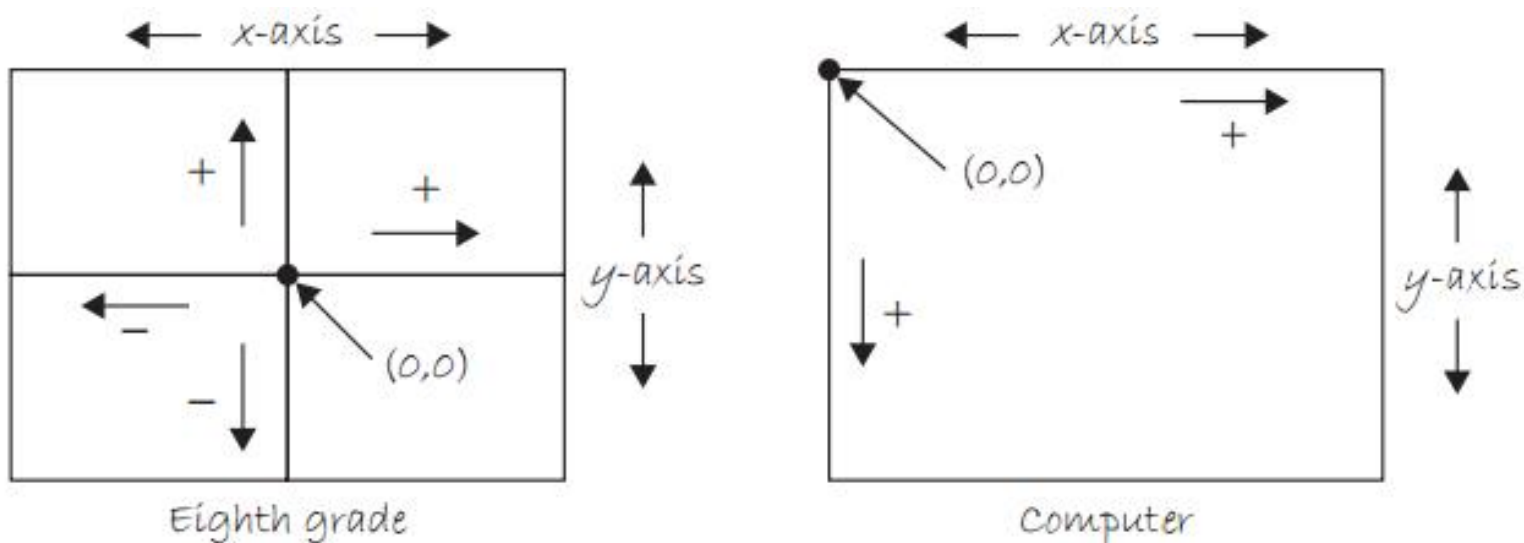
3

- Μια εικόνα αλλάζει ανάλογα με τις ενέργειες του θεατή.
- Αναπαριστά μια συνεργασία του καλλιτέχνη με το κοινό.
- Ψηφιακές Εικόνες
  - ▣ αποτελούνται από pixels
  - ▣ κάθε pixel έχει ένα χρώμα
    - τιμές για κόκκινο, μπλε, πράσινο (RGB)
  - ▣ κάθε pixel βρίσκεται σε μια συγκεκριμένη θέση του πλέγματος
- Τα στοιχεία μιας ψηφιακής εικόνας
  - ▣ Θέση των pixel
  - ▣ Το μέγεθος των pixel
  - ▣ Η κίνηση των pixel
  - ▣ Το χρώμα των pixel
- μπορεί να εξαρτάται από κάτι που συμβαίνει στο περιβάλλον

# Ψηφιακή Εικόνα

4

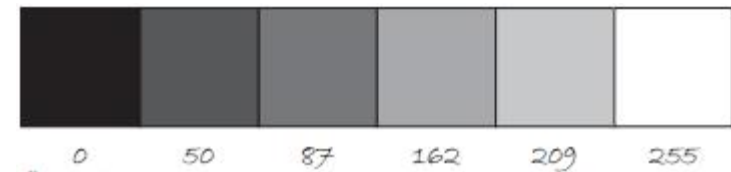
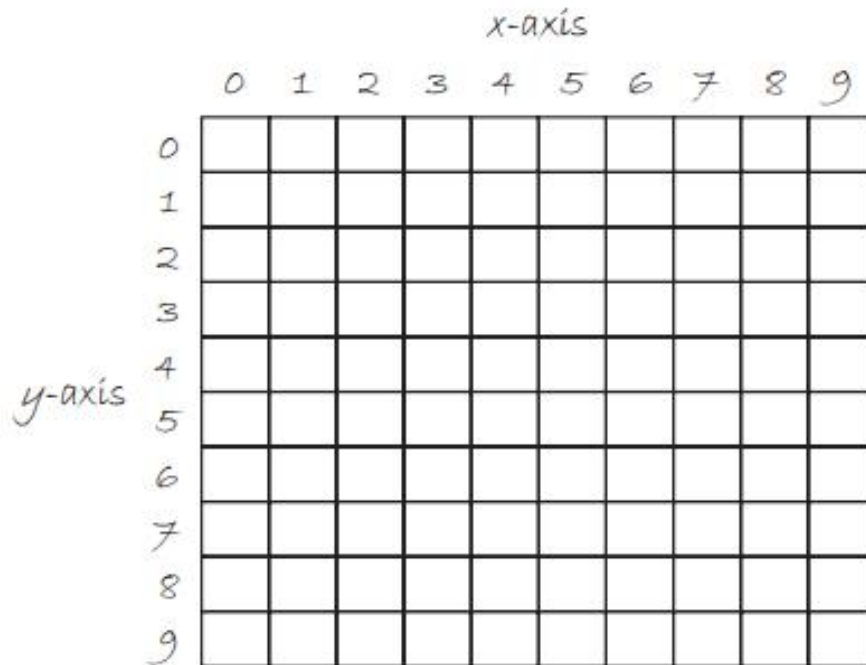
## □ Σύστημα Συντεταγμένων



# Ψηφιακή Εικόνα

5

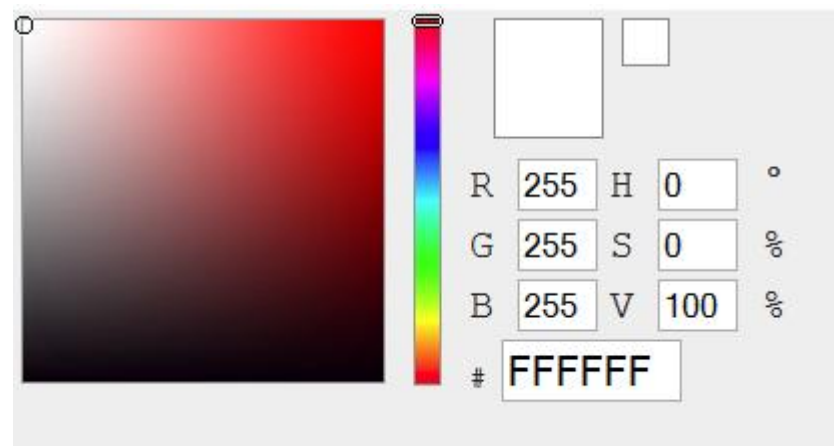
- Σύστημα Συντεταγμένων και Χρώμα



# Ψηφιακή Εικόνα

6

- Σύστημα Χρωμάτων RGB
  - ▣  $RGB = (R*65536)+(G*256)+B$ 
    - White RGB code =  $255*65536+255*256+255 = \#FFFFFF$
    - Blue RGB code =  $0*65536+0*256+255 = \#0000FF$
    - Red RGB code =  $255*65536+0*256+0 = \#FF0000$



# Image Recoder, Richard Colson, 2006

7

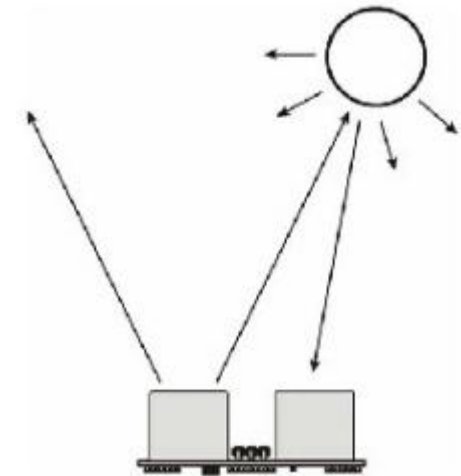
- Το μέγεθος κάθε pixel εξαρτάται από πόσο κοντά είναι ο θεατής
  - ▣ δύο αισθητήρες υπερήχων
- Τα δεδομένα διαβάζονται από έναν μικροελεγκτή.
- Ο στόχος του έργου ήταν
  - ▣ να υπογραμμίσει το γεγονός ότι η ανθρώπινη αντίληψη δεν είναι αντικειμενική
  - ▣ ότι υπάρχουν πάρα πολλά πράγματα που τείνουν να παρεμβαίνουν σε μια αδιαμφισβήτητη και αδιάλειπτη όψη του κόσμου
- <https://www.ravensbourne.ac.uk/staff/richard-colson/>



# Αισθητήρες Υπερήχων (Ultrasonic sensors)

8

- Υπέρηχος
  - κύμα το οποίο βρίσκεται πάνω από την μέγιστη συχνότητα που μπορεί να ακούσει το ανθρώπινο αυτί.
- Αρχή λειτουργίας αισθητήρα
  - Από το μεγάφωνο γίνεται αποστολή ενός υπερήχου
  - Αν υπάρχει κάποιο αντικείμενο η ηχώ του υπερήχου επιστρέφει προς τον αισθητήρα
  - Γίνεται αισθητή μέσω μικροφώνου
  - Γνωρίζοντας τον χρόνο του ταξιδιού του υπερήχου από την μετάδοση ως την λήψη του
  - Υπολογίζουμε πόση απόσταση διέσχισε ο υπέρηχος
- Υπολογίζει απόσταση από 2 εκ. έως 4 μ.
- Μικρό κόστος (4€)





# Wooden Mirror

## Daniel Rozin, 1999

9

- Κάθε ένα από τα ξύλινα κομμάτια του καθρέφτη περιστρέφεται
  - ▣ 830 σερβοκινητήρες ελέγχουν την γωνία κάθε κομματιού
  - ▣ κάμερα παίρνει την εικόνα
  - ▣ λογισμικό επεξεργάζεται την εικόνα από την κάμερα
  - ▣ βρίσκει την γωνία κατά την οποία πρέπει να περιστραφεί κάθε κομμάτι
  - ▣ αντικατοπτρίζει διαφορετική ποσότητα φωτός και έτσι δημιουργείται η εικόνα
- διερευνά τη διαχωριστική γραμμή μεταξύ ψηφιακού και φυσικού, χρησιμοποιώντας ένα ζεστό και φυσικό υλικό, όπως το ξύλο για να απεικονίσει την αφηρημένη έννοια του ψηφιακού pixel
- <http://www.smoothware.com/danny/woodenmirror.html>



# Swarm

## Daniel Shiffman, 2004

10

- Εμπνευσμένο από την τεχνική του Pollack “drip and splash”
- Λογισμικό επανατοποθετεί τα pixel σε κάθε καρέ του βίντεο σε πραγματικό χρόνο ως απάντηση στην κίνηση του θεατή
- η κίνηση υλοποιείται ε την τεχνική του bird flocking
- <http://shiffman.net/projects/swarm/>



# Νοημοσύνη Σμήνους – Flocking Birds

11

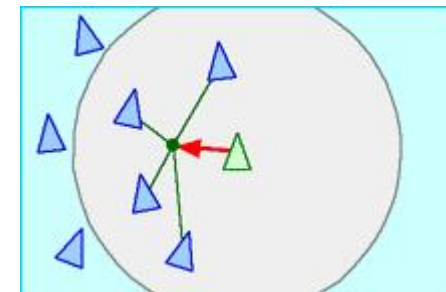
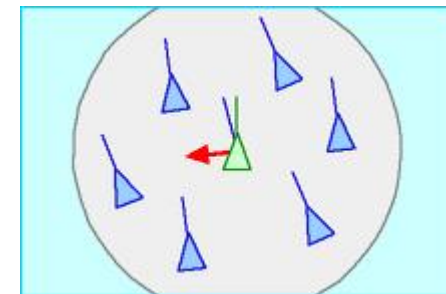
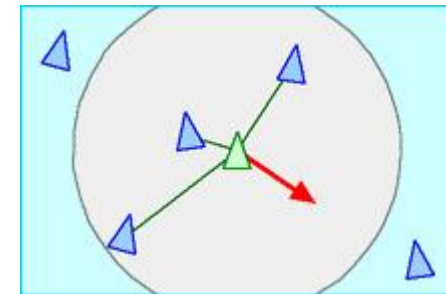
- Προσομοίωση της συμπεριφοράς ομάδας οργανισμών από την φύση
- Η συμπεριφορά ενός σμήνους πουλιών ακολουθεί κάποιους κανόνες
  - ικανότητα να αυτό-οργανώνονται με περίπλοκο τρόπο και να επιδεικνύουν σύνθετες και αυτορυθμιζόμενες συμπεριφορές



# Νοημοσύνη Σμήνους – Flocking Birds

12

- Το 1987 ο Craig Reynolds διατύπωσε τρεις βασικούς κανόνες που πρέπει να ακολουθήσουν τα πουλιά για να σχηματίσουν ένα σμήνος
  - ▣ **Αποχωρισμού:** Να μην πλησιάζουν πέρα από ένα ορισμένο όριο άλλα αντικείμενα ή πουλιά.
  - ▣ **Ευθυγράμμισης:** Να διατηρούν την ίδια ταχύτητα και κατεύθυνση με τους πλησιέστερους συντρόφους τους.
  - ▣ **Συνοχής:** Να κατευθύνονται προς το κέντρο της πιο κοντινής ομάδας



# Νοημοσύνη Σμήνους – Flocking Birds

13

- Λογισμικό Boids (birds + oids)
  - Νυχτερίδες και πιγκουίνοι στο «Η επιστροφή του Μπάτμαν» (1992)
  - Δεινόσαυροι στο «Τζουράσικ Πάρκ» (1993)
  - Διαστημικά πλάσματα στο Voyager του Star Trek
- <https://www.youtube.com/watch?v=M028vafB0l8>
- <https://processing.org/examples/flocking.html>

# Διαδικασία Αλληλεπίδρασης

14

- Είσοδος εικόνας
  - ▣ webcam
  - ▣ video camera
  - ▣ εικόνες σταθερές (bitmap)
- Είσοδος αλλαγών στο περιβάλλον
  - ▣ ανάλυση εικόνων (frames από βίντεο) για εξαγωγή κίνησης
  - ▣ αισθητήρες
- Λογισμικό
  - ▣ ανάλυση δεδομένων εισόδου
  - ▣ επεξεργασία εικόνας pixel-by-pixel
  - ▣ τροποποίηση εικόνας
  - ▣ προγραμματισμός (π.χ. Processing)

# Παράδειγμα επεξεργασίας εικόνας σε Processing

15

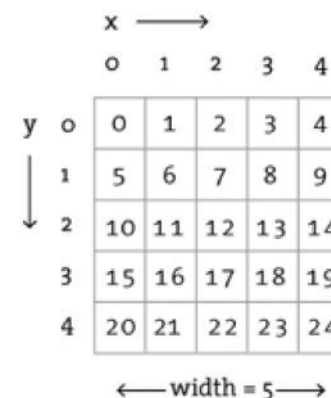
- Οι εικόνες είναι δύο διαστάσεων αλλά οι τιμές για κάθε pixel αποθηκεύονται σε μονοδιάστατο πίνακα

How the pixels look:

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24

How the pixels are stored:

0	1	2	3	4	5	6	7	8	9	.	.	.		
---	---	---	---	---	---	---	---	---	---	---	---	---	--	--



Pixel 13 has an x value of 3 and y value of 2.

$$\begin{aligned}x + (y * \text{width}) \\&= 3 + (2 * 5) \\&= 3 + 10 \\&= 13\end{aligned}$$

# Παράδειγμα επεξεργασίας εικόνας σε Processing

16

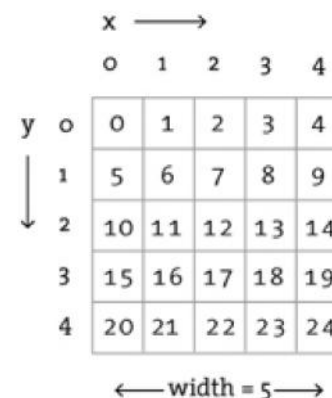
- Οι εικόνες είναι δύο διαστάσεων αλλά οι τιμές για κάθε pixel αποθηκεύονται σε μονοδιάστατο πίνακα

How the pixels look:

0	1	2	3	4
5	6	7	8	9
10	11	12	13	14
15	16	17	18	19
20	21	22	23	24

How the pixels are stored:

0	1	2	3	4	5	6	7	8	9	.	.	.		
---	---	---	---	---	---	---	---	---	---	---	---	---	--	--



Pixel 13 has an x value of 3 and y value of 2.

$$\begin{aligned}x + (y * \text{width}) \\&= 3 + (2 * 5) \\&= 3 + 10 \\&= 13\end{aligned}$$



# Το αντικείμενο Pimage στην Processing

17

- για να «ανοίξουμε» μια εικόνα και
- να την επεξεργαστούμε pixel-by-pixel
- χρησιμοποιούμε ένα αντικείμενο που ονομάζεται PImage
- στη συνέχεια μπορούμε να διαβάσουμε την τιμή χρώματος κάθε pixel
- αν θέλουμε να αλλάξουμε τις τιμές σε κάποια pixel
- και τέλος να αποθηκεύσουμε την τροποποιημένη εικόνα ή
- να την εμφανίσουμε στην οθόνη

# Βασικό Παράδειγμα με PImage

18

□ <https://processing.org/tutorials/pixels/>

```
PImage img;

void setup() {
  size(200, 200);
  img = loadImage("sunflower.jpg");
}

void draw() {
  loadPixels();
  // Since we are going to access the image's pixels too
  img.loadPixels();
  for (int y = 0; y < height; y++) {
    for (int x = 0; x < width; x++) {
      int loc = x + y*width;

      // The functions red(), green(), and blue() pull out the 3 color components from a pixel.
      float r = red(img.pixels[loc]);
      float g = green(img.pixels[loc]);
      float b = blue(img.pixels[loc]);

      // Image Processing would go here
      // If we were to change the RGB values, we would do it here, before setting the pixel in the c

      // Set the display pixel to the image pixel
      pixels[loc] = color(r,g,b);
    }
  }
  updatePixels();
}
```

# Αισθητήρες

19

- Ακουστικής, ήχου, δόνησης
- Αυτοκίνητου, Μέσων Μεταφοράς
- Χημικοί
- Ηλεκτρικού ρεύματος, Τάση Ρεύματος, Μαγνητικοί
- Περιβάλλοντος, Καιρικών Συνθηκών, Υγρασίας
- Ροής, Ταχύτητας Ρευστών
- Ιονίζουσας Ακτινοβολίας
- Όργανα πλοήγησης
- Θέσης, γωνίας, μετατόπισης, απόστασης, ταχύτητας, επιτάχυνσης
- Οπτικοί, Φωτός, Απεικόνισης, Φωτονίων
- Πίεσης
- Δύναμης, Πυκνότητας, Έντασης
- Θερμικοί
- [http://en.wikipedia.org/wiki/List\\_of\\_sensors](http://en.wikipedia.org/wiki/List_of_sensors)