Γραφική με Υπολογιστή
Computer Graphics

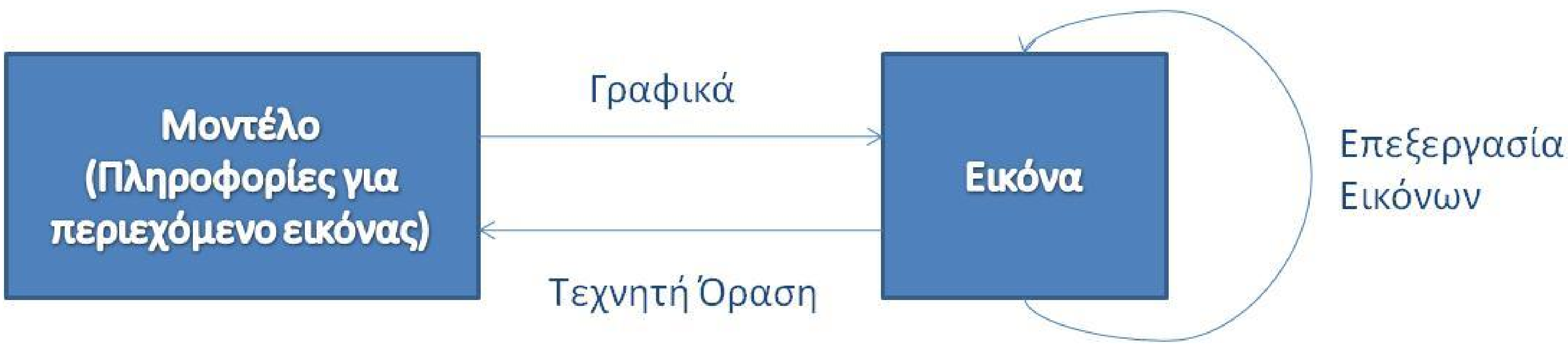
1. Βασικοί γραφικοί αλγόριθμοι
2. Αρχές γραφικών πλεγματικών οθονών raster
3. Μετασχηματισμοί 2 και 3 διαστάσεων και συστήματα συντεταγμένων
4. Προβολές και μετασχηματισμοί παρατήρησης
5. Αλγόριθμοι απόκρυψης επιφανειών και αλγόριθμοι φωτισμού

διαφάνειες από Βιβλίο Θεοχάρη Μπέμ, Συμμετρία, Αθήνα 1999

Ιστορική αναδρομή

Ivan Sutherland (1963 διδακτορική διατριβή), Διανυσματική οθόνη, γραφίδα-δεικτική συσκευή

Περιοχές σχετικές με την Εικόνα

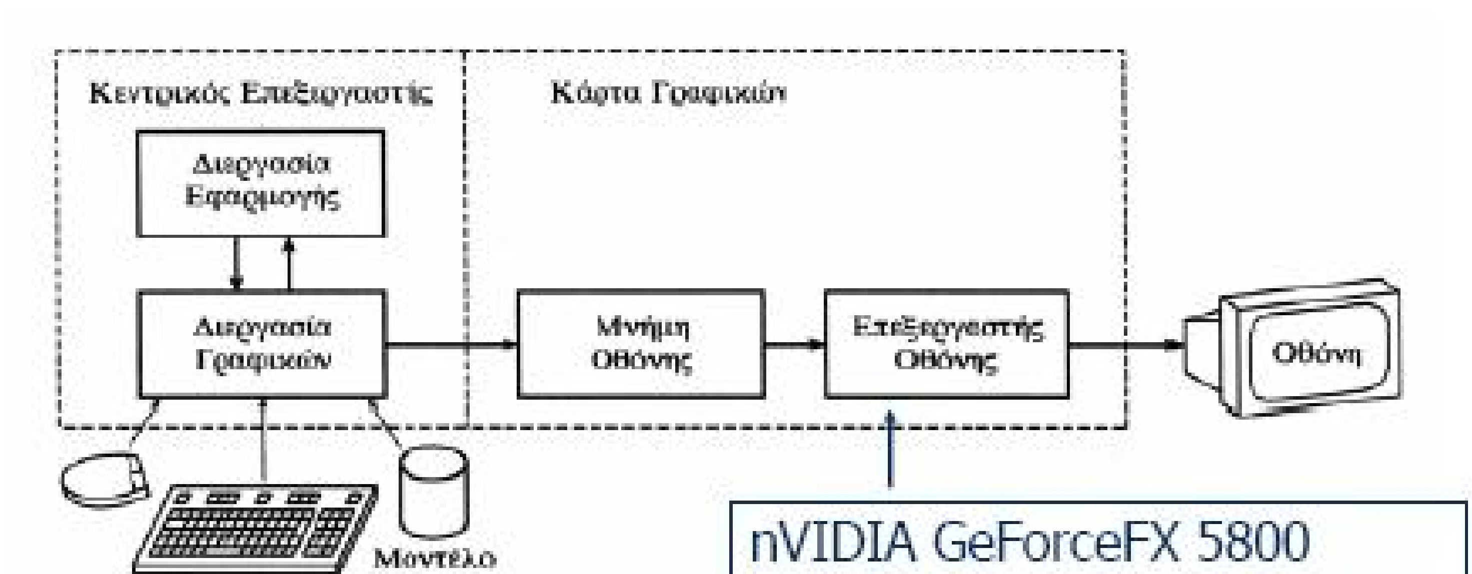


- Γραφική με Υπολογιστές (Computer Graphics)
- Επεξεργασία - Ανάλυση Εικόνων (Image Processing - Analysis)
- Τεχνητή όραση (Computer vision)
- Αλληλεπίδραση Ανθρώπου - Υπολογιστή

Περιοχές εφαρμογών

- Γραφική αλληλεπίδραση με χρήστη (GUI).
- Σχεδίαση (CAD): πχ Αρχιτεκτονική , VLSI design
- Γεωγραφικά Συστήματα Πληροφοριών (GIS).
- Προσομοιώσεις (π.χ. πτήσεων).
- Συνθετικές ταινίες & διαφήμιση – Computer Animation.
- Επαυξημένη Πραγματικότητα - Augmented Reality
- Εικονική Πραγματικότητα – Virtual Reality
- Ιατρικές εφαρμογές.
- Οπτικοποίηση δεδομένων.
- Τέχνη (π.χ. fractals).
- Παιχνίδια.

Γραφικό σύστημα επεξεργασίας

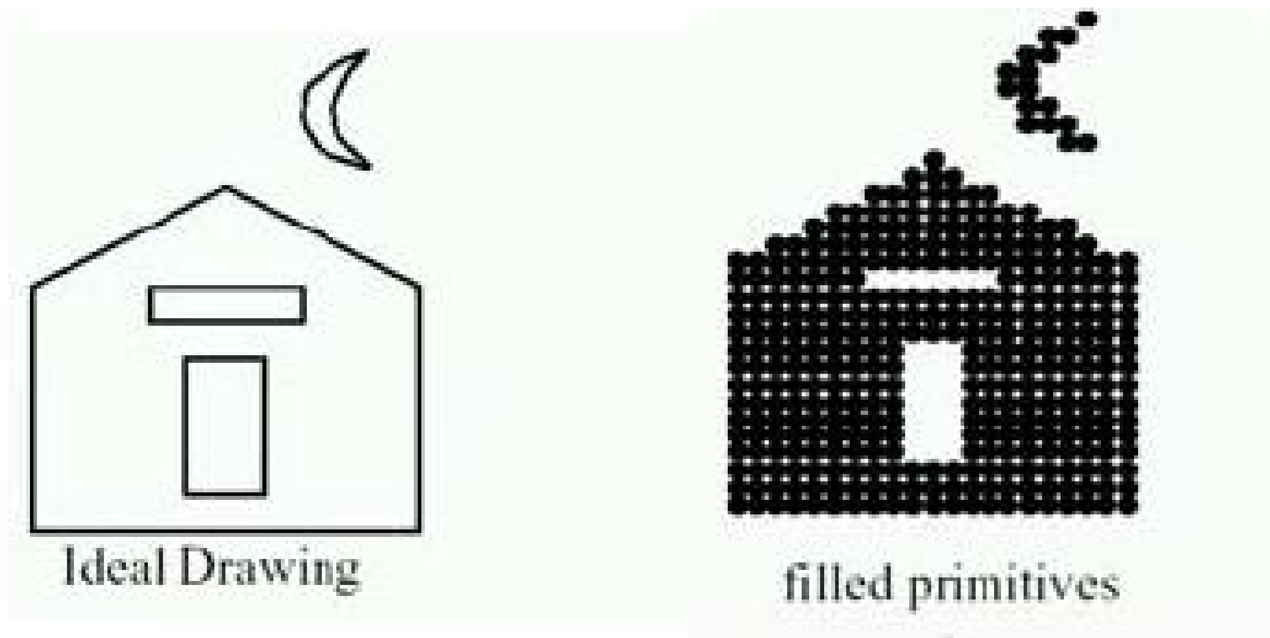


nVIDIA GeForceFX 5800

nVIDIA GeForce4 Ti 4800

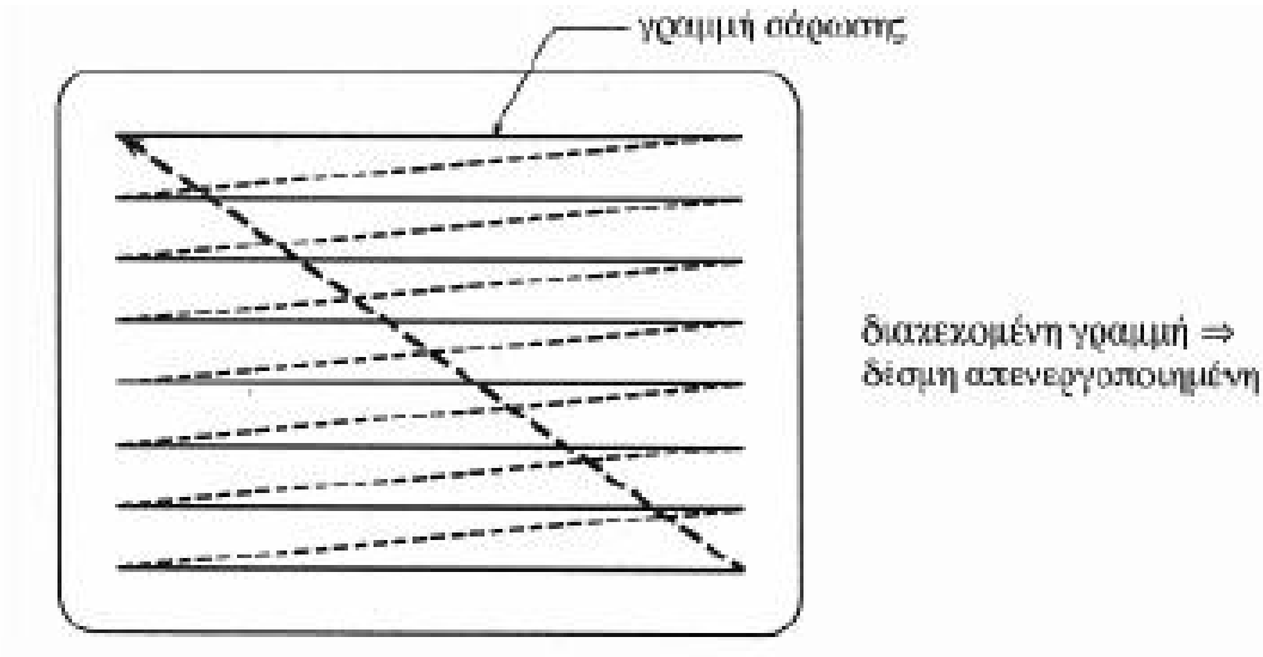
ATI Radeon 9700 Pro

Πλεγματική ή διανυσματική απεικόνιση



Πλεγματική οθόνη (raster display)

- Σάρωση της οθόνης ανά γραμμή και σε συγκεκριμένο χρονικό ρυθμό



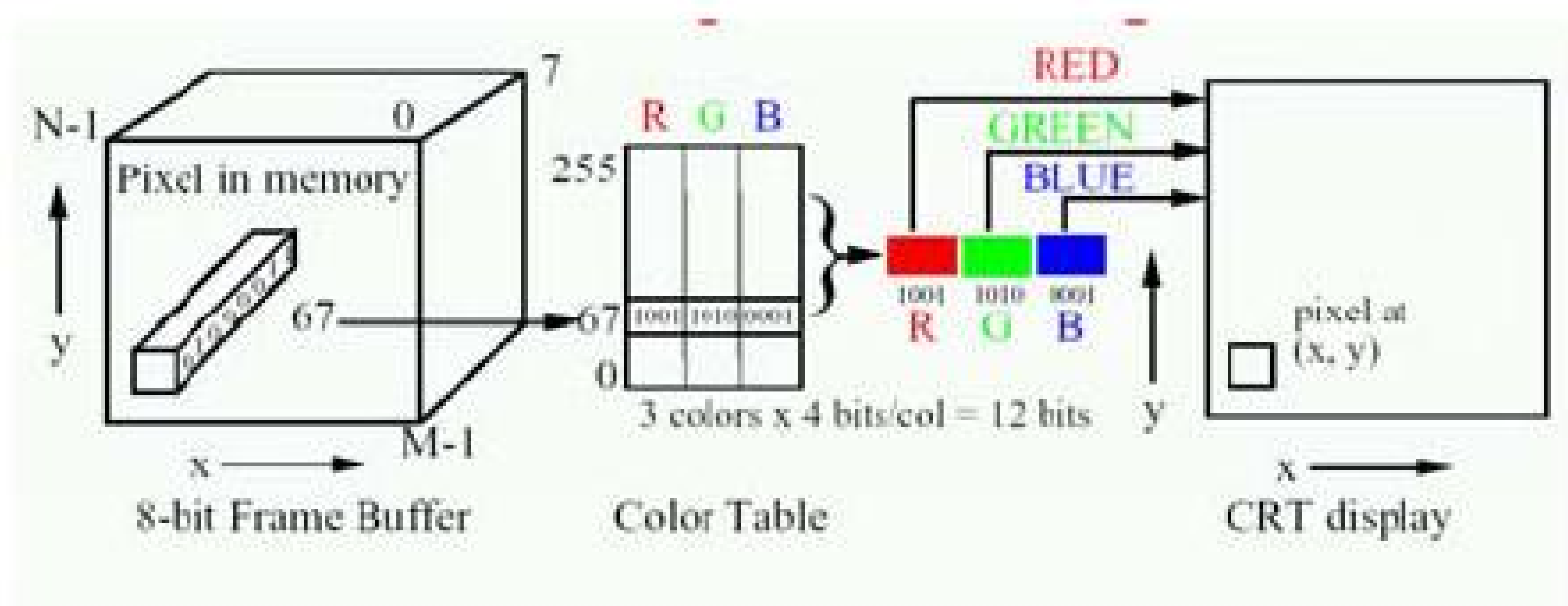
Πλεγματική οθόνη: αποθήκευση εικόνας στη μνήμη

- Αποθήκευση της εικόνας στη μνήμη ως πίνακα τριών διαστάσεων

(πλάτος x ύψος x χρώμα)

- Χρήση παλέτας για μείωση απαιτήσεων χρώματος.

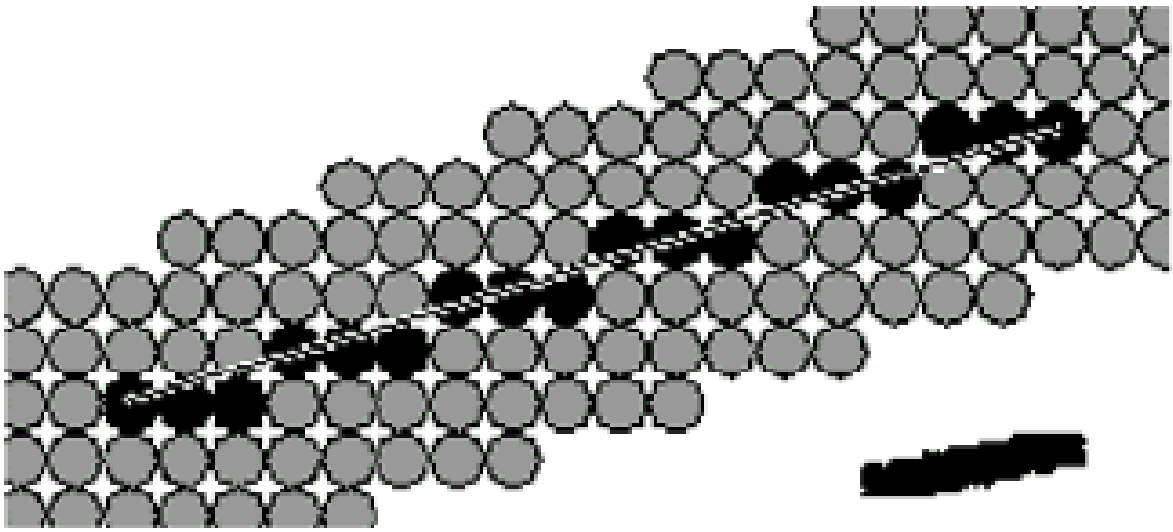
Χρήση παλέτας χρωμάτων



Πλεγματική οθόνη/ 2

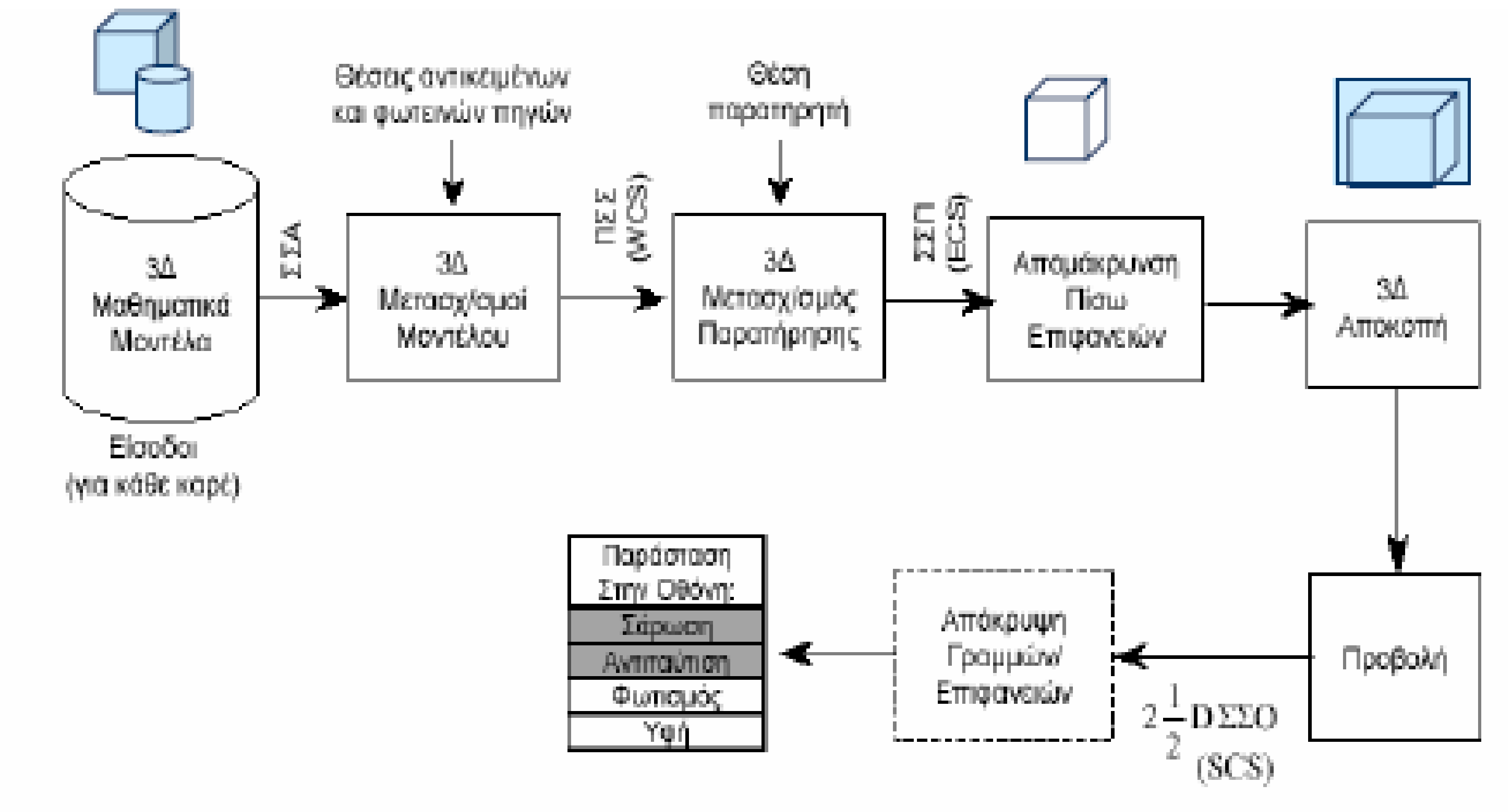
- 2-Δ πλέγμα χρωματιζόμενων εικονοστοιχείων (pixels)
- Φρεσκάρισμα σε σταθερό ρυθμό από μνήμη
- Κανένας περιορισμός σε εικόνα
- Αποδέσμευση φρεσκαρίσματος από άλλες λειτουργίες
- Μεγάλη (;) απαίτηση μνήμης
  (1024 x 1024 x 24= 3MB)
- Μείωση με χρήση παλέτας (lookup table)
- Προβλήματα ταύτισης (aliasing)
- Νέες τεχνολογίες (LCD, plasma)

Πλεγματική οθόνη /3



Απαιτούνται αλγόριθμοι απεικόνισης ευθείας, καμπύλης, πλήρωσης πολυγώνου σε πλεγματική οθόνη

Αλγόριθμος γραφικής απεικόνισης



Παράσταση σχημάτων

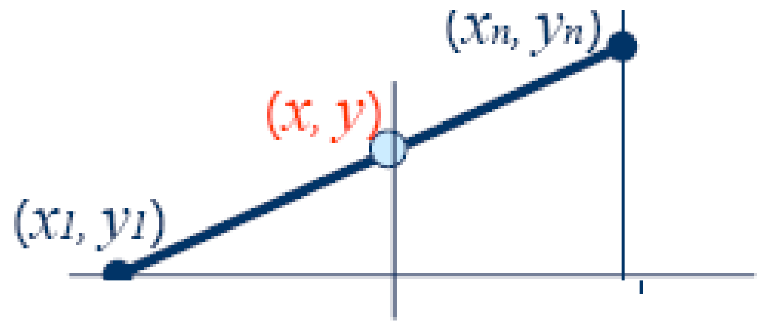
ΠΑΡΑΣΤΑΣΗ ΣΤΗΝ ΟΘΟΝΗ
Σάρωση
Αντιταύτιση
Φωτισμός
Υφή

Απεικόνιση Ευθύγραμμου τμήματος

- Αλγόριθμοι υποστηρίζονται και από hardware.
- Κριτήρια «καλού» αλγορίθμου:
  - Όσο γίνεται πιο κοντά στη μαθηματική πορεία
  - Πάχος σταθερό και ανεξάρτητο από μήκος και κλίση.
  - Με όσο το δυνατόν μικρότερο υπολογιστικό κόστος (μεγαλύτερη ταχύτητα).

Αλγόριθμος 1

- Έστω ευθύγραμμο τμήμα μεταξύ pixels $P_1(x_1,y_1)$, $P_n(x_n,y_n)$ στο πρώτο οκταμόριο.



Τότε: $y=s*x+b$

όπου

$$s = \frac{y_n - y_1}{x_n - x_1} = \frac{\Delta y}{\Delta x} \quad \text{και} \quad b = \frac{y_1 x_n - y_n x_1}{x_n - x_1}$$

Υλοποίηση

```
void line1( x1, y1, xn, yn, colour)
int x1,y1, xn, yn, colour;
/* colour: ακέραια αναπαράσταση του χρώματος*/
{
     float s, b, y;
    int x;
    s=( yn- y1)/( xn- x1);
    b=( y1* xn- yn* x1)/( xn- x1);
    for (x= x1; x<= xn; x++){
            y= s* x+ b;
            setpixel( x, round( y), colour);
    }
}
```

Αλγόριθμος 2 (αποδοτικότερος)

- Ο πολλαπλασιασμός μέσα στο βρόγχο μπορεί να αποφευχθεί μετατρέποντας τον τύπο σε αναδρομικό:

$$
\begin{aligned}
& y_{i+1} = sx_{i+1} + b = \\
& s(x_i + 1) + b = \\
& sx_i + b + s = \\
& y_i + s
\end{aligned}
$$

Υλοποίηση

```
line2 (x1, y1, xn, yn, colour)
int x1,y1, xn, yn, colour;
{
float s, y;
int x;
s=( yn- y1)/( xn- x1);
y= y1;
    for (x= x1; x<= xn; x++){
            setpixel( x, round( y), colour);
            y= y+ s;
    }
}
```

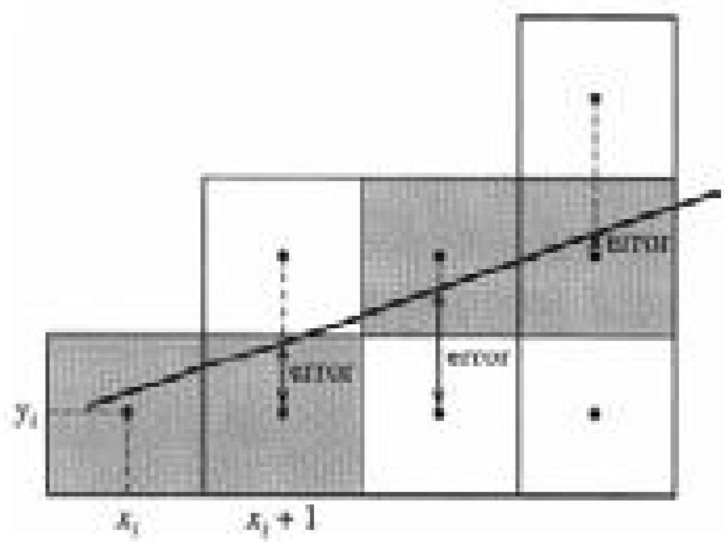
Αλγόριθμος 3

- Αποφυγή της στρογγυλοποίησης μέσα στο βρόγχο με χωρισμό του y σε ακέραιο και δεκαδικό μέρος (error) και χρήση μόνο του ακεραίου για εύρεση του επομένου σημείου.

- Σφάλμα =

  απόσταση pixel

  από ιδεατή πορεία.



Υλοποίηση

```
line3 (x1, y1, xn, yn, colour)
int x1,y1, xn, yn, colour;

{
float s, error;
int x, y;
    s=( yn- y1)/( xn- x1);
    y= y1;
    error= 0;
    for (x= x1; x<= xn; x++){
            setpixel( x, y, colour);
            error= error+ s;
        if (error>= 0.5){ y++; error--}
    }
}
```

Αλγόριθμος 4 (Bresenham)

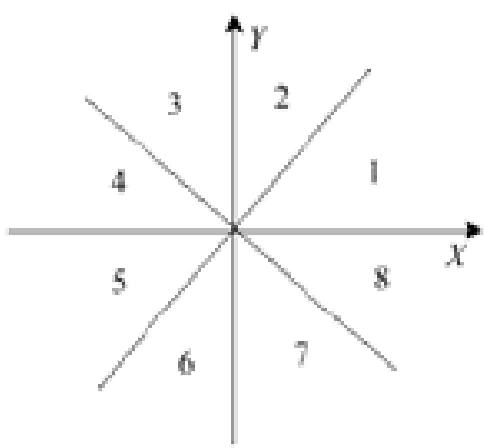
- Πολλαπλασιάζοντας με dx=xn-x1, οι μεταβλητές s και error γίνονται ακέραιες.
- Η σύγκριση για την επιλογή του επόμενου pixel γίνεται με floor(dx/2). Επίσης, αφαιρώντας από την αρχική τιμή του error dx, η σύγκριση μπορεί να γίνει και με το 0.
- Λειτουργεί για θετικές κλίσεις.

Υλοποίηση

```
line4 (x1, y1, xn, yn, colour)
int x1,y1, xn, yn, colour;
{
int error, x, y, dx, dy;
dx= xn- x1; dy= yn- y1;
error=- dx/ 2; y= y1;
    for (x= x1; x<= xn; x++){
            setpixel( x, y, colour);
            error= error+ dy;
            if (error>= 0){ y++; error= errordx}
    }
}
```

Περιορισμοί

- Μόνο στο 1ο ογδοημόριο. Εύκολα μπορεί να επεκταθεί και στα άλλα με κατάλληλες εναλλαγές των ρόλων x, y, αρχικού και τελικού σημείου. Άξονας ταχύτερης κίνησης και είδος μεταβολής του άλλου άξονα δίνονται στο σχήμα.



Οκταμόριο	Άξονας ταχυτ. Κίνησης	Άλλος άξονας
1	x	Αυξάνεται
2	y	»
3	y	Μειώνεται
4	x	»
5	x	Αυξάνεται
6	y	»
7	y	Μειώνεται
8	x	»

Περιορισμοί (συνέχεια)

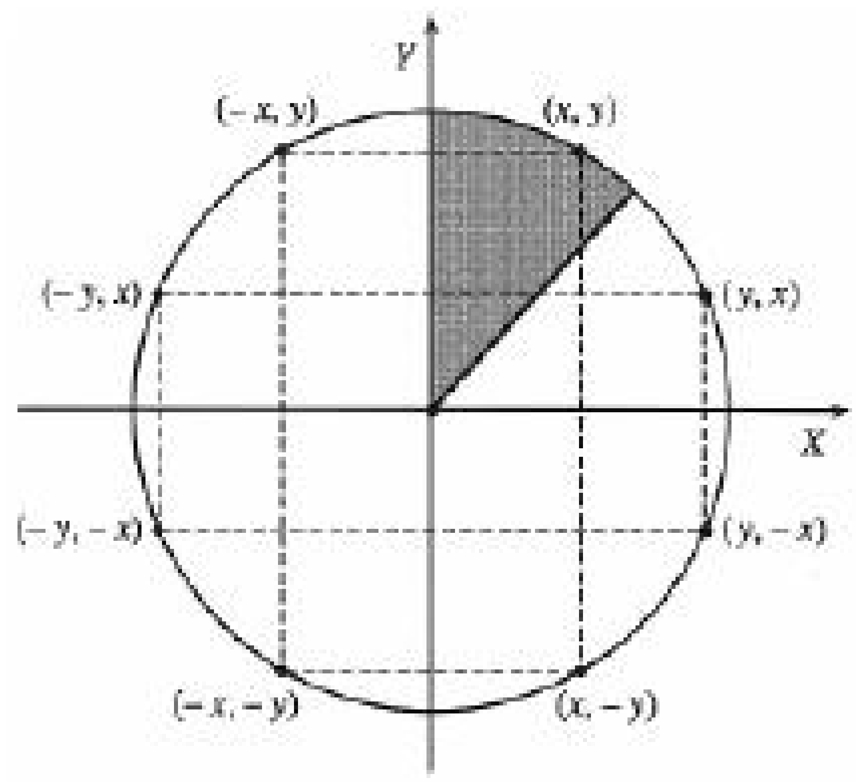
- Ευθύγραμμα τμήματα διαφορετικών κλίσεων μπορεί να έχουν διαφορετική φωτεινότητα. Το φαινόμενο αυτό μπορεί να διορθωθεί με τεχνικές αντι-ταύτισης (anti-aliasing).

Το φαινόμενο της ταύτισης (aliasing) προβλήματα δειγματοληψίας

- Τεθλασμένη εμφάνιση ευθυγράμμων τμημάτων και ακμών πολυγώνων
- Λανθασμένη εμφάνιση επαναλαμβανόμενων τμημάτων ενός σχήματος
- Λανθασμένη εμφάνιση μικρών σχημάτων που μπορεί να εμφανίζονται ή να εξαφανίζονται ανάλογα με τη θέση τους
- Μέθοδοι αντιταύτισης: Pitteway&Watkinson, postfiltering, stochastic sampling, A-buffer…)

Αλγόριθμος σχεδίασης κύκλου σε πλεγματική οθόνη

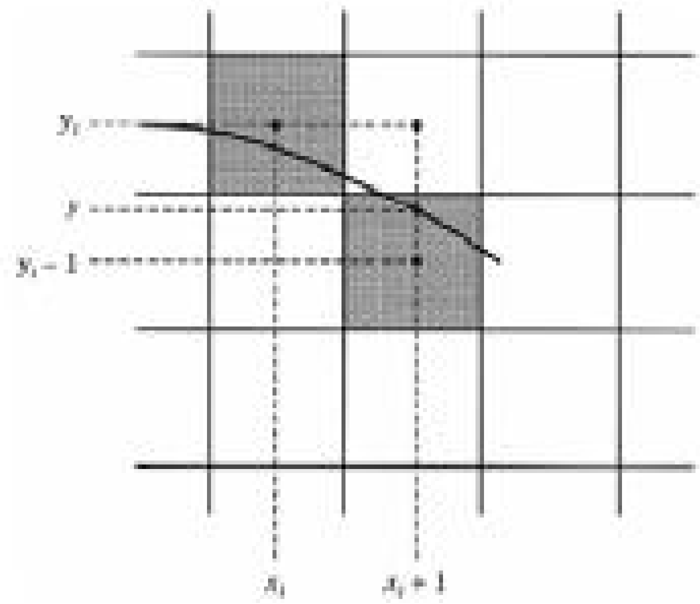
- Εκμετάλλευση 8-πλής συμμετρίας:



Συμμετρία κύκλου

```
circle_ symmetry (x, y, colour)
int x, y, colour;
{
      setpixel( x, y, colour);
      setpixel( y, x, colour);
      setpixel( y,- x, colour);
      setpixel( x,- y, colour);
      setpixel(- x,- y, colour);
      setpixel(- y,- x, colour);
      setpixel(- y, x, colour);
      setpixel(- x, y, colour);
}
```

Αλγόριθμος Bresenham



- Έστω ότι επελέγη το σημείο (x_i, y_i). Το επόμενο θα είναι (x_i+1, y_i) ή (x_i+1, y_i-1);

Αλγόριθμος

Μεταβλητή απόφασης:

$$e_i = d_1 - d_2$$

όπου

$d_1 = y_i^2 - y^2$, $d_2 = y^2 - (y_i - 1)^2$

Εάν $e_i \geq 0$, επιλέγεται το (x_i+1, y_i-1)

αλλιώς επιλέγεται το (x_i+1, y_i)

Υπολογισμός του e

- Για $x=x_i+1$ ισχύει ότι $y^2=r^2$, οπότε:

  $e_i = y_i^2 - r^2 + (x_i+1)^2 + (y_i-1)^2 - r^2 + (x_i+1)^2 \Leftrightarrow$

  $e_i = 2(x_i+1)^2 + y_i^2 + (y_i-1)^2 - 2r^2$

Αναδρομικός υπολογισμός e

$$
\begin{aligned}
e_{i+1} &= 2\,(x_{i+1}+1)^2 + y_{i+1}{}^2 + (y_{i+1}-1)^2 - 2\,r^2 \Leftrightarrow \\
&= 2\,(x_i+2)^2 + y_{i+1}{}^2 + (y_{i+1}-1)^2 - 2\,r^2 \Leftrightarrow \\
&= 2x_i{}^2 + 8x_i + 8 + y_{i+1}{}^2 + y_{i+1}{}^2 - 2\,y_{i+1} + 1 - 2r^2 \Leftrightarrow \\
&= 2(x_i+1)^2 + 4x_i + 6 + 2y_{i+1}{}^2 - 2y_{i+1} + 1 - 2r^2 \Leftrightarrow \\
&= e_i + 4x_i + 6 + 2(y_{i+1}{}^2 - y_i{}^2) - 2(y_{i+1} - y_i)
\end{aligned}
$$

Τελικός τύπος υπολογισμού

Αν $e_i<0 \Rightarrow y_{i+1} = y_i \Rightarrow e_{i+1}= e_i + 4(x_i+ 1)+ 2$

Αν $ei\geq 0 \Rightarrow y_{i+1} = y_i - 1$

$\Rightarrow e_{i+1} = e_i + 4x_i+ 6 + 2((y_i -1)^2 - y_i^2)$

$- 2(y_i-1- y_i)$

$\Rightarrow e_{i+1}= e_i + 4(x_i+1) + 2 - 4(y_i - 1)$

Παρατηρήσεις

Αρχική τιμή:

$e_1=2+ r^2 + (r-1)^2 - 2r^2 = 3 - 2r$

Για μεταφορά του κύκλου από το (0,0) στο (x_0,y_0) αρκεί να αντικατασταθεί η κλήση

circle_symmetry(x,y,colour);

από:

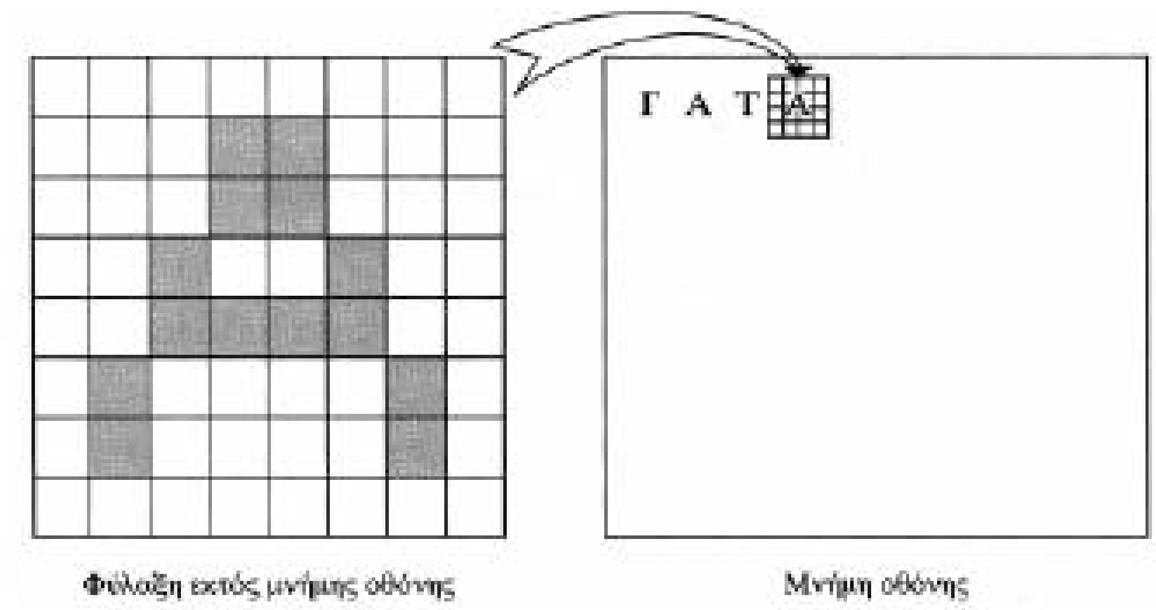
circle_symmetry(x-x$_0$,y-y$_0$,colour);

Υλοποίηση

```
circle (r, colour)
int r, colour;
{
    int x, y, e;
    x= 0;
    y= r;
    e= 3- 2* r;
    while (x<= y){
        circle_ symmetry( x, y, colour);
          x++;
          if (e>= 0) {   y--;     e= e- 4* y  }
          e= e+ 4* x+ 2;
          }
}
```

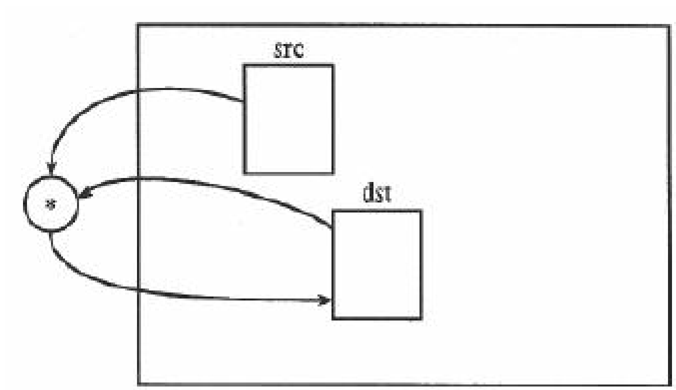
Λειτουργίες σε πλεγματική οθόνη RasterOp

- Πλεγματική οθόνη: εικόνα αποθηκεύεται στη μνήμη οθόνης (frame buffer)
- Πολλές διαδικασίες με μετακίνηση ή συνδυασμούς στη μνήμη οθόνης. (π.χ. επεξεργασία κειμένου)



RasterOp

- Γενική ρουτίνα μετακίνησης/συνδυασμού παραλληλογράμμων τμημάτων εικόνας.
- Βελτιστοποιημένη, γρήγορη υλοποίηση.



Υλοποίηση

```
RasterOp( src_x_min, src_y_min, dst_x_min, dst_y_min, sizex, sizey, funct)
int src_x_min, src_y_min, dst_x_min, dst_y_min, sizex,sizey;
ftype funct)
/* (src_x_min, src_y_min) και (dst_x_min, dst_y_min) */
/* ή κάτω αριστερή γωνία των source και destination */
/* sizex, sizey οι x και y διαστάσεις σε pixel */
/* function κάποιος τελεστής με pixel */
{
int i, j;
for (i= 0; i< sizex; i++)
    for (j= 0; j< sizey; j++)
            set_pixel( dst_x_min+ i, dst_y_min+ j,
            funct( read_pixel( dst_x_min+ i, dst_y_min+ j)
            read_pixel( dst_x_min+ i, dst_y_min + j)));
    } }
}
```

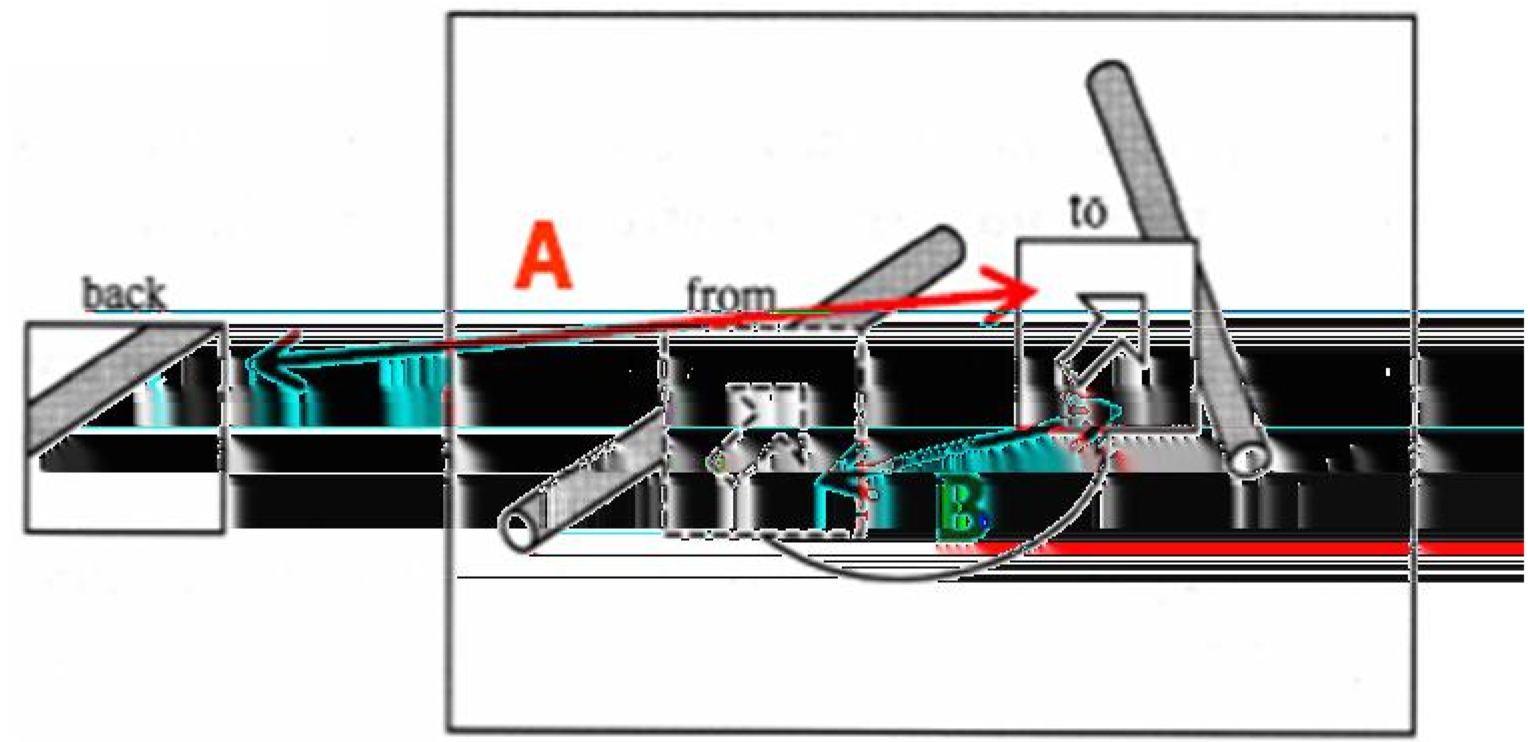
Παράδειγμα: exchange με χρήση XOR

Αντιμετάθεση σχημάτων χωρίς χρήση βοηθητικού χώρου.

```
void exchange (src_ x_ min, src_ y_ min, dst_ x_ min, dst_y_ min,
    sizex, sizey)
int src_ x_ min, src_ y_ min, dst_ x_ min, dst_ y_ min,
sizex, sizey;
{
RasterOp (src_ x_ min, src_ y_ min, dst_ x_ min,
dst_ y_ min, sizex, sizey, XOR);
RasterOp (dst_ x_ min, dst_ y_ min, src_ x_ min,
src_ y_ min, sizex, sizey, XOR);
RasterOp (src_ x_ min, src_ y_ min, dst_ x_ min,
dst_ y_ min, sizex, sizey, XOR);
}
```

RasterOp: Μετακίνηση

Η **exchange μπορεί να χρησιμεύσει για** διαδοχικές μετακινήσεις αντικειμένων (π.χ.cursor)



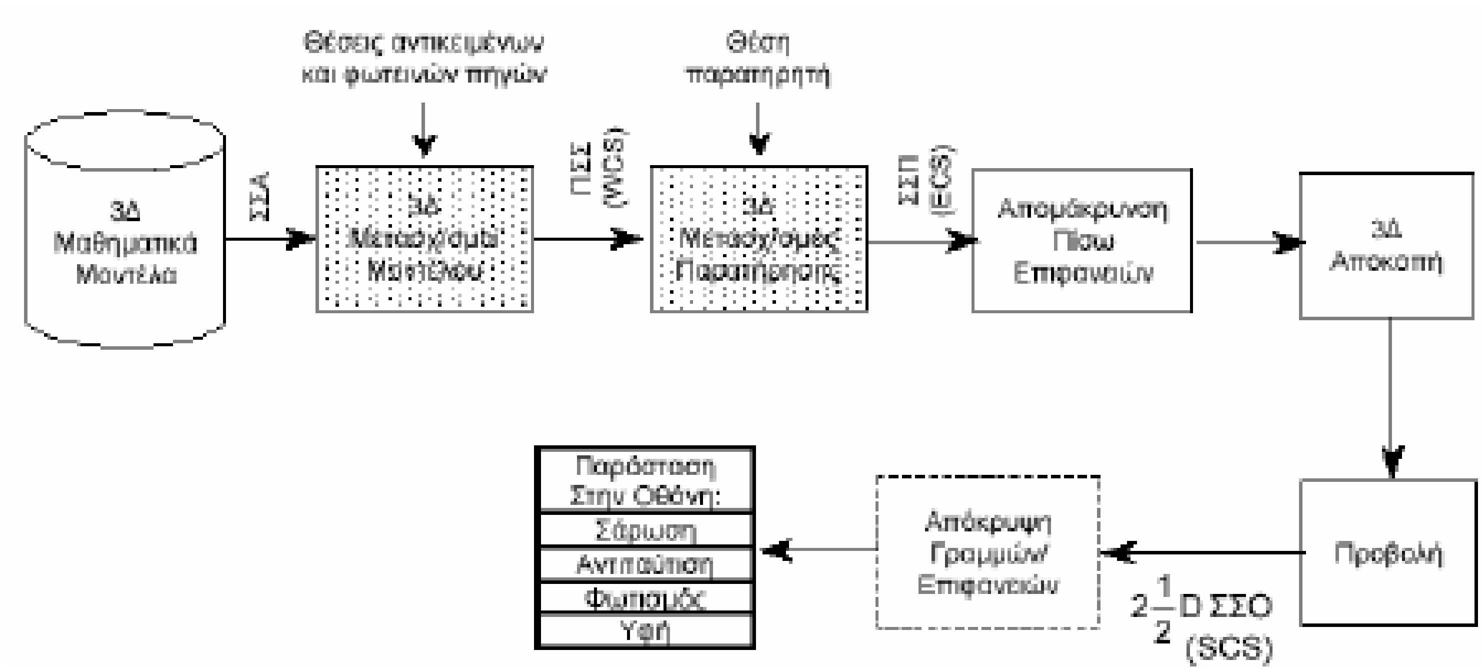
move

```
move (from_x_min, from_y_min, to_x_min,
to_y_min, back_x_min, back_y_min, sizex, sizey)

int from_x_min, from_y_min, to_x_min,to_y_min,
back_x_min, back_y_min, sizex, sizey;
{
exchange (to_x_min, to_y_min,back_x_min,back_y_min, sizex,
    sizey);
exchange (from_x_min, from_y_min,to_x_min, to_y_min, sizex,
    sizey);
}
```

Μετασχηματισμοί 2Δ & 3Δ

- Συνδυασμοί βασικών μετασχηματισμών: μετατόπιση, αλλαγή κλίμακας, περιστροφή, στρέβλωση.



Σημεία και Διανύσματα

- $\mathbf{E}^3$ ο 3Δ Ευκλείδειος χώρος σημείων, $\underline{\mathbf{p}}$ σημείο.
- $\mathbf{R}^3$ ο 3Δ Ευκλείδειος χώρος διανυσμάτων, $\underline{\mathbf{v}}$ διάνυσμα.
- Βασικές ιδιότητες:
  - $\forall\ \underline{\mathbf{p}}_1, \underline{\mathbf{p}}_2 \in \mathbf{E}^3,\ \exists\ \underline{\mathbf{u}} \in \mathbf{R}^3$ με $\underline{\mathbf{u}} = \underline{\mathbf{p}}_1 - \underline{\mathbf{p}}_2$.
  - $\forall\ \underline{\mathbf{p}} \in \mathbf{E}^3,\ \exists\ \underline{\mathbf{u}} \in \mathbf{R}^3,\ \underline{\mathbf{p}} + \underline{\mathbf{u}} = \underline{\mathbf{q}} \in \mathbf{E}^3$
  - $\forall\ \underline{\mathbf{p}}, \underline{\mathbf{q}}, \underline{\mathbf{r}} \in \mathbf{E}^3,\ (\underline{\mathbf{p}} - \underline{\mathbf{q}}) + (\underline{\mathbf{q}} - \underline{\mathbf{r}}) = (\underline{\mathbf{p}} - \underline{\mathbf{r}})$

Διανυσματικοί χώροι

- Σε κάθε διανυσματικό χώρο Δ (π.χ. $\mathbf{R}^3$) ορίζονται 2 πράξεις
  - Διανυσματική πρόσθεση ($\underline{q} + \underline{v}$)
  - Βαθμωτός (scalar) πολλαπλασιασμός ($\lambda\underline{v}$)

- Ιδιότητες πρόσθεσης:
  - Αντιμεταθετική: $\underline{q} + \underline{v} = \underline{v} + \underline{q}$
  - Προσεταιριστική: $(\underline{p} + \underline{q}) + \underline{v} = \underline{p} + (\underline{q} + \underline{v})$
  - Ύπαρξη ουδετέρου: $\exists\ \underline{0} \in \Delta$: $\underline{0} + \underline{v} = \underline{v} + \underline{0} = \underline{v}$
  - Ύπαρξη αντιθέτου: $\forall\ \underline{p} \in \Delta\ \exists\ -\underline{p} \in \Delta$: $\underline{p} + (-\underline{p}) = \underline{0}$

Διανυσματικοί χώροι

- Ιδιότητες βαθμωτού πολλαπλασιασμού (β.π.)
  - Επιμερισμός ως προς την πρόσθεση:

    $\lambda(\underline{v} + \underline{q}) = \lambda\underline{v} + \lambda\underline{q}$
  - Επιμερισμός πρόσθεσης ως προς βαθμωτό πολλαπλασιασμό :

    $(\lambda + \mu)\ \underline{v} = \lambda\underline{v} + \mu\underline{v}$
  - Προσεταιρισμός:

    $(\lambda\mu)\ \underline{v} = \lambda\ (\mu\ \underline{v})$
  - Ουδέτερο στοιχείο:

    $\exists\ \underline{1} \in \Delta\text{:}\ \underline{v}\bullet\underline{1} = \underline{1}\bullet\underline{v} = \underline{v}$

Διανυσματικοί χώροι

- Παραδείγματα:
  - 2Δ και 3Δ Ευκλείδειοι χώροι.
  - Πολυώνυμα διαφόρων βαθμών.
- Γραμμικός συνδυασμός $\underline{x}_1, \underline{x}_2, \ldots \underline{x}_\nu \in Δ$:

$$y = \lambda_1 \underline{x}_1 + \lambda_2 \underline{x}_2 + \ldots + \lambda_\nu \underline{x}_\nu$$

- Γραμμική ανεξαρτησία $\underline{x}_1, \underline{x}_2, \underline{x}_\nu, \ldots \in Δ$:

$$\text{αν } \lambda_1 \underline{x}_1 + \lambda_2 \underline{x}_2 + \ldots + \lambda_\nu \underline{x}_\nu = 0,$$

έχει μόνη λύση την

$$\lambda_1 = \lambda_2 = \ldots = \lambda_\nu = 0$$

Παράδειγμα τα (1,0,0), (0,1,0), (0,0,1) του $\mathbf{E}^3$

Διανυσματικοί χώροι

- Βάση: κάθε ολοκληρωμένο σύνολο γραμμικώς ανεξαρτήτων διανυσμάτων.
  - Το πλήθος τους καλείται διάσταση του χώρου.
  - Αν y = $\lambda_1 \underline{i} + \lambda_2 \underline{j} + \lambda_3 \underline{k}$ όπου (i, j, k) βάση τότε $(\lambda_1, \lambda_2, \lambda_3)$ συντεταγμένες.
  - ($\underline{O}$, $\underline{i}$, $\underline{j}$, $\underline{k}$) ονομάζεται σύστημα συντεταγμένων με βάση το σημείο $\underline{O}$, και βάση ($\underline{i}$, $\underline{j}$, $\underline{k}$)
  - ($\underline{i}$, $\underline{j}$, $\underline{k}$) ορίζουν άξονες συντεταγμένων
- Μήκος: διανύσματος (x,y,z)

$$|\underline{v}| = \sqrt{x^2 + y^2 + z^2}$$

Διανυσματικοί χώροι

- Απόσταση μεταξύ δύο σημείων p_1, p_2

$$\left|\underline{p}_1 - \underline{p}_2\right| = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

Διανυσματικοί χώροι

- Εσωτερικό γινόμενο (πραγματικός αριθμός)

  $\underline{v} \bullet \underline{w} = \Sigma\,(v_i w_i)$

  π.χ. Ευκλείδειος 3Δ: $\underline{v} \bullet \underline{w} = v_x w_x + v_y w_y + v_z w_z$

Ιδιότητες:

- Αντιμεταθετική: $\underline{v} \bullet \underline{w} = \underline{w} \bullet \underline{v}$
- $\underline{v} \bullet \underline{v} = 0 \Rightarrow \underline{v} = 0$
- $\underline{v} \,/\, |\underline{v}| = \underline{v}'$, μοναδιαίο διάνυσμα (κανονικοποίηση)
- ισχύει $\underline{v} \bullet \underline{w} = |\underline{v}|\ |\underline{w}| \cos(\theta)$, απ' όπου μπορεί να υπολογιστεί και η γωνία μεταξύ διανυσμάτων.

Διανυσματικοί χώροι

- Εξωτερικό γινόμενο (διάνυσμα)

  $\underline{v} \times \underline{w} = (v_y w_z + v_z w_y)\, \underline{i} + (v_x w_z + v_z w_x)\underline{j}$
  $\qquad + (v_x w_y + v_y w_x)\underline{k}$

- $\underline{v} \times \underline{w}$ είναι κάθετο στο επίπεδο που ορίζουν τα δύο άλλα διανύσματα.
- δεν ισχύει η αντιμεταθετική ιδιότητα.

  $\underline{v} \times \underline{w} = - \underline{w} \times \underline{v}$

Συσχετισμένοι (Affine) Μετασχηματισμοί

- Συσχετισμένος ή βαρυκεντρικός μετασχηματισμός σημείων $p_0 \ldots p_n \in \mathbf{E^3}$:

$$p = \Sigma\, \alpha_j p_j \text{ όπου } \quad \alpha_j \in \mathbf{R} \quad \text{και} \quad \Sigma\, \alpha_j = 1$$

  - αποτέλεσμα $\underline{p} \in \mathbf{E^3}$.
  - {αj} ονομάζονται οι συσχετισμένες συντεταγμένες του $\underline{p}$ ως προς $\underline{p}_0 \ldots \underline{p}_n$.
  - Ο μετασχηματισμός είναι κυρτός αν $\alpha_j \geq 0$

- Συσχετισμένος μετασχηματισμός Φ: $\mathbf{E^3} \rightarrow \mathbf{E^3}$ που αφήνει τους συσχετισμένους μετασχηματισμούς αναλλοίωτους:

$$\Phi(p) = \Sigma\, \alpha_j\, \Phi(\underline{p}_j)$$

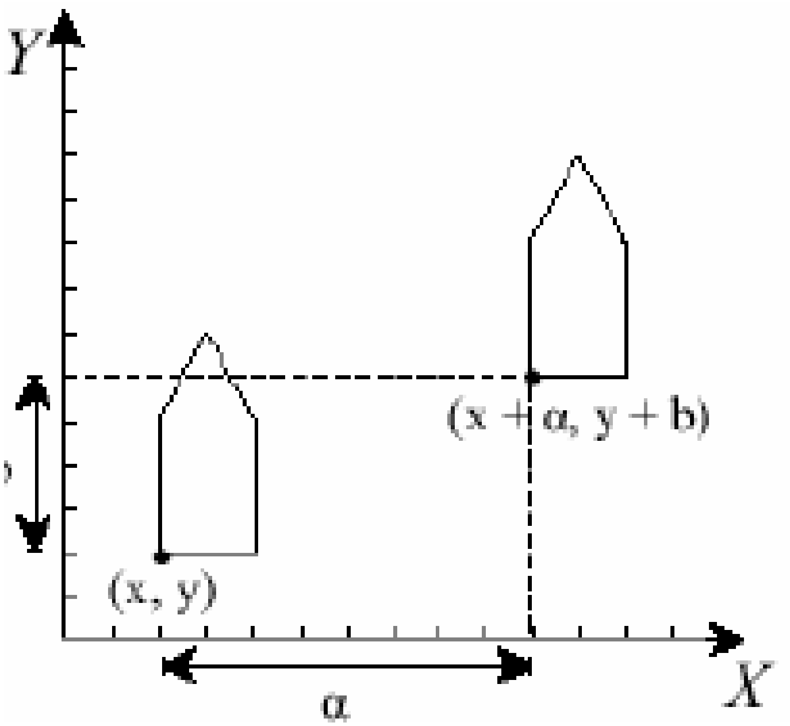
Συσχετισμένοι Μετασχηματισμοί

- π.χ. εφαρμογή συσχετισμένου μετασχηματισμού πάνω σε ευθύγραμμο τμήμα s απεικονίζει το μέσον του στο μέσο της συσχετισμένης εικόνας Φ(s).
- Συσχετισμένος μετασχηματισμός με μορφή πίνακα: $\Phi(\underline{p}) = \underline{A}\cdot\underline{p} + \underline{t}$ ($\underline{A}$ πίνακας, 3x3 για το $\mathbf{E^3}$ **)**
- Απόδειξη:

$$
\begin{aligned}
\Phi(\Sigma\ \alpha j p j\ ) &= \underline{A}(\Sigma\ \alpha_j \underline{p}_j) + \underline{t} \\
&= \Sigma\ \underline{A}\alpha_j \underline{p}_j + \Sigma\ \alpha_j \underline{t} \\
&= \Sigma\ \alpha_j\ (\underline{A}\ \underline{p}_j + \underline{t}) \\
&= \Sigma\ \alpha_j\ \Phi(\underline{p})
\end{aligned}
$$

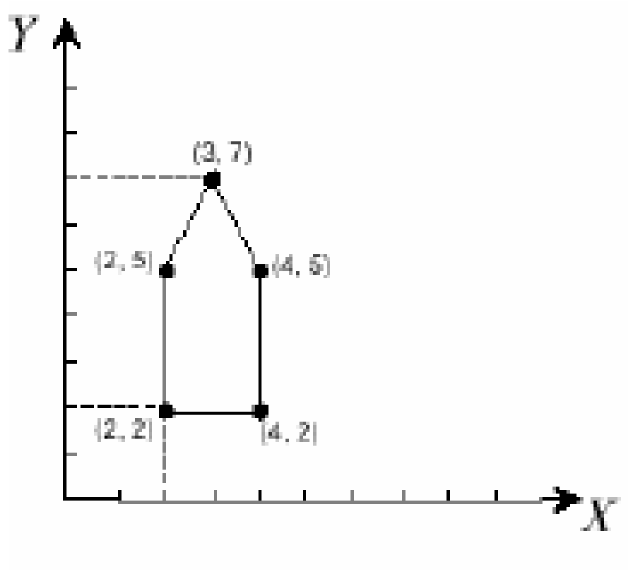
2D Μετασχηματισμοί

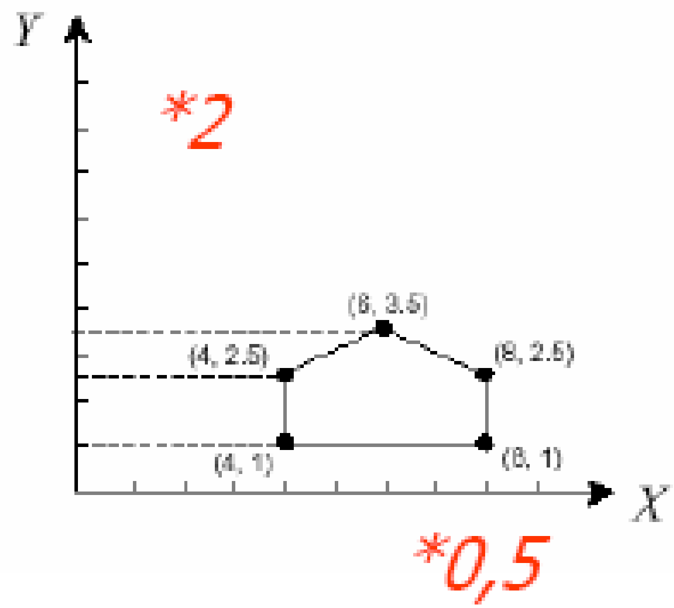
- Μετατόπιση: x'=x+a, y'=y+b



Μετασχηματισμοί

- Αλλαγή κλίμακας (διαφορετική ανά άξονα):
  $x' = x * a, \quad y' = y * b$





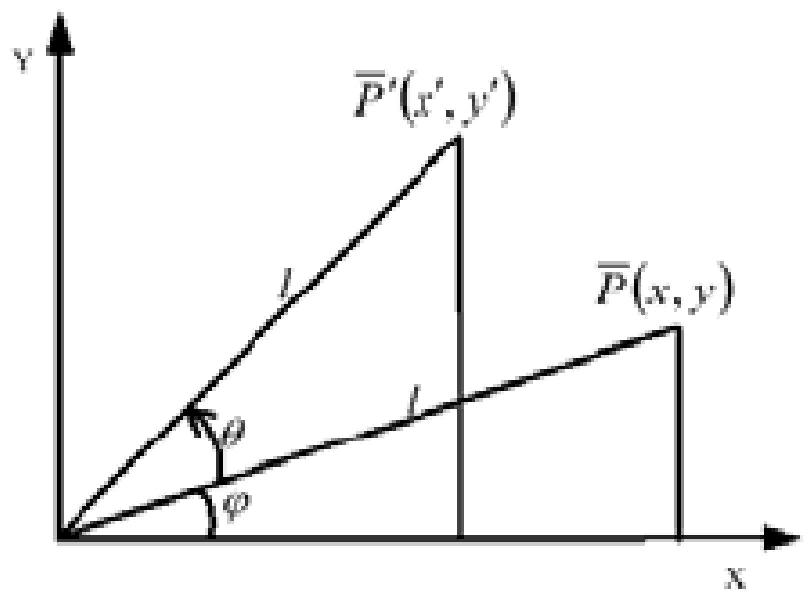
Μετασχηματισμοί

Στροφή κατά γωνία θ

(+ve αντίθετα από δείκτες ρολογιού)

$$x' = l\cos(\varphi + \theta) = l(\cos\varphi\cos\theta - \sin\varphi\sin\theta) = x\cos\theta - y\sin\theta$$

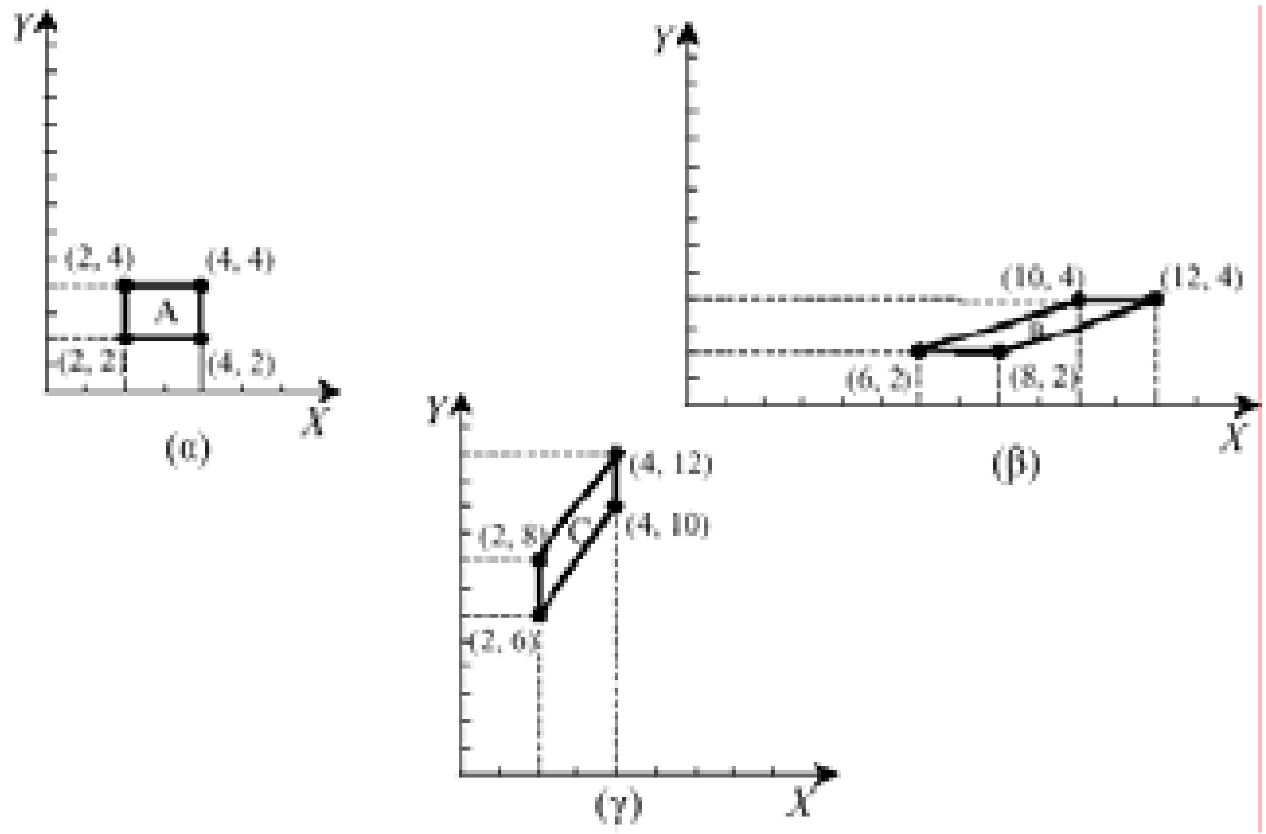
$$y' = l\sin(\varphi + \theta) = l(\cos\varphi\sin\theta + \sin\varphi\cos\theta) = x\sin\theta + y\cos\theta$$



Μετασχηματισμοί

- Στρέβλωση κατά άξονα x και y με παράγοντα 2:

$$x' = x + 2y, \quad y' = y + 2x$$



Μετασχηματισμοί

Οποιοσδήποτε άλλος μετασχηματισμός, δημιουργείται σαν συνδυασμός των τεσσάρων βασικών μετασχηματισμών :

- **M** μετατόπιση,
- **R** στροφή,
- **H** στρέβλωση,
- **S** αλλαγή κλίμακος.

Μετασχηματισμοί: άσκηση

- Στρέβλωση (shearing), Να βρεθούν οι σχέσεις μετασχηματισμούκατά x και y με z σταθερό:

<u>Απάντηση</u>

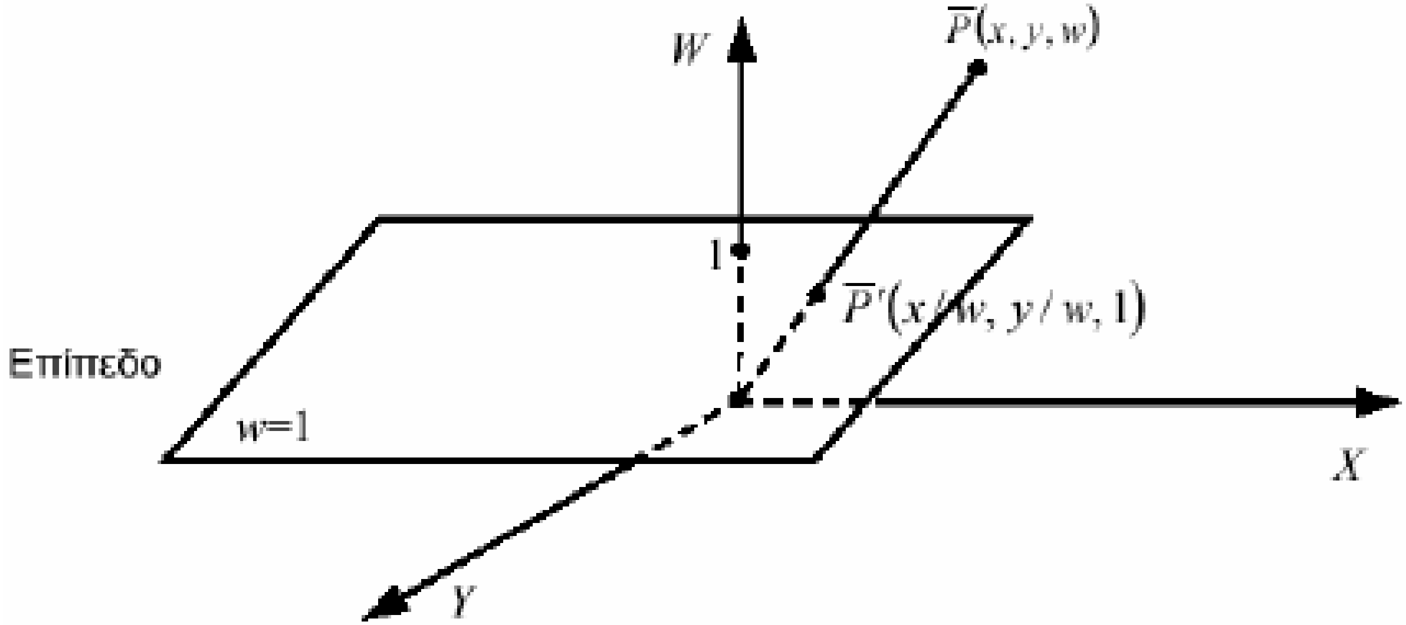
$SH(\underline{p}) = H_{x,y} \cdot \underline{p}$, όπου:

$$H_{x,y} = \begin{bmatrix} 1 & a & b \\ c & 1 & d \\ 0 & 0 & 1 \end{bmatrix}$$

Ομογενείς συντεταγμένες

- Προβλήματα:
  - Μεταφορά δεν υλοποιείται με πολλαπλασιασμό πινάκων.
  - Ύπαρξη σταθερού σημείου O , σταθερού για όλους τους μετασχηματισμούς.
- Ομογενείς συντεταγμένες: $(x,y) \rightarrow (x,y,w)$ με $w \neq 0$.
  - (x,y,w) παριστάνει το $(x/w,y/w)$
  - Άπειρες τριάδες για κάθε σημείο.
  - Βασική παράσταση: $(x,y,1)$

Ομογενείς συντεταγμένες



Ομογενείς Συντεταγμένες

- Συσχετισμένοι μετασχηματισμοί: πίνακες 3x3
- **Μεταφορά:**

$$\begin{bmatrix} 1 & 0 & d_x \\ 0 & 1 & d_y \\ 0 & 0 & 1 \end{bmatrix}$$

Ομογενείς Συντεταγμένες

- Αλλαγή κλίμακας (αν si< 1 σμίκρυνση)

$$\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Ομογενείς συντεταγμένες

- **Στροφή**

$$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Ομογενείς συντεταγμένες

- Στρέβλωση (κατά άξονες x και y):

$$\underline{H_x} = \begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad \underline{H_y} = \begin{bmatrix} 1 & 0 & 0 \\ b & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Σύνθεση μετασχηματισμών

- Σύνθεση μετασχηματισμών:

  π.χ. Αλλαγή κλίμακας, ως προς $\underline{C}=(\underline{c}x,\underline{c}y,1)$
  - μεταφορά κατά $-\underline{c}= -(\underline{O} - \underline{C})$
  - αλλαγή κλίμακας
  - μεταφορά κατά $\underline{c}$
- Γενικά δεν ισχύει η αντιμετάθεση. Πρώτος εκτελείται ο τελεστής που γράφεται τελευταίος.
- Σύνθεση: Εκτέλεση σειράς μετασχηματισμών από ένα τελικό πίνακα.
- Αυξάνει την απόδοση (μειώνει την πολυπλοκότητα)

Σύνθεση μετασχηματισμών

Ιδιότητες:

- $T(x_1,y_1) \cdot T(x_2,y_2) = T(x_2,y_2) \cdot T(x_1,y_1)$
  $= T(x1 + x2, y_1 + y_2)$
- $S(s_{x1},s_{y1}) \cdot S(s_{x2},s_{y2}) = S(s_{x2},s_{y2}) \cdot S(s_{x1},s_{y1})$
  $= S(s_{x1}s_{x2}, s_{y1}s_{y2})$
- $R(\theta_1) \cdot R(\theta_2) = R(\theta_2) \cdot R(\theta_1) = R(\theta_1 + \theta_2)$
- $S(s_x,s_y) \cdot R(\theta) = R(\theta) \cdot S(s_x,s_y)$
  μόνο αν $s_x = s_y$

Γεωμετρικές Ιδιότητες

- Για κάθε συσχετισμένο μετασχηματισμό F και σημεία <u>p</u> και <u>q</u> ισχύει:

  $F(\lambda\underline{p}+(1-\lambda)\underline{q}) = \lambda F(\underline{p})+ (1-\lambda)\, F(\underline{q})$ για $0 \le \lambda \le 1$

  - $\lambda\underline{p}+(1-\lambda)\underline{q}$ είναι το ευθύγραμμο τμήμα μεταξύ <u>p</u> και <u>q</u>.
  - Η F παράγει πάλι ευθύγραμμο τμήμα.
  - Η σχέση $\lambda/(1-\lambda)$ παραμένει αναλλοίωτη από F.
  - Εξακολουθεί να αρκεί απεικόνιση άκρων μόνο.
  - Παράλληλες ευθείες παραμένουν παράλληλες.
  - π.χ. οι μετασχηματισμοί που είδαμε νωρίτερα.

Γεωμετρικές ιδιότητες

- Ορθογώνιος μετασχηματισμός:

  Μετασχηματισμός Α λέγεται ορθογώνιος αν

  – $|\underline{\alpha}_1| = |\underline{\alpha}_2| = 1$
  – $\underline{\alpha}_1 \cdot \underline{\alpha}_2 = 0$
  – Ορίζουσα του $\underline{A} = 1$

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

Γεωμετρικές ιδιότητες

$$\underline{M} = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

- Ο <u>Μ</u> είναι μετασχηματισμός ομοιότητας αν ο <u>Α</u> είναι ορθογώνιος.
- Ένας μετασχηματισμός ομοιότητας διατηρεί αναλλοίωτα μήκη και γωνίες.
- Οποιαδήποτε σύνθεση T & R είναι μετασχηματισμός ομοιότητας.
- Αν στη σύνθεση έχουμε S και H, έχουμε συσχετισμένο μετασχηματισμό αλλά όχι ομοιότητας (διατηρείται παραλληλία αλλά όχι μήκη και γωνίες).

Άσκηση

- Ζητείται να υπολογιστεί ο πίνακας μετασχηματισμού της συμμετρικής απεικόνισης εικόνας ως προς τον άξονα w ο οποίος σχηματίζει γωνία θ με τον θετικό άξονα των Χ
- **Βοήθεια:** Ο μετασχηματισμός μπορεί να

προκύψει από :

  – στροφή της εικόνας κατά –θ
  – Συμμετρία ως προς τον άξονα Χ
  – Στροφή της εικόνας κατά θ

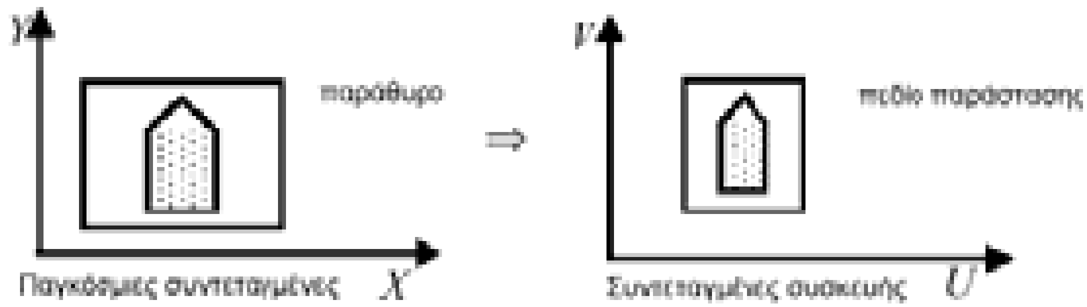
Φροντιστηριακή άσκηση #4

1. Να μελετήσετε και να περιγράψετε την τρέχουσα τεχνολογία καρτών γραφικών
2. Δίδεται σημείο Α (χ=3,ψ=2). Ζητείται να βρεθεί το σημείο Α’ που προκύπτει από την περιστροφή του σημείου Α ως προς σημείο Β με συντεταγμένες Β (Χ=1, ψ=1) κατά γωνία θ=30

2-Δ Μετασχηματισμός Παρατήρησης

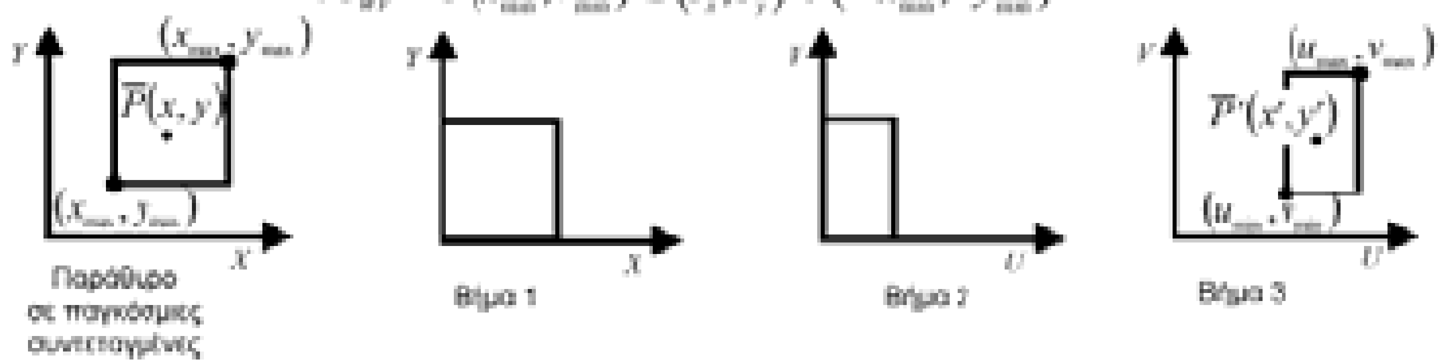
Δημιουργία εικόνας στο ΠΣΣ (WCS), απεικόνιση στο ΣΣΟ (PDC)

- Χρήστης ορίζει παράθυρο WCS και πεδίο παρατήρησης PDC



Υπολογισμός M_{wv} από $(x_{min}, y_{min}),(x_{max}, y_{max}),(u_{min}, v_{min}),(u_{max}, v_{max})$

- $T(-x_{min}, -y_{min})$
- $S(s_x, s_y)$ με $s_x = \dfrac{u_{max} - u_{min}}{x_{max} - x_{min}}$ $\quad s_y = \dfrac{v_{max} - v_{min}}{y_{max} - y_{min}}$
- $T(u_{min}, v_{min})$
- $M_{wv} = T(u_{min}, v_{min}) \cdot S(s_x, s_y) \cdot T(-x_{min}, -y_{min})$



2-Δ Μετασχηματισμός Παρατήρησης

- $$M_{wv} = \begin{bmatrix} \dfrac{u_{max}-u_{min}}{x_{max}-x_{min}} & 0 & u_{min} - \dfrac{u_{max}-u_{min}}{x_{max}-x_{min}} x_{min} \\ 0 & \dfrac{v_{max}-v_{min}}{y_{max}-y_{min}} & v_{min} - \dfrac{v_{max}-v_{min}}{y_{max}-y_{min}} y_{min} \\ 0 & 0 & 1 \end{bmatrix}$$

- Αλλοιώσεις σχημάτων αν $s_x \neq s_y$ ή $a_w \neq a_v$

  με $a_w = \dfrac{w_{dx}}{w_{dy}}$, $a_v = \dfrac{v_{dx}}{v_{dy}}$ όπου $w_{dx} = x_{max} - x_{min}$ $w_{dy} = y_{max} - y_{min}$

  $v_{dx} = u_{max} - u_{min}$ $v_{dy} = v_{max} - v_{min}$

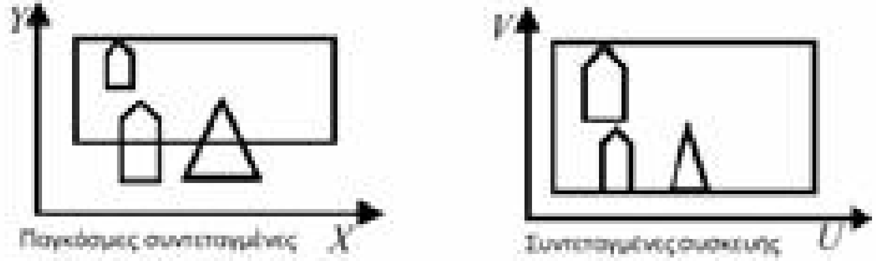
- Διόρθωση παραμόρφωσης με μείωση πεδίου παράστασης

  if $a_v > a_w$ *then* $v_{dx} = v_{dy} * w_{dx} / w_{dy}$

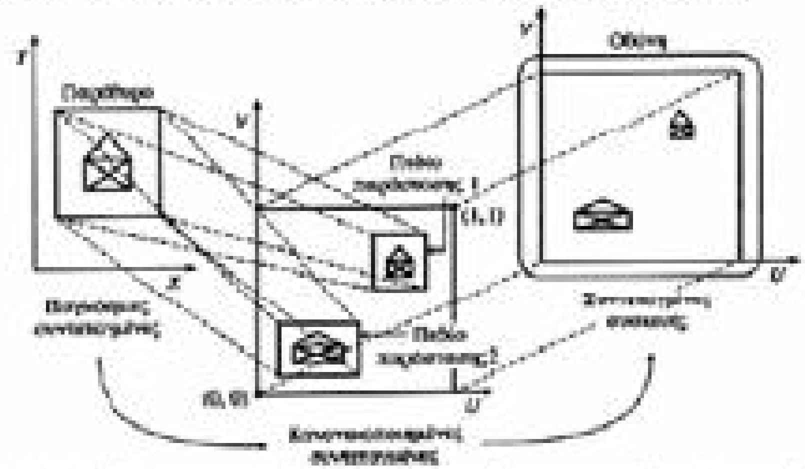
  else if $1/a_v > 1/a_w$ *then* $v_{dy} = v_{dx} * w_{dy} / w_{dx}$

2-Δ Μετασχηματισμός Παρατήρησης

- Συνδυασμός με αποκοπή

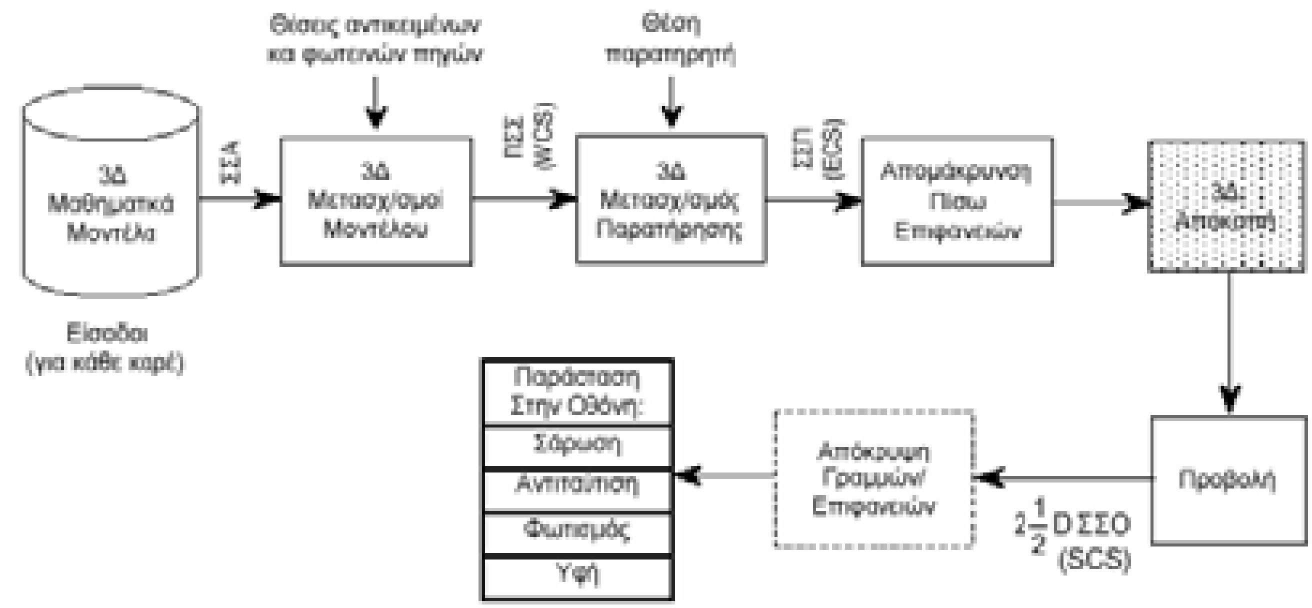


- Πολλαπλές συσκευές εξόδου: χρήση κανονικοποιημένων συντεταγμένων συσκευής (NDC) [0,1] x [0,1]
  - WCS ➔ NDC & NDC ➔ {PDC1, PDC2 … } (οδηγοί συσκευών)
  - NDC ➔ PDC είναι ομοιόμορφος μετασχηματισμός



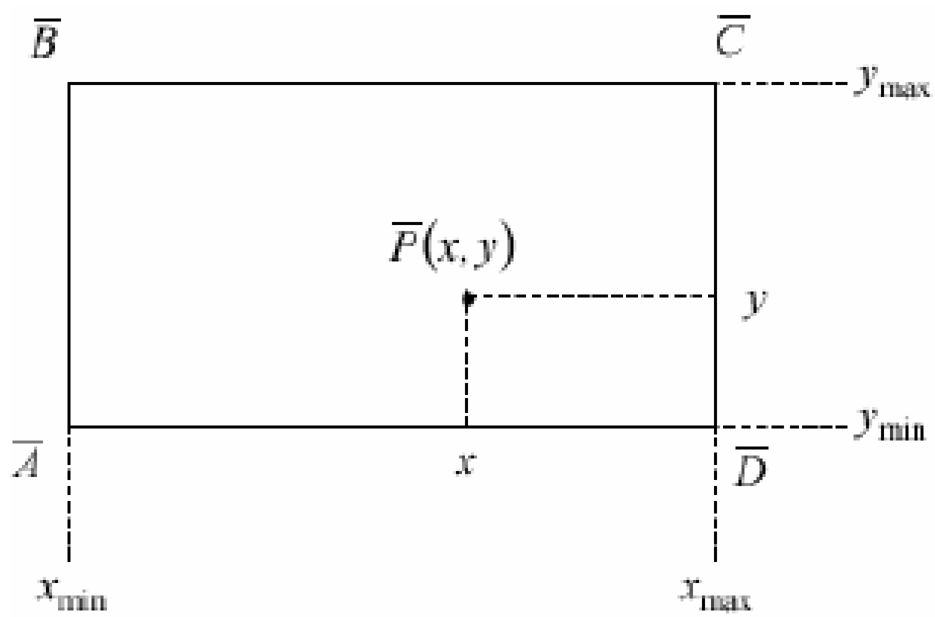
Αποκοπή / clipping

- Αποκοπή αντικειμένου (π.χ. πολυγώνου, πυραμίδας, κύβου) ω.π. σημείο αποκοπής:
  - Αποφυγή αντεστραμμένης εμφάνισης αντικειμένων (π.χ. 3-Δ, όπισθεν παρατηρητή).
  - Μείωση του όγκου δεδομένων (φίλτρο).



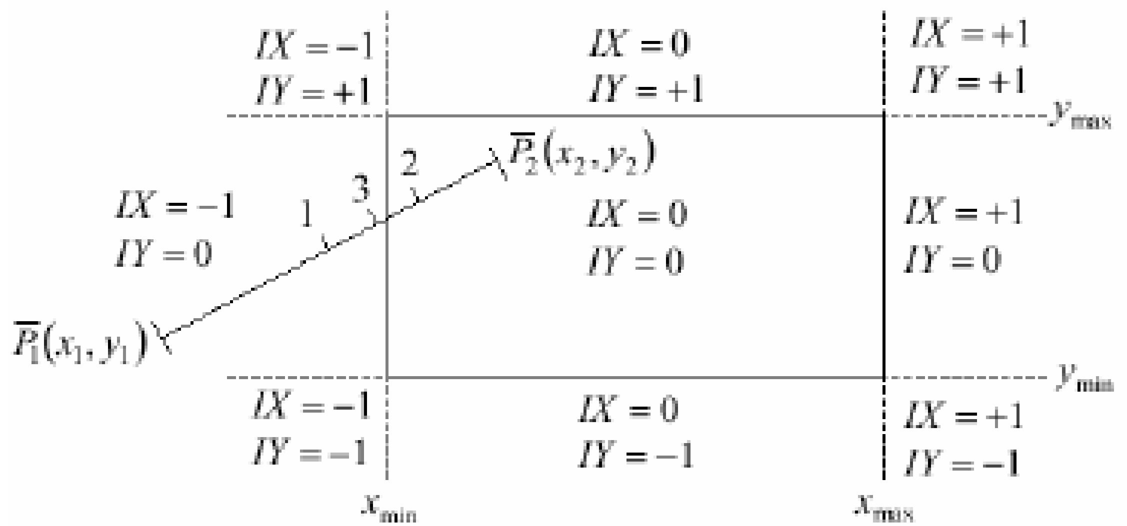
Αποκοπή

- 2-Δ: στην αντιταύτιση, απόκρυψη, παράλληλη επεξεργασία, 2Δ πακέτα.
- 3-Δ: γραφική σωλήνωση (γενίκευση 2-Δ αποκοπής)
- 2-Δ αποκοπή σημείου: εντός εάν $x_{min} \leq x \leq x_{max}$ και $y_{min} \leq y \leq y_{max}$.



Αποκοπή

- Ευθύγραμμα τμήματα: Αλγόριθμος μέσου.
- Διαδοχική διάσπαση ευθυγράμμου τμήματος στη μέση (ολίσθηση).
- Χρήση «φθηνών» ελέγχων εντός/ εκτός.
- Κωδικοποίηση 9 περιοχών του επιπέδου
- Τέλος εάν $\underline{p_1 p_2}$ < pixel.

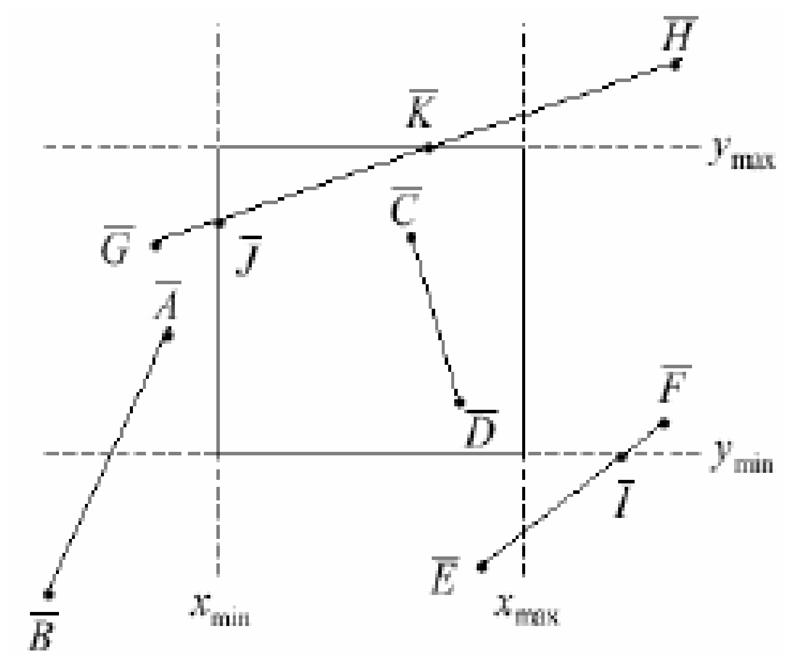


Αλγόριθμος Μέσου

```
typedef struct {float x, y} point;
midpoint (P1, P2, xmin, xmax, ymin, ymax)
point P1, P2; float xmin, xmax, ymin, ymax;
{ point M;
/* Υπολογισμός IX 1, IY 1, IX 2, IY 2 */
if (( IX1== 0)&&( IY1== 0)&&( IX2== 0)&&( IY2== 0))
    /* Το P1P2 είναι εντός παραθύρου */
else if (( IX1== IX2)&&( IX1!= 0))||(( IY1== IY2)&&(IY1!= 0))
    /* Το P1P2 είναι εκτός παραθύρου */
else {
    M. x=( P1. x+ P2.x)/ 2;
    M. y=( P1. y+ P2.y)/ 2;
    midpoint (P1,M, xmin, xmax, ymin, ymax);
    midpoint (M, P2, xmin, xmax, ymin, ymax);
    }
}
```

Αποκοπή ευθυγράμμων τμημάτων

- Αλγόριθμος **Cohen-Sutherland:**
  - Αρχικά «φθηνός» έλεγχος για τις απλές περιπτώσεις.
  - π.χ. AB έξω, CD μέσα.
  - Για άλλα ευθύγραμμα τμήματα, κόψιμο με ευθεία παραθύρου και αναδρομή.



Αλγόριθμος Cohen-Sutherland

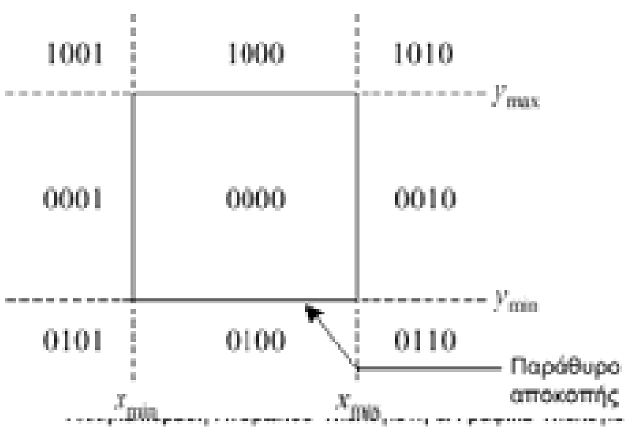
- Αντιστοίχηση κωδικών σε 9 περιοχές του επιπέδου.
- Κώδικες προκύπτουν από τα πρόσημα διαφορών (π.χ. 1ο bit $y_{max} – y$).

  Πρώτο bit 1, περιοχή πάνω από την ευθεία $y = y_{max}$

  Δεύτερο bit 1, περιοχή κάτω από την ευθεία $y = y_{min}$

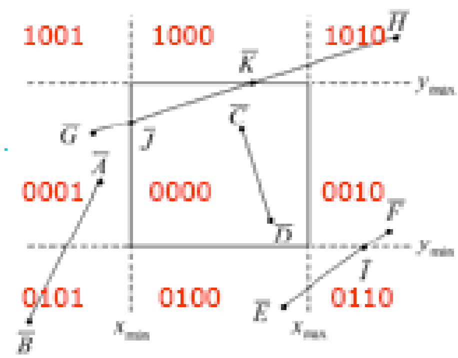
  Τρίτο bit 1, περιοχή δεξιά από την ευθεία $x = x_{max}$

  Τέταρτο bit 1, περιοχή αριστερά από την ευθεία $x = x_{min}$



Αλγόριθμος Cohen-Sutherland

- Υπολόγισε κώδικες c_1, c_2 για $\underline{p_1 p_2}$.
- Αν $c_1 \vee c_2 = 0000$, $\underline{p_1 p_2}$ είναι εντός (π.χ. CD).
- Αν $c_1 \wedge c_2 = 0000$, $\underline{p_1 p_2}$ είναι εκτός(π.χ. AB).
- Διαφορετικά:
  - Εύρεση ευθείας του παραθύρου που αντιστοιχεί σε bit με διαφορετικές τιμές.
  - τομή $\underline{p_1 p_2}$ με ευθεία.
  - Αναδρομική κλήση για το ένα από τα τμήματα ως προς την ευθεία τμήμα (εκείνο με bit=0)

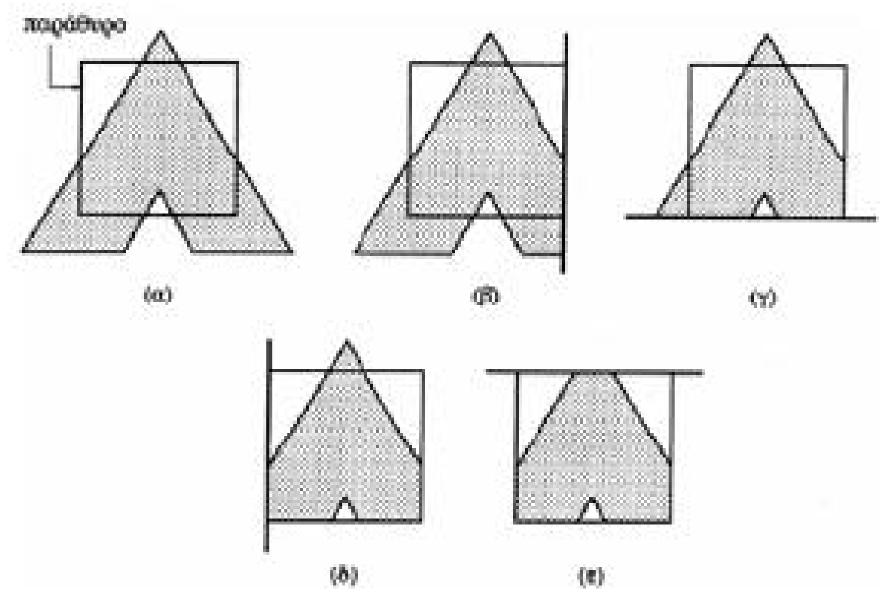


Αλγόριθμος Cohen-Sutherland

```
void CS (P1,P2, xmin, xmax, ymin, ymax)
point P1, P2; float xmin, xmax, ymin, ymax;
{
int c1, c2; point I;
c1= code( P1); /* Εύρεση κώδικα P1*/
c2= code( P2); /* Εύρεση κώδικα P2*/
if (( c1| c2)== 0) /* Το P1P2 είναι εντός παραθύρου */
else if (( c1& c2)!= 0) /* Το P1P2 είναι εκτός παραθύρου */
else {
      intersect (P1,P2,I,xmin,xmax,ymin,ymax);
      if exoteriko(P1)
              CS( I,P2,xmin,xmax,ymin,ymax);
      else
              CS( P1, I, xmin, xmax, ymin, ymax);
}
}
```

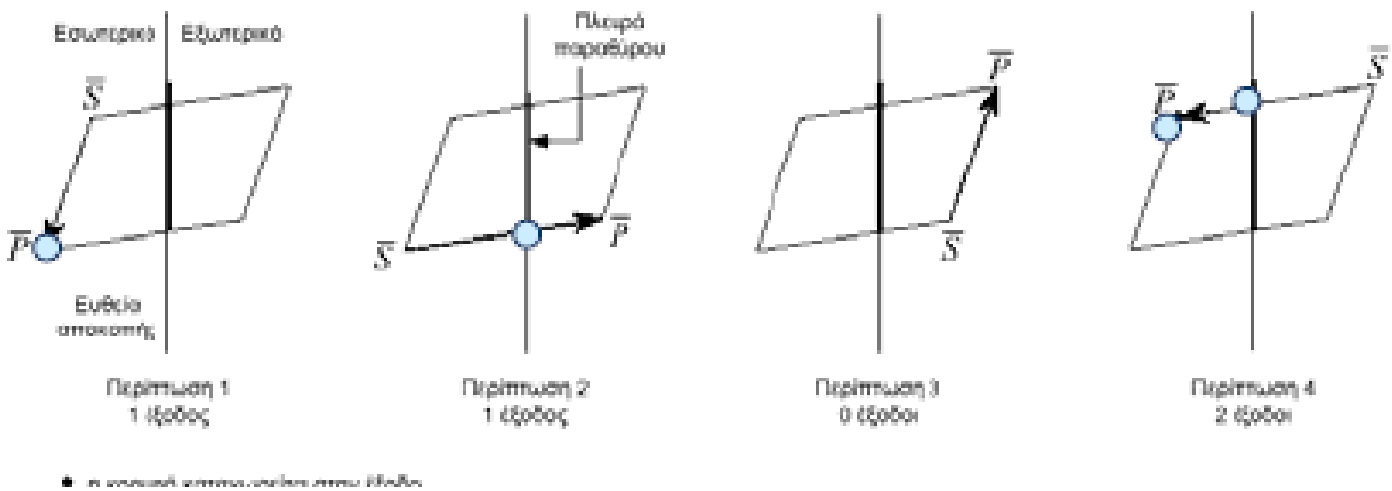
Αλγόριθμος Sutherland-Hodgman
2-Δ αποκοπής πολυγώνων

- Κατάλληλος για αποκοπή τυχαίου πολυγώνου με κυρτό πολύγωνο (παράθυρο) αποκοπής.
- m βήματα για m πλευρές παραθύρου αποκοπής.
- είσοδος στο βήμα i (i = 1 … m) μετά από αποκοπή με πλευρά i-1.



Αλγόριθμος Sutherland-Hodgman 2-Δ αποκοπής πολυγώνων

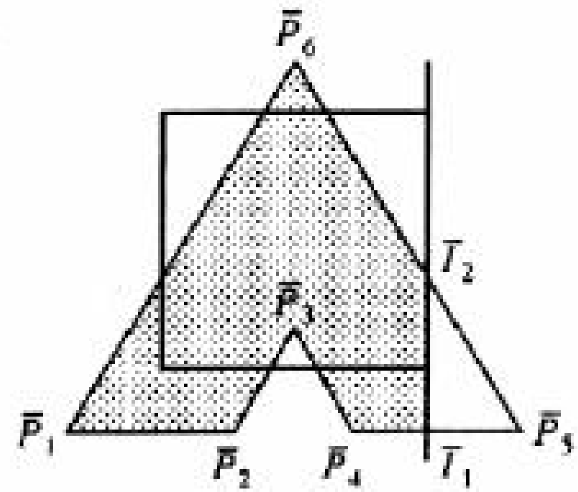
- πολύγωνο ορίζεται από τις κορυφές του p_1 , p_2 … p_n με θετική μαθηματική φορά.
- Βήμα i εξετάζει σχέση πλευράς SP με ακμή παραθύρου i.



Αλγόριθμος Sutherland-Hodgman 2-Δ Αποκοπής Πολυγώνων

- Παράδειγμα βήματος Sutherland-Hodgman:

$\bar{S}$	$\bar{P}$	Περίπτωση	Αποτελέσματα
$\bar{P}_1$	$\bar{P}_2$	1	$\bar{P}_2$
$\bar{P}_2$	$\bar{P}_3$	1	$\bar{P}_3$
$\bar{P}_3$	$\bar{P}_4$	1	$\bar{P}_4$
$\bar{P}_4$	$\bar{P}_5$	2	$\bar{I}_1$
$\bar{P}_5$	$\bar{P}_6$	4	$\bar{I}_2 \bar{P}_6$
$\bar{P}_6$	$\bar{P}_1$	1	$\bar{P}_1$



Αποκοπή σε 3-Δ

- Αντικείμενο αποκοπής:
  - Περιορισμένη πυραμίδα (προοπτική)
  - Κύβος (παράλληλη)
  - 6 επίπεδα αποκοπής γι αυτά τα αντικείμενα.

Cohen – Sutherland 3-Δ

- 6 – bit κωδικοί για κάθε άκρο p(x, y, z)
- Έστω κύβος αποκοπής:
  - πρώτο bit = 1, $\Leftrightarrow z_p > z_{max}$, σημείο πίσω από κύβο.
  - δεύτερο bit = 1, $\Leftrightarrow z_p < z_{min}$, σημείο εμπρός από κύβο
  - τρίτο bit = 1, $\Leftrightarrow y_p > y_{max}$, σημείο πάνω από κύβο
  - τέταρτο bit = 1, $\Leftrightarrow y_p < y_{min}$, σημείο κάτω από κύβο
  - πέμπτο bit = 1, $\Leftrightarrow x_p > x_{max}$, σημείο δεξιά από κύβο
  - έκτο bit = 1, $\Leftrightarrow x_p < x_{min}$, σημείο αριστερά από κύβο

Αλγόριθμος Cohen – Sutherland 3-Δ

- Αν $C_1 \vee C_2 = 000000$, $\underline{p_1 p_2}$ εντός.
- Αν $C_1 \wedge C_2 \neq 000000$, $\underline{p_1 p_2}$ εκτός.
- Διαφορετικά:
  - Εύρεση επιφάνειας κύβου που αντιστοιχεί σε bit με διαφορετικές τιμές.
  - τομή $\underline{p_1 p_2}$ με επιφάνεια
  - Αναδρομική κλήση για το εσωτερικό τμήμα ως προς επιφάνεια..

Αλγόριθμος Cohen – Sutherland 3-Δ

- Τομή ευθυγράμμου τμήματος p1p2 με επίπεδο βάσει της παραμετρικής εξίσωσης:

  $x(t) = x1 + t(x2 – x1) = x1 + t\ \Delta x$

  $y(t) = y1 + t(y2 – y1) = y1 + t\ \Delta y$

  $z(t) = z1 + t(z2 – z1) = z1 + t\ \Delta z$

- π.χ. τομή με $y = Y$

  $Y = y(t) = y_1 + t\ \Delta t \Rightarrow t = (Y – y_1) / \Delta y$

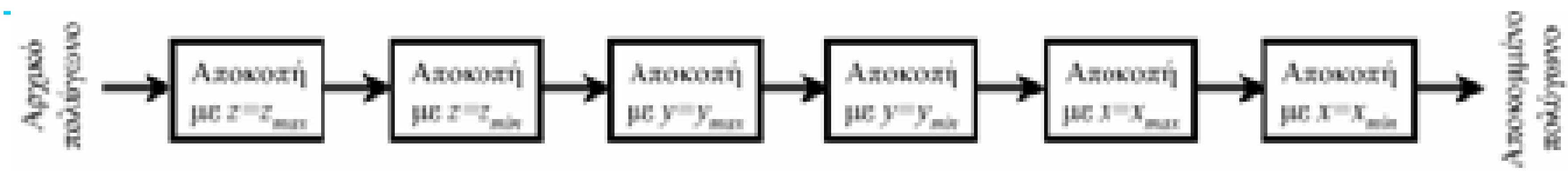
Αν $t \in [0,1]$, υπάρχει σημείο τομής με συντεταγμένες:

$(x_1 + t\ \Delta x\ , Y, z_1 + t\ \Delta z)$

$= (x_1 + (Y – y_1)\ \Delta x/\Delta y, Y, z_1 + (Y – y_1)\Delta z/\Delta y)$

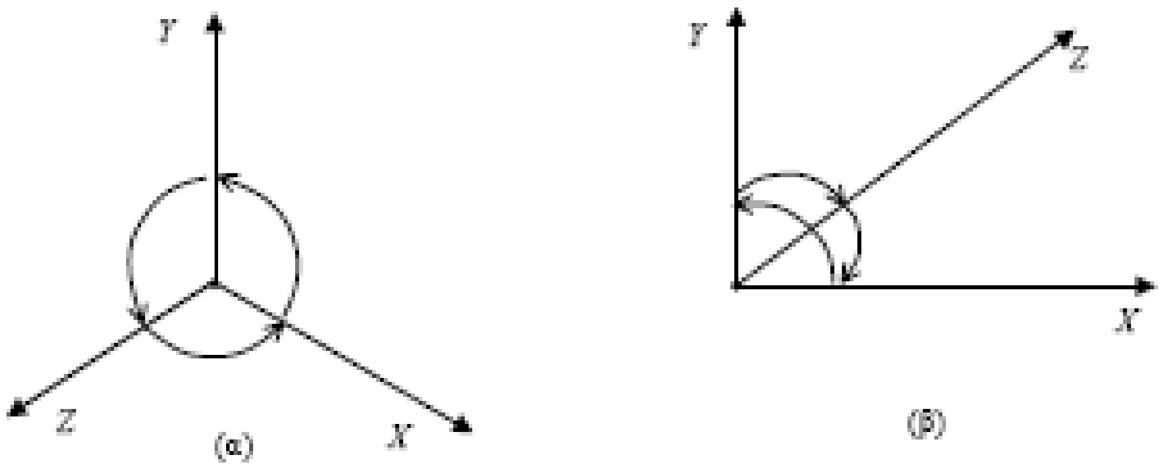
Αλγόριθμος Sutherland-Hodgman 3-Δ

- 6 στάδια αποκοπής για τα 6 επίπεδα
- έλεγχος αν p(x,y,z) εσωτερικό επιπέδου (a,b,c,d) από το πρόσημο $f(x,y,z) = ax + by + cz + d$
- Υπολογισμός τομής ευθυγράμμου τμήματος με επίπεδο: π.χ. όπως στη περίπτωση Cohen - Sutherland



Μετασχηματισμοί 3-Δ

- Ομογενείς συντεταγμένες σημείων **E^3 : (x,y,z,w)**
- Βασική παράσταση: (x,y,z,1)
- Δεξιόστροφα (εδώ) ή αριστερόστροφα συστήματα.



Μετασχηματισμοί 3-Δ

- Μεταφορά

$$T(\vec{d}) = \begin{bmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- Αλλαγή κλίμακας

$$S(s_x, s_y, s_z) = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Μετασχηματισμοί 3-Δ

- Στροφή: Ανάγκη καθορισμού θετικής / αρνητικής φοράς
- Ορθογώνιοι μετασχηματισμοί

$$R_x(\theta)=\begin{bmatrix}1 & 0 & 0 & 0\\ 0 & \cos\theta & -\sin\theta & 0\\ 0 & \sin\theta & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix} \quad R_y(\theta)=\begin{bmatrix}\cos\theta & 0 & \sin\theta & 0\\ 0 & 1 & 0 & 0\\ -\sin\theta & 0 & \cos\theta & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

$$R_z(\theta)=\begin{bmatrix}\cos\theta & -\sin\theta & 0 & 0\\ \sin\theta & \cos\theta & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Μετασχηματισμοί 3-Δ

- Στρέβλωση στο XY επίπεδο
  - a, b παράγοντες στρέβλωσης κατά X και Y άξονα
  - z συντεταγμένη αμετάβλητη

$$SH_{xy}(a,b)=\begin{bmatrix}1 & 0 & a & 0\\ 0 & 1 & b & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

- Στρέβλωση στο YZ επίπεδο

$$SH_{yz}(a,b)=\begin{bmatrix}1 & 0 & 0 & 0\\ a & 1 & 0 & 0\\ b & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

- Στρέβλωση στο XZ επίπεδο

$$SH_{xz}(a,b)=\begin{bmatrix}0 & a & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & b & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}$$

Φροντιστηριακή άσκηση Φ5

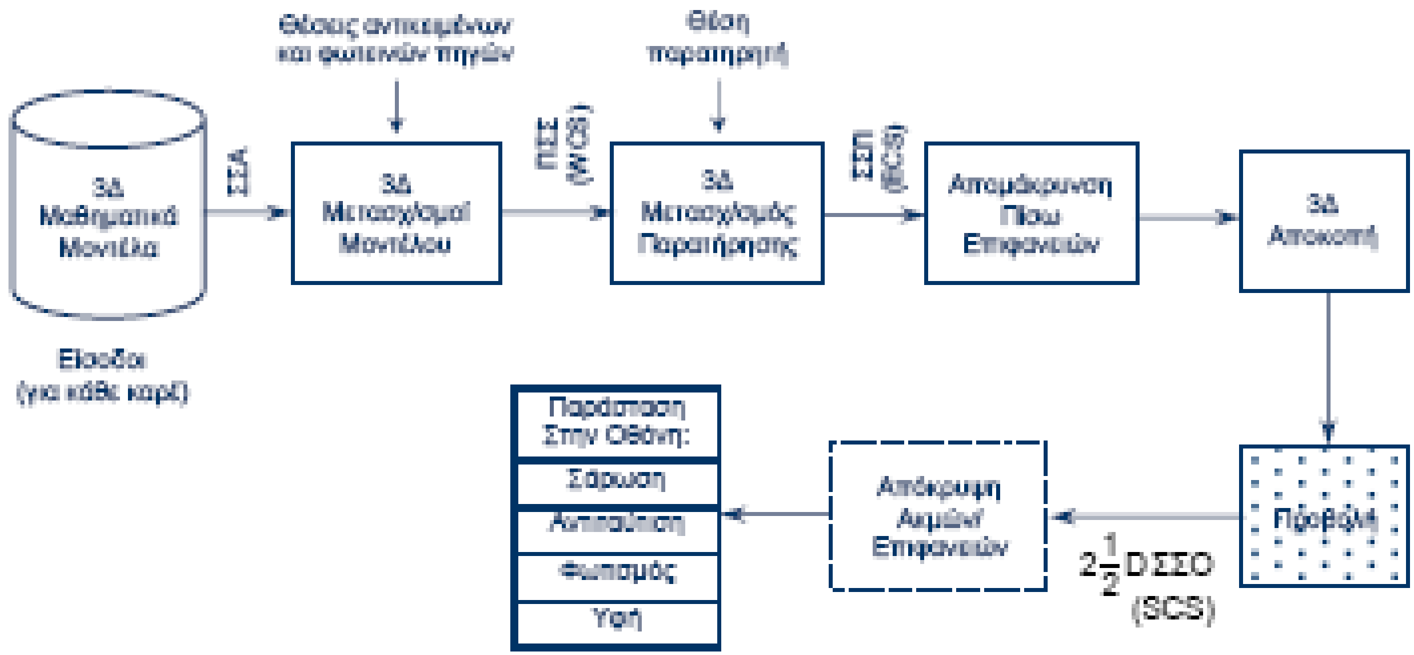
- Μετασχηματισμοί 3 διαστάσεων

Σελ 119 (Θεοχάρης) άσκηση α.3.9

Περιστροφή πυραμίδας ως προς άξονα

ΜΑΘΗΜΑ #9 Προβολές

- Απαραίτητες αφού 3Δ αντικείμενα απεικονίζονται σε 2Δ συσκευές.

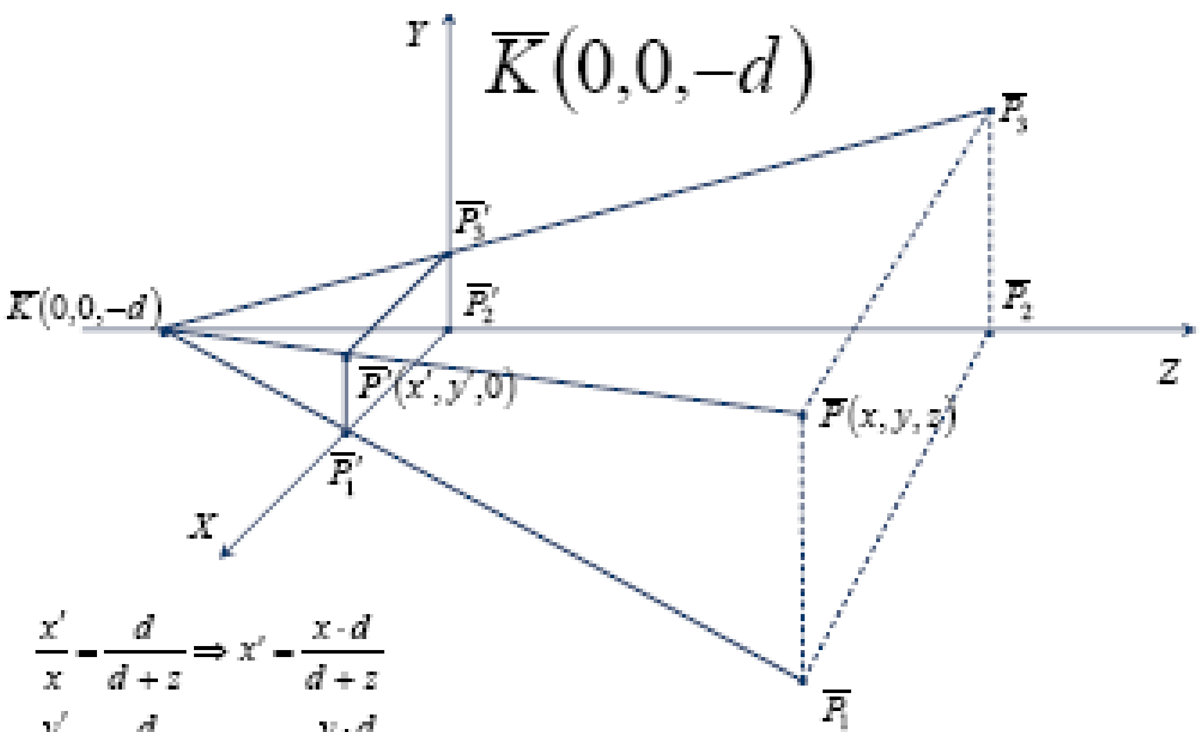


Προβολές

- Προοπτική: πεπερασμένη απόσταση κέντρου προβολής από επίπεδο προβολής.
- Παράλληλη: άπειρη απόσταση κέντρου προβολής από επίπεδο προβολής.
- Ιδιότητες προβολών:
  - Ευθείες προβάλλονται σε ευθείες.
  - Αποστάσεις αλλάζουν (γενικά).
  - 3Δ παράλληλες ευθείες, μη παράλληλες με επίπεδο προβολής, δεν προβάλλονται σε παράλληλες ευθείες.
  - Γωνία μεταξύ ευθειών αλλάζει, εκτός αν επίπεδο γωνίας παράλληλο με επίπεδο προβολής.

Προοπτική Προβολή

- Έστω προβολή στο επίπεδο *XY με κέντρο* προβολής



$$\frac{x'}{x} = \frac{d}{d+z} \Rightarrow x' = \frac{x \cdot d}{d+z}$$

$$\frac{y'}{y} = \frac{d}{d+z} \Rightarrow y' = \frac{y \cdot d}{d+z}$$

Προοπτική Προβολή

- Δεν είναι γραμμικός μετασχηματισμός (διαίρεση με *z).*
- Δεν μπορεί να δοθεί με μορφή πίνακα.
- Ωστόσο μπορούμε να χρησιμοποιήσουμε μεταβολή του *w.*

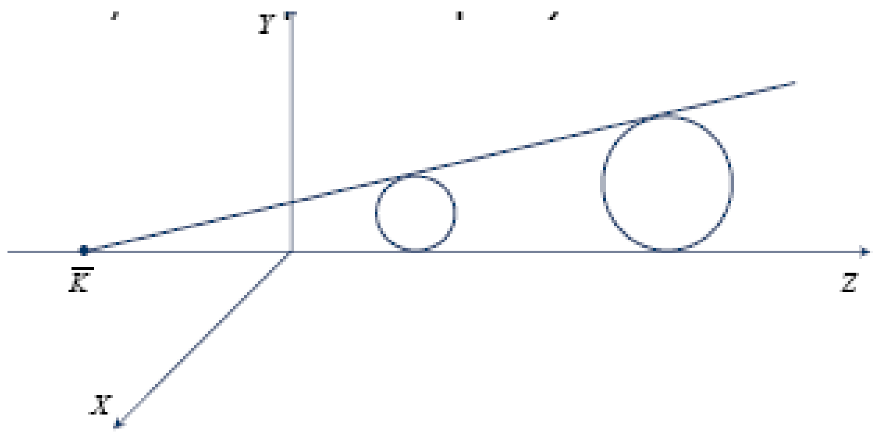
$$P_{pers} = \begin{bmatrix} d & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & d \end{bmatrix} \qquad \begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = P_{pers} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \cdot d \\ y \cdot d \\ 0 \\ z + d \end{bmatrix}$$

- Ακολουθεί διαίρεση με *w*
  (αφού πρέπει w=1)

$$P' = \begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} / w' = \begin{bmatrix} \frac{x \cdot d}{z + d} \\ \frac{y \cdot d}{z + d} \\ 0 \\ 1 \end{bmatrix}$$

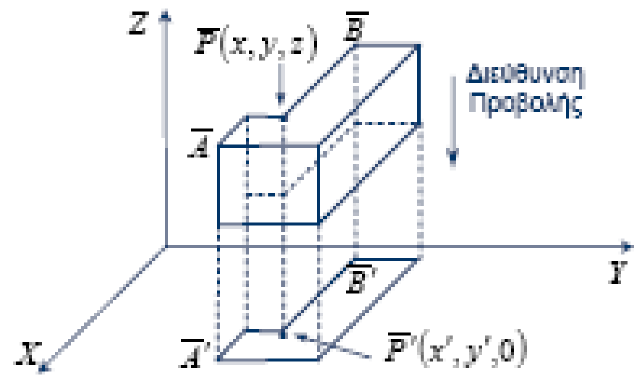
Προοπτική Προβολή

Χαρακτηριστικό: κεντρική σμίκρυνση
(όπως το ανθρώπινο μάτι).



Παράλληλη Προβολή

- Κέντρο προβολής στο άπειρο, δίνεται κατεύθυνση προβολής.
  - Διατηρεί αποστάσεις, χρήσιμο στοιχείο π.χ. στην αρχιτεκτονική.

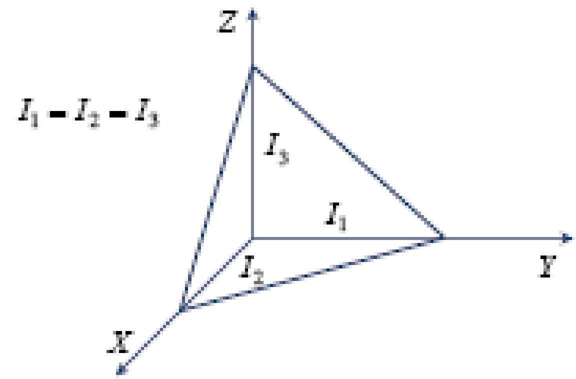


- Ορθογώνια παράλληλη προβολή: πάνω σε ένα από τα βασικά επίπεδα με κάθετες ακτίνες προβολής.
  - Πίνακας μετασχηματισμού για ορθογώνια π.χ. στο *XY*

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Παράλληλη Προβολή

- Παράλληλες προβολές διακρίνονται σε:
  – Ορθογραφικές: ακτίνες προβολής κάθετες στο επίπεδο προβολής.
  – Πλάγιες: ακτίνες όχι κάθετες.



- Ορθογραφικές διακρίνονται σε:
  – Ορθογώνιες: ακτίνες προβολής παράλληλες με *X, Y ή Z.*
  – Αξονομετρικές: μη ορθογώνιες.
  – Ισομετρικές: ακτίνες προβολής παράλληλες με κύρια διαγώνιο χώρου.

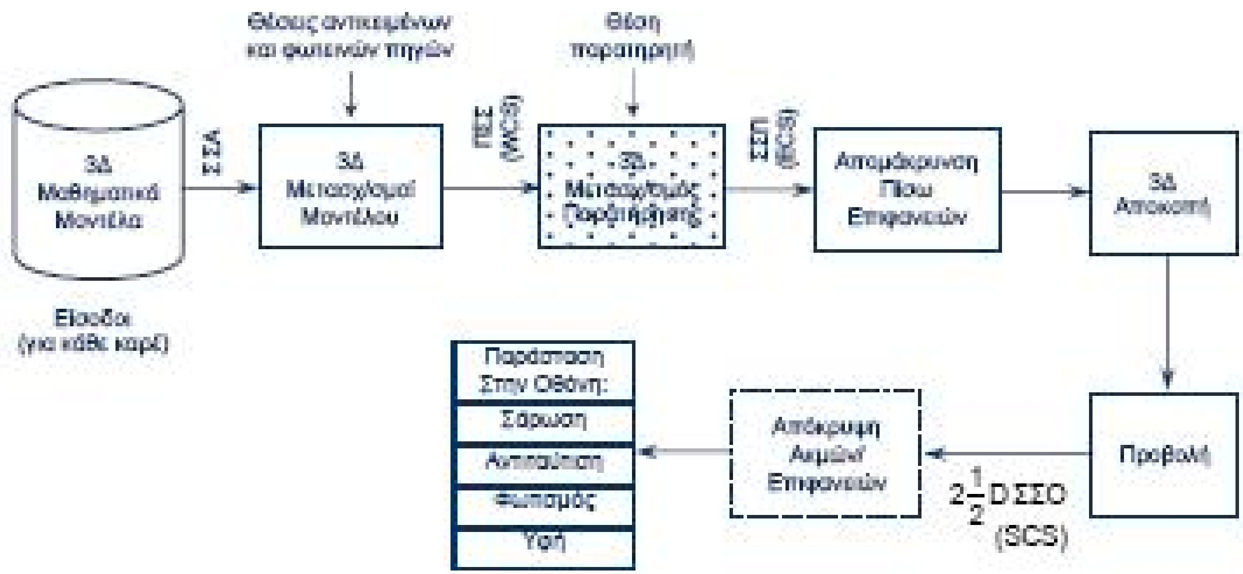
Παράδειγμα - άσκηση

- Προβολή μοναδιαίου κύβου στο επίπεδο XY για προοπτική προβολή με d=1 και d=10

  (μοναδιαίος κύβος έχει μια κορυφή στο 0,0,0 και μία κορυφή στο (1,1,1), περιγράφεται από 8 σημεία, μήτρα 8*4)

Μετασχηματισμός Παρατήρησης

- Παγκόσμιο Σύστημα Συντεταγμένων Σύστημα Συντεταγμένων Παρατηρητή.
- Σύνθεση βασικών μετασχηματισμών.
- Καθορίζει όρια αποκοπής & παραμέτρους προβολής
- Θα εξετάσουμε ΜΠ Ι



Μετασχηματισμός Παρατήρησης Ι

- ΜΠ Ι χρησιμοποιεί προοπτική προβολή και καθορίζεται από:

  α. Σημείο παρατήρησης <u>Ο</u>

  β. 2 ανύσματα για $Y_Π$ *και* $Z_Π$ *(αριστερόστροφο).*

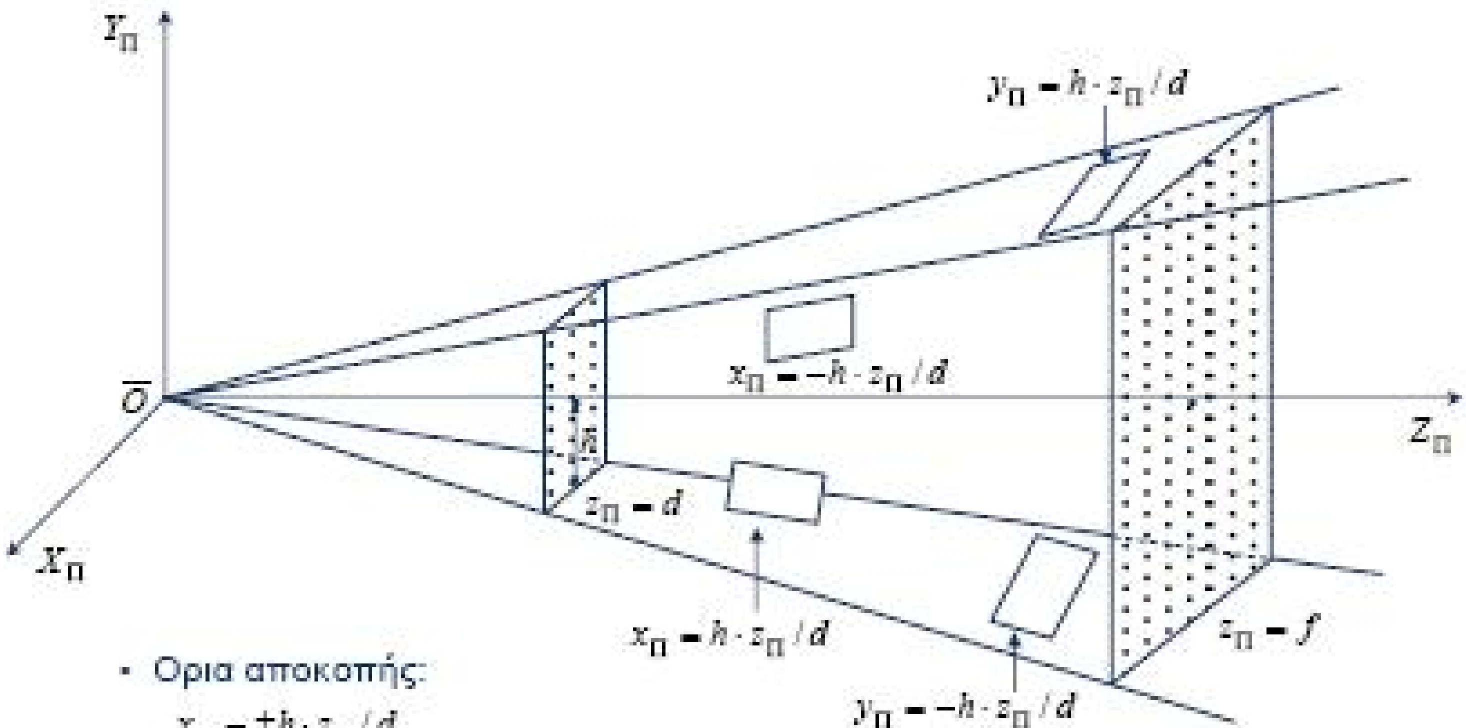
  Από Παγκόσμιο Σύστημα Συντεταγμένων (ΠΣΣ) σε σύστημα συντεταγμένων παρατηρητή (ΣΣΠ) :

  1. Μεταφορά στην αρχή ΠΣΣ.
  2. Στροφές γύρω από *Χ, Υ και Ζ του ΠΣΣ ώστε να ταυτισθούν* με αντίστοιχους ΣΣΠ.
  3. Αποκοπή.
  4. Προοπτική Προβολή.

Χρησιμοποιούνται 3 ακόμα παράμετροι για καθορισμό ορίων αποκοπής:

- γ. Μέγεθος παραθύρου προβολής *2h.*
- δ. Απόσταση *d* ***έμπροσθεν επιπέδου αποκοπής από (κάθετο στον*** $Z_Π$***).***
- ε. Απόσταση *f* ***όπισθεν επιπέδου αποκοπής από (κάθετο στον*** $Z_Π$***).***

Μετασχηματισμός Παρατήρησης I



- Ορια αποκοπής:

  $x_\Pi = \pm h \cdot z_\Pi / d$

  $y_\Pi = \pm h \cdot z_\Pi / d$

  $z_\Pi = d$

  $z_\Pi = f$

Μετασχηματισμός Παρατήρησης Ι

- Χαρακτηριστικά ΜΠ Ι:
  - Επίπεδο προβολής ταυτόσημο με έμπροσθεν επίπεδο αποκοπής $z_{\Pi}=d$.
  - Κέντρο προβολής ταυτόσημο με σημείο παρατήρησης.
  - Προοπτική προβολή.
  - Τετράγωνο παράθυρο, συμμετρικό ως προς Z_{Π}.
- **Πληροφορία z διατηρείται κατά την προβολή** για 2 λόγους:
  - 3Δ αποκοπή ευκολότερη σαν ενδιάμεσο στάδιο της προβολής.
  - Απόκρυψη απαιτεί z πληροφορία. (Z-Buffer)

Προβολή στον ΜΠ Ι

- Θα χρειαστούμε ΡΜΠ1 από ΣΣΠ σε ΣΣ Οθόνης (ΣΣΟ):

$$x_O = \frac{d \cdot x_\Pi}{h \cdot z_\Pi} \qquad -1 \le x_O \le 1$$

$$y_O = \frac{d \cdot y_\Pi}{h \cdot z_\Pi} \qquad -1 \le y_O \le 1$$

- Μετασχηματισμός Z πρέπει να πληρεί:
  - Ευθείες ΣΣΠ →ευθείες ΣΣΟ.
  - Επίπεδα ΣΣΠ →επίπεδα ΣΣΟ.
  - Κανονικοποιημένες τιμές z_O.
- $z_O = A + B/z_\Pi$ είναι ΟΚ, με περιορισμούς:
  1. $B<0$, ώστε αύξηση $z_\Pi \Rightarrow$ αύξηση z_O.
  2. $z_\Pi \in [d, f] \Rightarrow z_O \in [0, 1]$
- 2 εξισώσεις με 2 αγνώστους:

  $0 = A + B/d \qquad 1 = A + B/f$
- Επίλυση με περιορισμό 1. δίνει:

  $A = f/(f-d) \qquad B = -fd/(f-d)$
- Τελικά:

$$z_O = \frac{f(1 - d/z_\Pi)}{f - d} \qquad 0 \le z_\Pi \le 1$$

Προβολή στον ΜΠ Ι

Ο παραπάνω μετασχηματισμός μπορεί να χωρισθεί σε:

– Γραμμικό μέρος (πίνακας).

$$\begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} = P_{\text{ΜΠ}} \cdot \begin{bmatrix} x_{\Pi} \\ y_{\Pi} \\ z_{\Pi} \\ 1 \end{bmatrix}$$

Με $P_{\text{ΜΠ}} = \begin{bmatrix} \frac{d}{h} & 0 & 0 & 0 \\ 0 & \frac{d}{h} & 0 & 0 \\ 0 & 0 & \frac{f}{f-d} & -\frac{fd}{f-d} \\ 0 & 0 & 1 & 0 \end{bmatrix}$

– Διαίρεση με ομογενή συντεταγμένη (= z_{Π})

$$\begin{bmatrix} x_O \\ y_O \\ z_O \\ 1 \end{bmatrix} = \begin{bmatrix} x' \\ y' \\ z' \\ w' \end{bmatrix} / w'$$

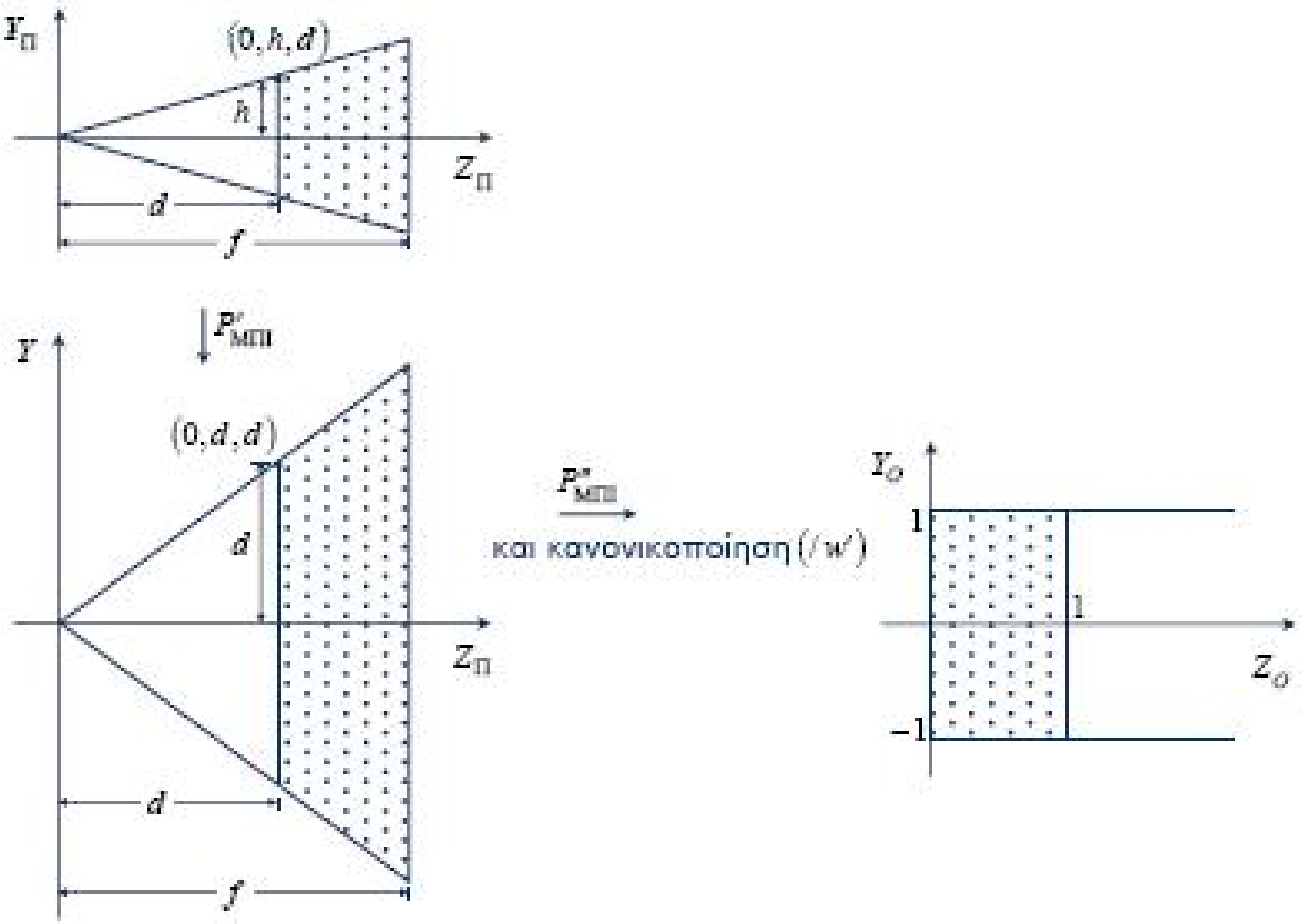
Προβολή στον ΜΠ Ι

- $P_{\text{ΜΠΙ}}$ πιο κατανοητός αν γραφεί:

$$P_{\text{ΜΠΙ}} = P''_{\text{ΜΠΙ}} \cdot P'_{\text{ΜΠΙ}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \dfrac{f}{f-d} & -\dfrac{fd}{f-d} \\ 0 & 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} \dfrac{d}{h} & 0 & 0 & 0 \\ 0 & \dfrac{d}{h} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- $P'_{\text{ΜΠΙ}}$ είναι αλλαγή κλίμακας κατά d/h :
  - Πυραμίδα αποκοπής γίνεται κανονική πυραμίδα.
  - π.χ. $[0,h,d,1] \rightarrow [0,d,d,1]$
- $P''_{\text{ΜΠΙ}}$ μετατρέπει κανονική πυραμίδα σε ορθογώνιο παραλληλεπίπεδο:
  - Εμπροσθεν επίπεδο αποκοπής $z_{\Pi}=d$ γίνεται XY .
  - Οπισθεν επίπεδο αποκοπής $z_{\Pi}=f$ γίνεται $z=1$.
  - π.χ. $[0,d,d,1] \rightarrow [0,d,0,d]$ δηλ. $[0,1,0,1]$

Προβολή στον ΜΠ Ι



Φροντιστηριακή άσκηση Φ6 (παραδοτέα- τελευταία)

Να βρεθεί η μήτρα μετασχηματισμού ενός σημείου (x,y,z) από σύστημα συντεταγμένων φυσικού συστήματος σε ένα σύστημα σ6[illegible] [illegible]ταγμένων παρατήρησης (ΣΣΠ) με χαρακτηριστικά:

ο άξονας Z του ΣΣΠ να δείχνει κατευθείαν στην αρχή του φυσικού συστήματος και ο άξονας X να είναι παράλληλος στο επίπεδο XY του φυσικού συστήματος οι δε άξονες X,Y του ΣΣΠ να είναι παράλληλοι στους άξονες της οθόνης που αποτελεί το επίπεδο προβολής

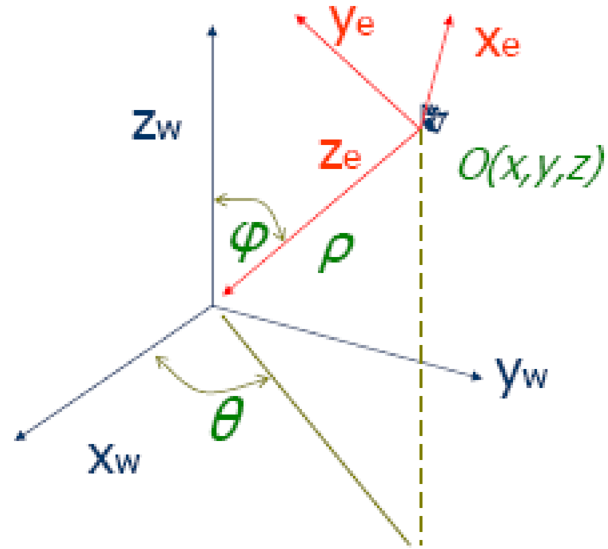
Άσκηση (συνέχεια)

- Χρησιμοποιείστε το σχήμα για τις εξισώσεις μετασχηματισμού

$x = \rho * \sin\phi * \cos\theta$

$y = \rho * \sin\phi * \sin\theta$

$z = \rho * \cos\phi$



Άσκηση (συνέχεια)

Στη συνέχεια χρησιμοποιήσατε τη μήτρα μετασχηματισμού καθώς και τις εξισώσεις της προοπτικής προβολής για να λύσετε το εξής πρόβλημα: Αν υποθέσουμε κύβο του οποίου το κέντρο βρίσκεται στην αρχή του φυσικού συστήματος συντεταγμένων με ακμή 1 εκ, έστω ότι παρατηρούμε τον κύβο από σημείο 6,10,8 με τον άξονα παρατήρησης z να δείχνει προς την αρχή του φυσικού συστήματος συντεταγμένων. Υποθέτουμε ότι ο άξονας των χ βρίσκεται στο επίπεδο z=8. Επίσης υποθέστε ότι βλέπουμε το αντικείμενο από απόσταση 0,6 μ. Και ότι η οθόνη (πεδίο προβολής) είναι διαστάσεων 0,3*0,3 μ. Με συντεταγμένες 1024*1024. Να βρεθεί και να υπολογιστεί η προοπτική προβολή του κύβου στο επίπεδο της οθόνης

Αποκοπή στο ΜΠ Ι

- Τιμή ορίζει και τα όρια αποκοπής:

$$w' \leq x' \leq w'$$

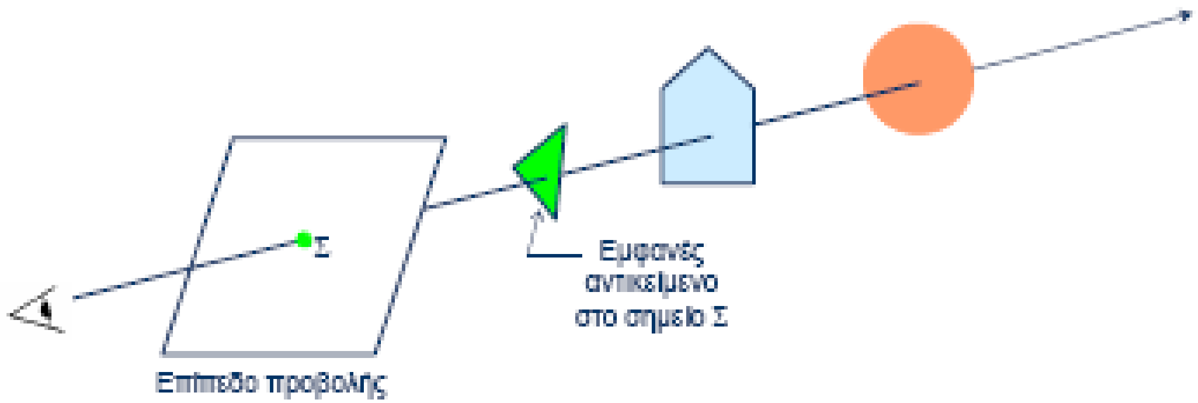
$$-w' \leq y' \leq w'$$

$$0 \leq z' \leq w'$$

- Αποκοπή γίνεται μετά την εφαρμογή $P_{ΜΠΙ}$ αλλά πριν τη διαίρεση με το w'

#10 Πρόβλημα Απόκρυψης

- Ποιο είναι το εμφανές αντικείμενο (χρώμα) σε κάθε σημείο του επιπέδου προβολής;



- Χωρίζονται σε αλγόριθμους απόκρυψης ακμών και επιφανειών.
- Χρησιμοποιούν άμεσα ή έμμεσα ταξινόμηση στις διαστάσεις *X Y και Z.*

Αλγόριθμοι Απόκρυψης

- Χωρίζονται σε
  - αλγόριθμους χώρου αντικειμένων και
  - αλγόριθμους χώρου εικόνας.

- Αλγόριθμοι χώρου αντικειμένων είναι $O(\Pi^2)$ ενώ αλγόριθμοι χώρου εικόνας είναι O(Π-P) όπου *Π ο αριθμός των πολυγώνων* και *P ο αριθμός των pixels*.

Αλγόριθμοι χώρου αντικειμένων

- Αλγόριθμοι χώρου αντικειμένων συγκρίνουν αντικείμενα μεταξύ τους για να βρουν το πλησιέστερο σε κάθε σημείο του επιπέδου προβολής:

Για κάθε αντικείμενο π

{

Εύρεση των ορατών τμημάτων του π μέσω σύγκρισης (X,Y,Z) με όλα τα άλλα αντικείμενα;

Χρωματισμός των ορατών τμημάτων του π στην οθόνη

}

Αλγόριθμοι χώρου εικόνας

- Αλγόριθμοι χώρου εικόνας έχουν την εξής μορφή:

Για κάθε pixel ρ της εικόνας

{

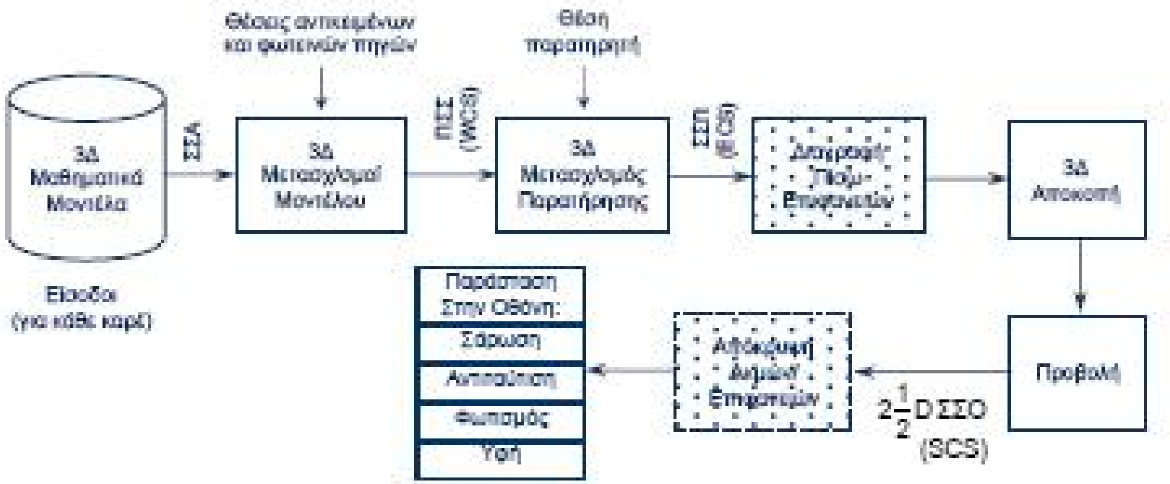
Εύρεση του πλησιέστερου αντικειμένου που τέμνεται από την ακτίνα προβολής που περνά από το ρ;

Χρωματισμός του ρ με το χρώμα του πλησιέστερου αντικειμένου στο σημείο τομής

}

Αλγόριθμοι Απόκρυψης

- Υπολογιστική ακρίβεια.
  - Αλγόριθμοι χώρου εικόνας: ακρίβεια που απαιτεί η ανάλυση της εικόνας.
  - Αλγόριθμοι χώρου αντικειμένων: ακρίβεια ορισμού αντικειμένων (=ακρίβεια υπολογιστή).
- Θέση στη γραφική σωλήνωση εξόδου.
  - Αλγόριθμοι χώρου αντικειμένων: μετά την προβολή (διακεκομμένη γραμμή).
  - Αλγόριθμοι χώρου εικόνας: ενσωματώνονται στη διαδικασία παράστασης στην οθόνη.
  - Διαγραφή πίσω επιφανειών.

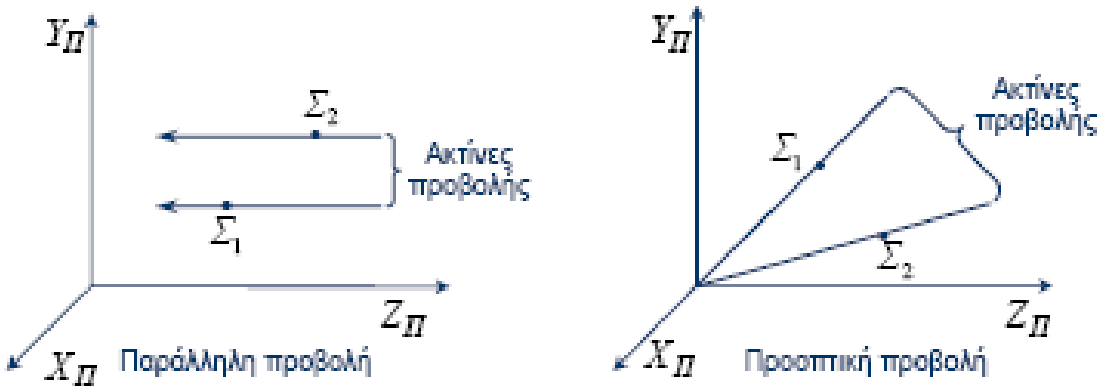


Αλγόριθμοι Απόκρυψης

- Τεχνικές βελτίωσης αποτελεσματικότητας αλγορίθμων απόκρυψης.
  - Εκμετάλλευση προοπτικού μετασχηματισμού.
  - Εκμετάλλευση συνάφειας.
  - Περιβάλλοντες όγκοι.
  - Διαμερισμός χώρου.

Προοπτικός Μετασχηματισμός

- Ζήτημα απόκρυψης μεταξύ δύο σημείων $Σ_1(x_1,y_1,z_1)$ και $Σ_2(x_2,y_2,z_2)$ υπάρχει μόνο αν τα δύο σημεία βρίσκονται πάνω στην ίδια ακτίνα προβολής.



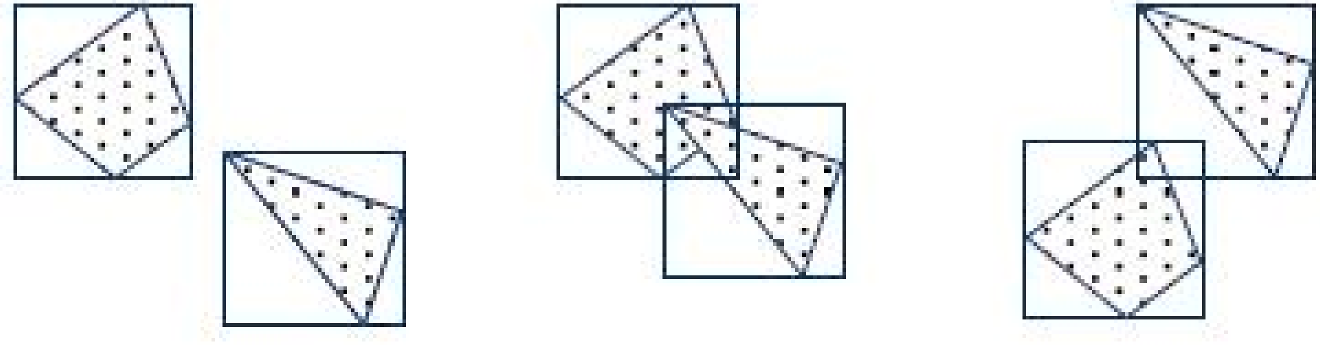
- Συνθήκη απόκρυψης στην παράλληλη προβολή: $(x_1=x_2)$ και $(y_1=y_2)$
- Συνθήκη απόκρυψης στην προοπτική προβολή: $(x_1/z_1=x_2/z_2)$, $(y_1/z_1=y_2/z_2)$
  - Απαιτεί (ακριβή) διαίρεση με z.
  - Η διαίρεση αυτή γίνεται κατά την προοπτική προβολή.
  - Η προοπτική προβολή μετατρέπει τις ακτίνες προβολής σε παράλληλες.
  - Φυλάμε z συντεταγμένη για σύγκριση βάθους.

Συνάφεια

- Συνάφεια: ιδιότητα γεωμετρικών οντοτήτων να διατηρούν τοπικά σταθερές τις τιμές των χαρακτηριστικών τους ή να τις μεταβάλλουν ομαλά.
  - π.χ. αυξητικός υπολογισμός z σημείων πολυγώνου.
- Είδη συνάφειας:
  - Συνάφεια ακμής.
  - Συνάφεια επιφάνειας.
  - Συνάφεια γραμμών σάρωσης.
  - Συνάφεια καρέ.

Περιβάλλοντες Όγκοι

- Απλούστεροι όγκοι από τα αντικείμενα που περιβάλλουν για μείωση κόστους συγκρίσεων κατά την ταξινόμηση στις διαστάσεις *X, Y και Z.*
  - Συχνά έχουν τη μορφή ορθογώνιου παραλληλογράμμου (2Δ) ή ορθογώνιου παραλληλεπιπέδου (3Δ).
  - Οχι απαραίτητα κλειστοί.
- Αν οι περιβάλλοντες όγκοι δύο αντικειμένων δεν τέμνονται τότε ούτε τα αντικείμενα τέμνονται. Το αντίθετο δεν ισχύει πάντα.

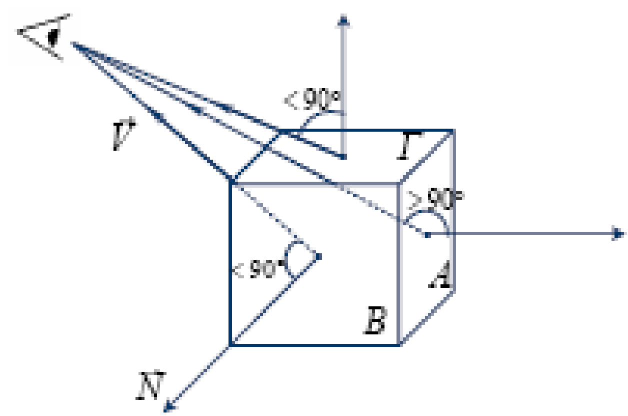


Διαμερισμός Χώρου

- Διαμερισμός χώρου σε σύνολο διατεταγμένων μερών (π.χ. voxels).
- Τα μέρη του χώρου που καταλαμβάνει ένα αντικείμενο ορίζουν έμμεσα τη διάταξή του σε σχέση με άλλα αντικείμενα.

Διαγραφή Πίσω Επιφανειών

- Αν η γωνία μεταξύ και είναι τότε η επιφάνεια είναι



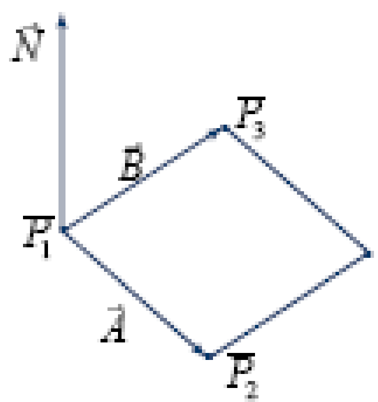
- Μια επιφάνεια είναι ορατή αν

$$\vec{V} \cdot \vec{N} = V_x \cdot N_x + V_y \cdot N_y + V_z \cdot N_z > 0$$

- Μειώνει όγκο δεδομένων κατά ~50%.
- Λύνει πρόβλημα απόκρυψης για ένα κυρτό αντικείμενο.πίσω (αόρατη).

Διαγραφή Πίσω Επιφανειών

- Λειτουργεί στο χώρο αντικειμένων και είναι Ο(Π) .
- V μπορεί να υπολογισθεί από μία κορυφή P της επιφάνειας.
  - Αν ο παρατηρητής βρίσκεται στο κέντρο του ΣΣΠ τότε V=-P
- N μπορεί να υπολογισθεί από 3 διαδοχικές, μη συγγραμμικές κορυφές P_1, P_2, P_3
  και $\vec{N} = \vec{A} \times \vec{B} = \left(\overline{P}_2 - \overline{P}_1\right) \times \left(\overline{P}_3 - \overline{P}_1\right)$



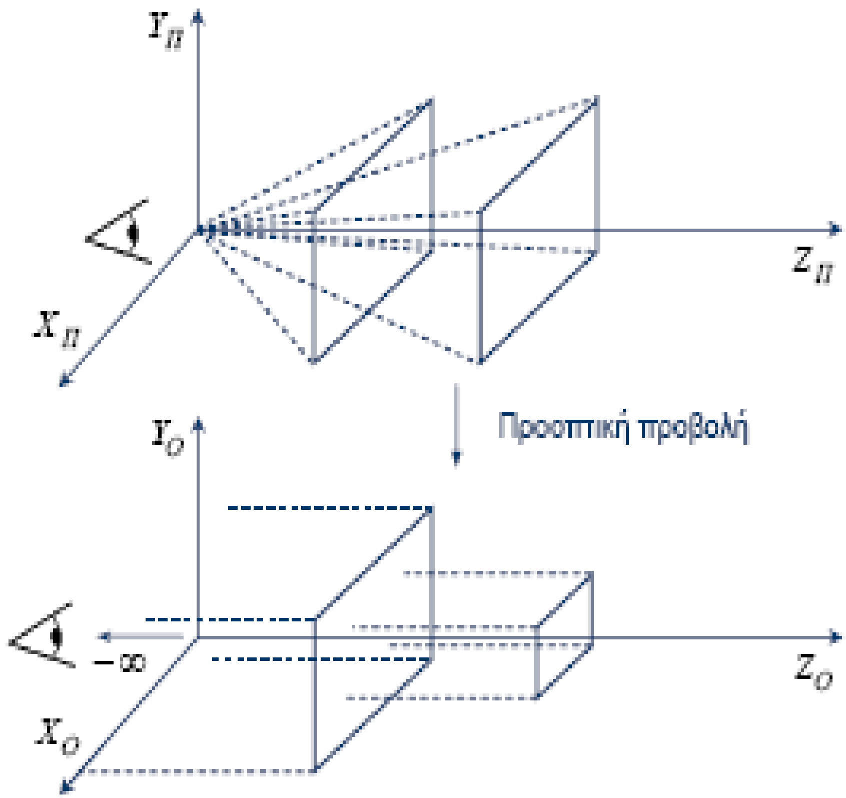
ΠΡΟΣΟΧΗ: $\vec{A} \times \vec{B} \neq \vec{B} \times \vec{A}$

Χρήση σειράς κορυφών

Αλγόριθμοι Απόκρυψης Επιφανειών

- Ακολούθησαν την εμφάνιση της πλεγματικής οθόνης.
- 4 Βασικές κατηγορίες.
  - z-buffer (πιο διαδεδομένος).
  - scanline.
  - ταξινόμηση κατά βάθος.
  - υποδιαίρεση επιφάνειας.

Αλγόριθμοι Απόκρυψης Επιφανειών



Βασική πράξη: σύγκριση βάθους 2 στοιχείων με ίδιο *XY.*

- Η προοπτική προβολή διευκολύνει.
- Μετατρέπει ακτίνες προβολής ώστε να είναι παράλληλες του *Z.*
- Ίσες αποστάσεις στον Z_π *δεν μετασχηματίζονται σε ίσες αποστάσεις στον* Z_o *.*
- Z_o *μεταβάλλεται ταχύτερα καθώς πλησιάζει τη μέγιστη τιμή του (1).*

Αλγόριθμος z-buffer

- Απαιτεί ύπαρξη μνήμης βάθους (z-buffer) για κάθε pixel.
  - Οι τιμές βάθους βρίσκονται (για τα περισσότερα σχήματα) με παρεμβολή.
  - Έμμεση ταξινόμηση.
  - Για κάθε pixel ο z-buffer φυλάει την ελάχιστη (ως τώρα) τιμή βάθους στο pixel αυτό.
  - Αρχικοποίηση στο μέγιστο z (πίσω επίπεδο αποκοπής).

Αλγόριθμος z-buffer

```
/* Αρχικοποίηση: m,n οι διαστάσεις της οθόνης */
for (x=0; x<m; x++) {
      for (y=0; y<n; y++) {
                z_buffer[x,y]=f; /*μέγιστο βάθος*/
                frame_buffer[x,y]=background; /*φόντο*/
      }
}
/*Επεξεργασία πολυγώνων*/
for (π=0; π<number_of_polygons(); π++)
{     /*Επεξεργασία scanlines πολυγώνου ymin…ymax*/
      for (y=ymin; y<=ymax; y++) {
/*Βρες με γραμμική παρεμβολή μεταξύ των αντίστοιχων κορυφών, τις τιμές τομής με
      scanline y :xleft,xright τις τιμές βάθους στις τομές :zleft,zright τις τιμές χρωματισμού
      στις τομές :cleft,cright*/
      for (x=xleft; x<=xright; x++) {
      /*Βρες με γραμμική παρεμβολή μεταξύ xleft και xright την τιμή βάθους z και χρώματος
      c σε κάθε pixel x,y*/
      if (z<z_buffer[x,y]) {
                z_buffer[x,y]=z;
                frame_buffer[x,y]=c;
```

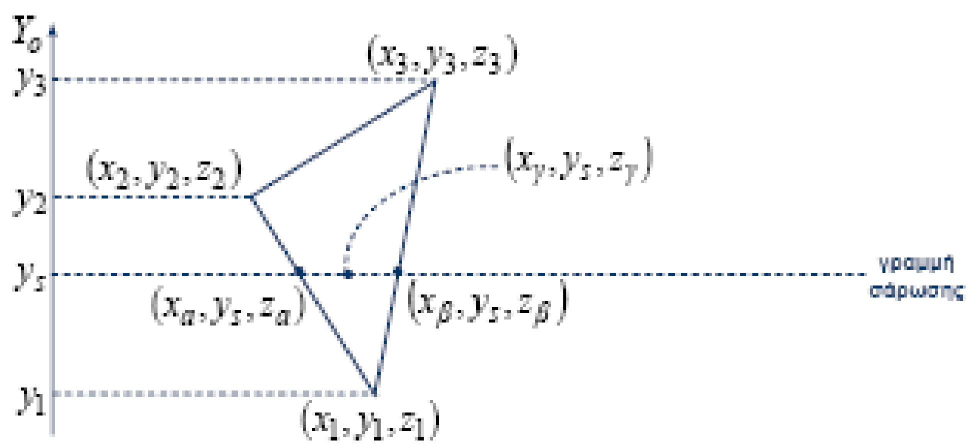
Αλγόριθμος z-buffer

- Τομές πλευρών πολυγώνου με scanlines βρίσκονται με λίστα ενεργών πλευρών.
- Γραμμική παρεμβολή z.

$$z_a = z_1 + (z_2 - z_1)\frac{y_s - y_1}{y_2 - y_1}$$

$$z_\beta = z_1 + (z_3 - z_1)\frac{y_s - y_1}{y_3 - y_1}$$

$$z_\gamma = z_a + (z_\beta - z_a)\frac{x_\gamma - x_a}{x_\beta - x_a}$$



Αλγόριθμος z-buffer

- O(Π,S), όπου *Π ο αριθμός πολυγώνων και S ο* μέσος αριθμός pixel ανά πολύγωνο.
  - Για τυπικές σκηνές $\mathit{\Pi} \cdot S \propto m \cdot n$ σταθερό).
- Πλεονεκτήματα z-buffer:
  - Ευκολία υλοποίησης σε H/W ή S/W.
  - Επεξεργασία πολυγώνων με τυχαία σειρά.
  - Δυνατότητα χρήσης για μη πολυγωνικά αντικείμενα (με συνάρτηση βάθους).
- Ανάγκη σε μνήμη:
  - Ένας καλός z-buffer απαιτεί 32 bits/pixel.
  - Για ανάλυση 1024x1024 αυτό συνεπάγεται 4Mbytes.

Ειδικές Εφαρμογές z-buffer

- Συνδυασμός εικόνων
  - Έστω οι frame και z-buffer εικόνων A και B.
  - A και B μπορεί να δημιουργήθηκαν χωριστά. Συνδυασμός:

```
for (x=0; x<m; x++)
  for (y=0; y<n; y++)
  {Z_C[x,y]=(Z_A[x,y]<Z_B[x,y])?Z_A[x,y]:Z_B[x,y];
  F_C[x,y]=(Z_A[x,y]<Z_B[x,y])?F_A[x,y]:F_B[x,y];}
```

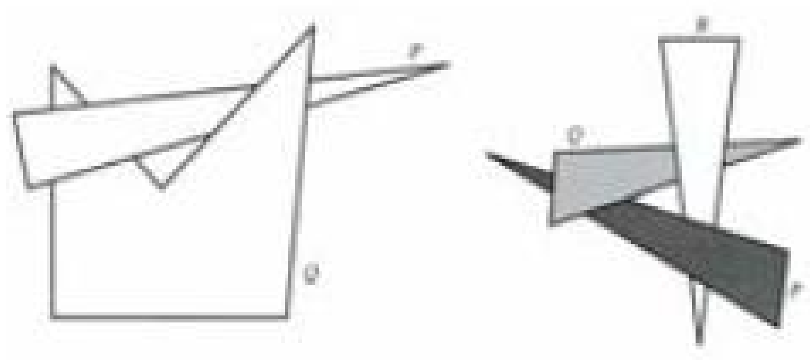
- Τοποθέτηση 3Δ αντικειμένων σε σκηνή με χρήση z-buffer.
  - 3Δ cursor.
  - Δεν μεταβάλλουμε περιεχόμενα z-buffer.

Αλγόριθμοι Λίστας Προτεραιότητας (αλγόριθμοι ζωγράφου)

- Ταξινόμηση πολυγώνων κατά βάθος και εμφάνιση με σωστή σειρά:
  - Οχι πάντα δυνατή (επικάλυψη z-έκτασης, τεμνόμενα πολύγωνα).
  - Διαμερισμός πολυγώνων.
- Μεγάλη πολυπλοκότητα.
- Χώρος αντικειμένων, αν αποτέλεσμα θεωρηθεί η ταξινομημένη λίστα.
- 1. Αλγόριθμος ταξινόμησης κατά βάθος (Newell 1972).
- 2. Αλγόριθμος BSP δένδρου (Fuchs 1983).

Αλγόριθμος Ταξινόμησης κατά Βάθος

- Ταξινόμηση κατά φθίνουσα απόσταση από κέντρο προβολής.
- Σύγκριση *P προ Q δεν είναι πάντα εύκολη.*



- Χρήση περιβάλλοντος παραλληλεπιπέδου $[x_{min}(P), y_{min}(P), z_{min}(P)],[x_{max}(P), y_{max}(P), z_{max}(P)]$.

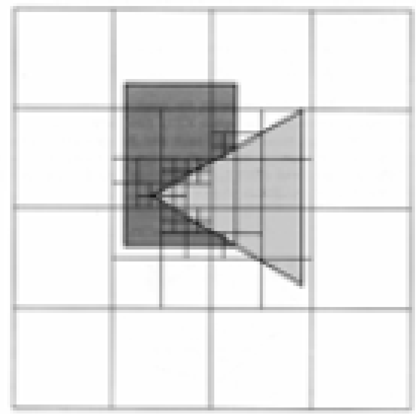
Αλγόριθμοι Υποδιαίρεσης Επιφάνειας

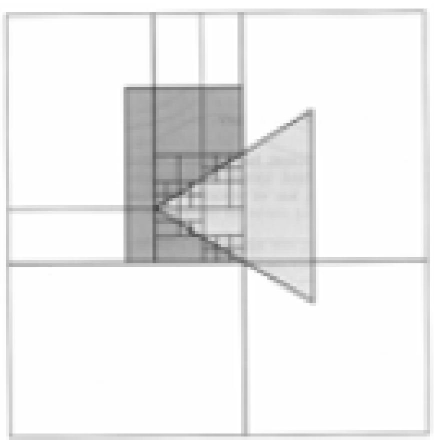
- Αναδρομική διαίρεση επιπέδου προβολής:
  - Αν 1 πολύγωνο καλύπτει τμήμα πλήρως, τότε αυτό παίρνει το χρώμα του πολυγώνου.
  - Αναδρομή σταματά επίσης στο μέγεθος του pixel.
  - Εκμετάλλευση συνάφειας επιφάνειας (μεγάλα πολύγωνα, σταθερού χρώματος).
- Αλγόριθμος Warnock:
  - Αναδρομή σταματά αν από ένα τμήμα περνά το πολύ 1 ακμή πολυγώνου.
  - Συγκρίσεις προβολής πολυγώνου P με περιοχή οθόνης B. Χρήση περιβάλλοντων ορθογωνίων:
- 4 περιπτώσεις σχέσης *P και B:*
  - (*D: Disjoint) P εξωτερικό της B.*
  - (*C: Contained) P εσωτερικό της B.*
  - (*S: Surrounding) P καλύπτον πλήρως την B.*
  - (*I: Intersecting) P τέμνον την B.*
  - Π.χ. σχέση *D εξασφαλίζεται εάν ισχύει:*

$$(x_{\min}(P) > x_{\max}(B)) \text{ ή } (x_{\max}(P) < x_{\min}(B)) \text{ ή }$$
$$(y_{\min}(P) > y_{\max}(B)) \text{ ή } (y_{\max}(P) < y_{\min}(B))$$

Αλγόριθμοι Υποδιαίρεσης Επιφάνειας

- Αναδρομικές διαιρέσεις Warnock:



- 

  Μείωση υποδιαιρέσεων με βάση κορυφές πολυγώνων (Weiler - Atherton):

Σύγκριση Μεθόδων Απόκρυψης

Αλγόριθμος	Αριθμός Πολυγωνικών Επιφανειών		
	200	5.000	120.000
depth - sort	$0,14 \bullet 10^6$	$1,4 \bullet 10^6$	$71 \bullet 10^6$
z-buffer	$7,5 \bullet 10^6$	$7,5 \bullet 10^6$	$7,5 \bullet 10^6$
Scan - line	$0,77 \bullet 10^6$	$2,9 \bullet 10^6$	$14 \bullet 10^6$
Υποδιαίρεση επιφάνειας	$1,6 \bullet 10^6$	$9 \bullet 10^6$	$43 \bullet 10^6$

Απόκρυψη και Παρακολούθηση Ακτίνας (Ray Tracing)

- Αλγόριθμος παρακολούθησης ακτίνας (Whitted):
  - Ακτίνες που ορίζονται από σημείο παρατήρησης και κάθε pixel ακολουθούνται ώσπου να συναντήσουν αντικείμενο.
  - Εκεί διασπώνται ανάλογα με μοντέλο.
  - Ουσιαστικά λύνει πρόβλημα απόκρυψης.