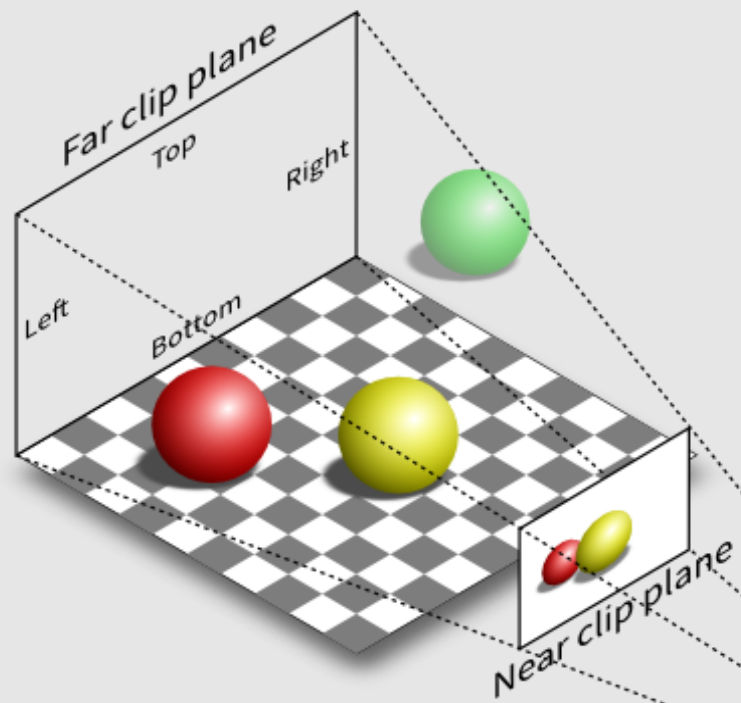
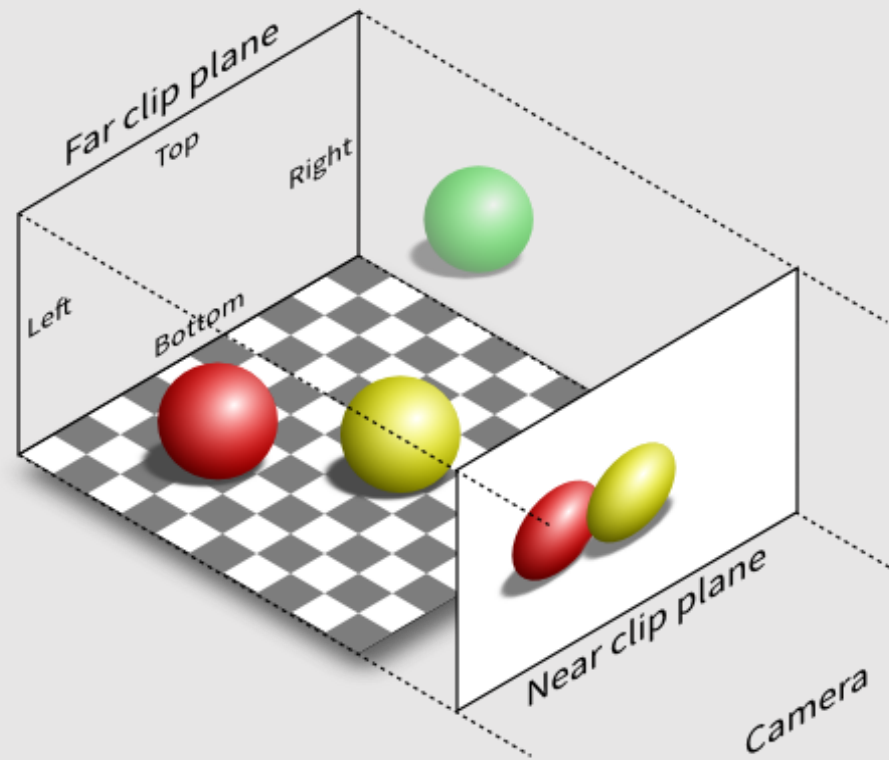




Perspective projection (P)



Orthographic projection (O)

Γραφικά Υπολογιστών

Αποκοπή (Clipping)

Andreas Aristidou

andarist@ucy.ac.cy

<http://www.andreasaristidou.com>

Ενδιάμεση Εξέταση

MARCH 2022						
Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2

Προβολή δύο διαστάσεων: Επανάληψη

- Μια περιοχή συντεταγμένων που επιλέγεται για προβολή ονομάζεται **παράθυρο**.
 - Το **παράθυρο** δηλώνει τι είναι αυτό που θα **δούμε**.
- Μια περιοχή σε μια συσκευή στην οποία αντιστοιχίζεται το παράθυρο προβολής (π.χ., οθόνη) ονομάζεται **viewport**.
 - Το **viewport** δηλώνει που θα γίνει η προβολή.

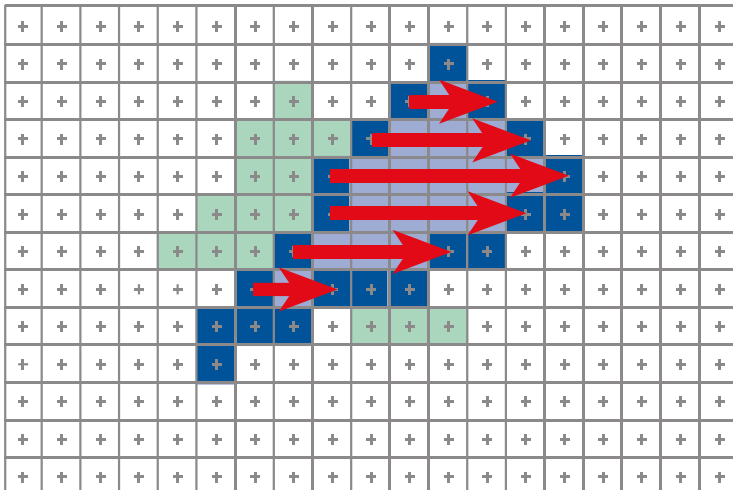


Δημιουργία εικόνων: Επανάληψη

A. Rendering Pipeline

- For each triangle
 - For each projected pixel

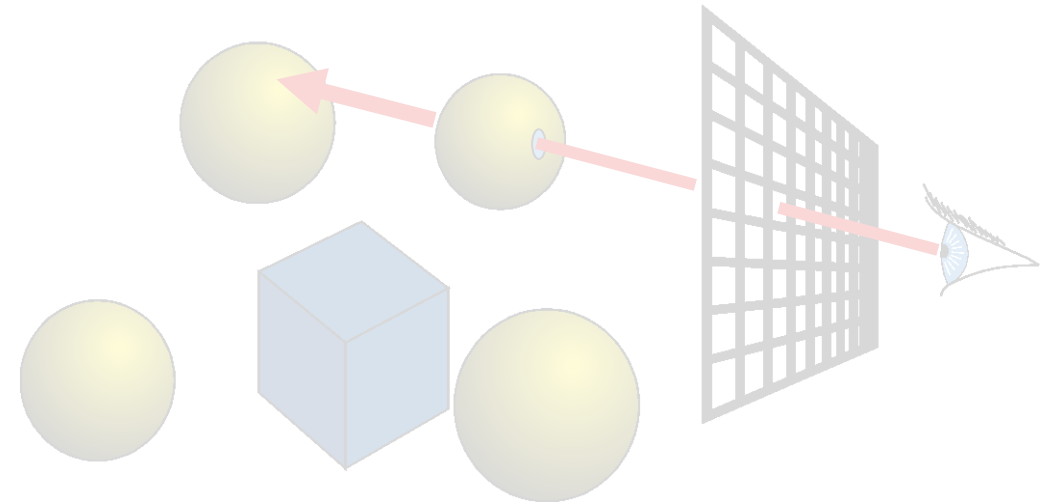
Project scene to the pixels



B. Ray Casting

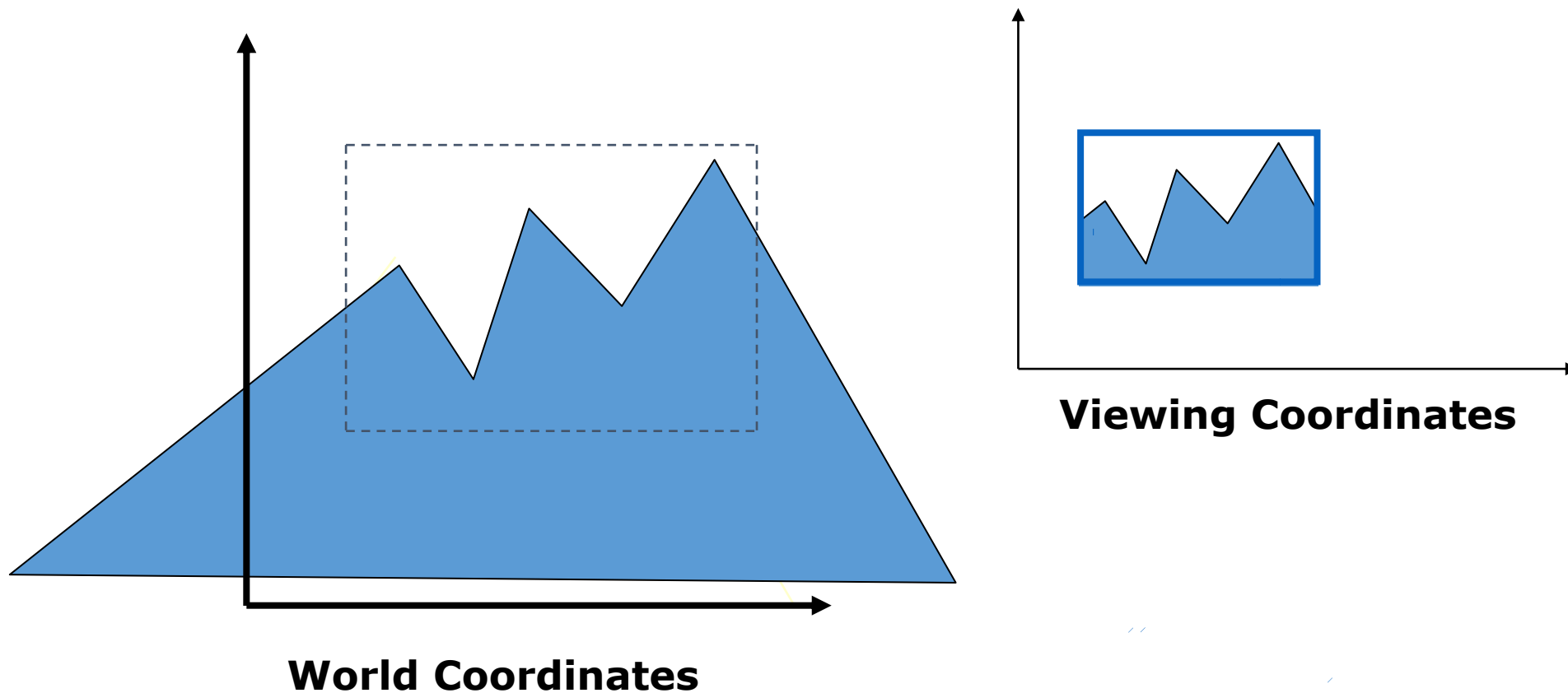
- For each pixel
 - For each object

Send pixels to the scene

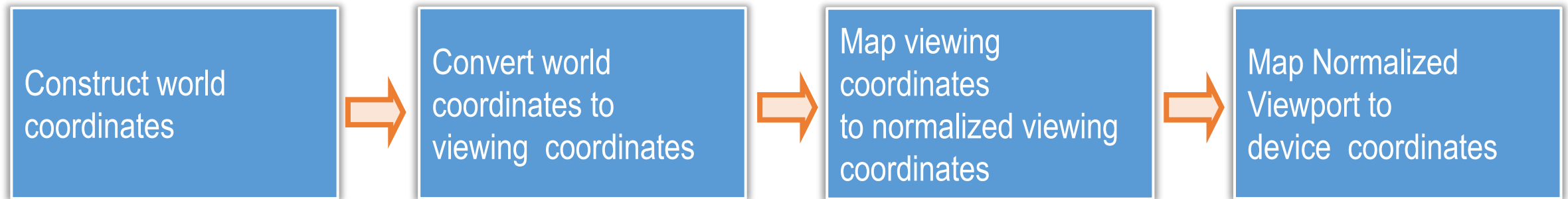


Προβολή δύο διαστάσεων: Επανάληψη

- Η απεικόνιση ενός μέρους από το παγκόσμιο σύστημα συντεταγμένων σε μια συσκευή ονομάζεται **μετασχηματισμός προβολής**

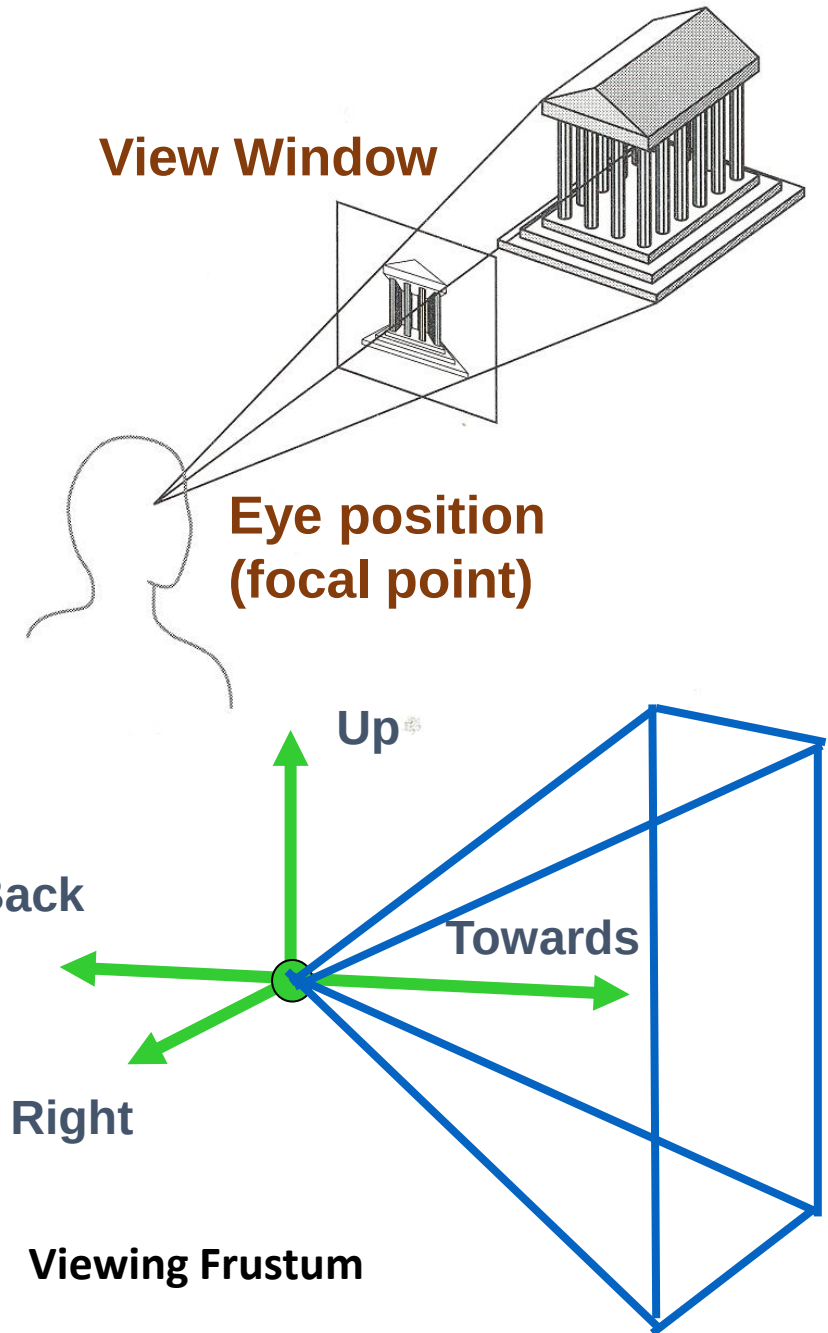
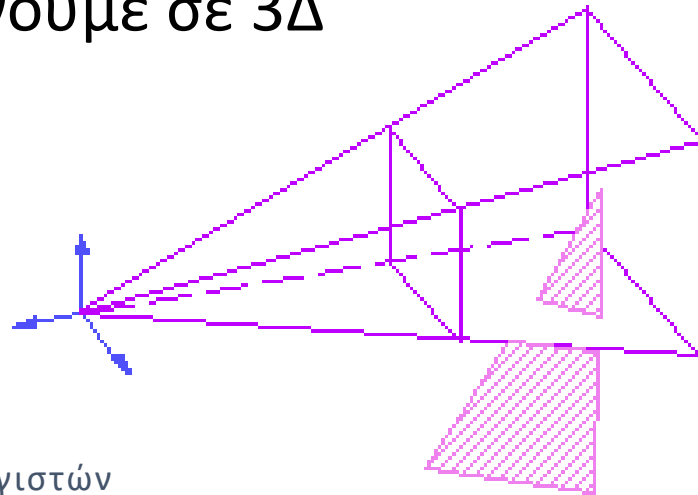


2Δ μετασχηματισμός προβολής: Επανάληψη



Αποκοπή (clipping)

- Είναι η διαδικασία εξεύρεσης του ακριβές μέρους του πολυγώνου που **βρίσκεται μέσα στο οπτικό πεδίο** (view volume)
- Για να διατηρηθεί η συνέπεια, το ψαλίδισμα ενός πολυγώνου πρέπει να οδηγήσει σε ένα πολύγωνο, όχι μια ακολουθία μερικώς αποσυνδεδεμένων γραμμών
- Θα εξετάσουμε αρχικά μια 2Δ λύση και έπειτα θα την επεκτείνουμε σε 3Δ

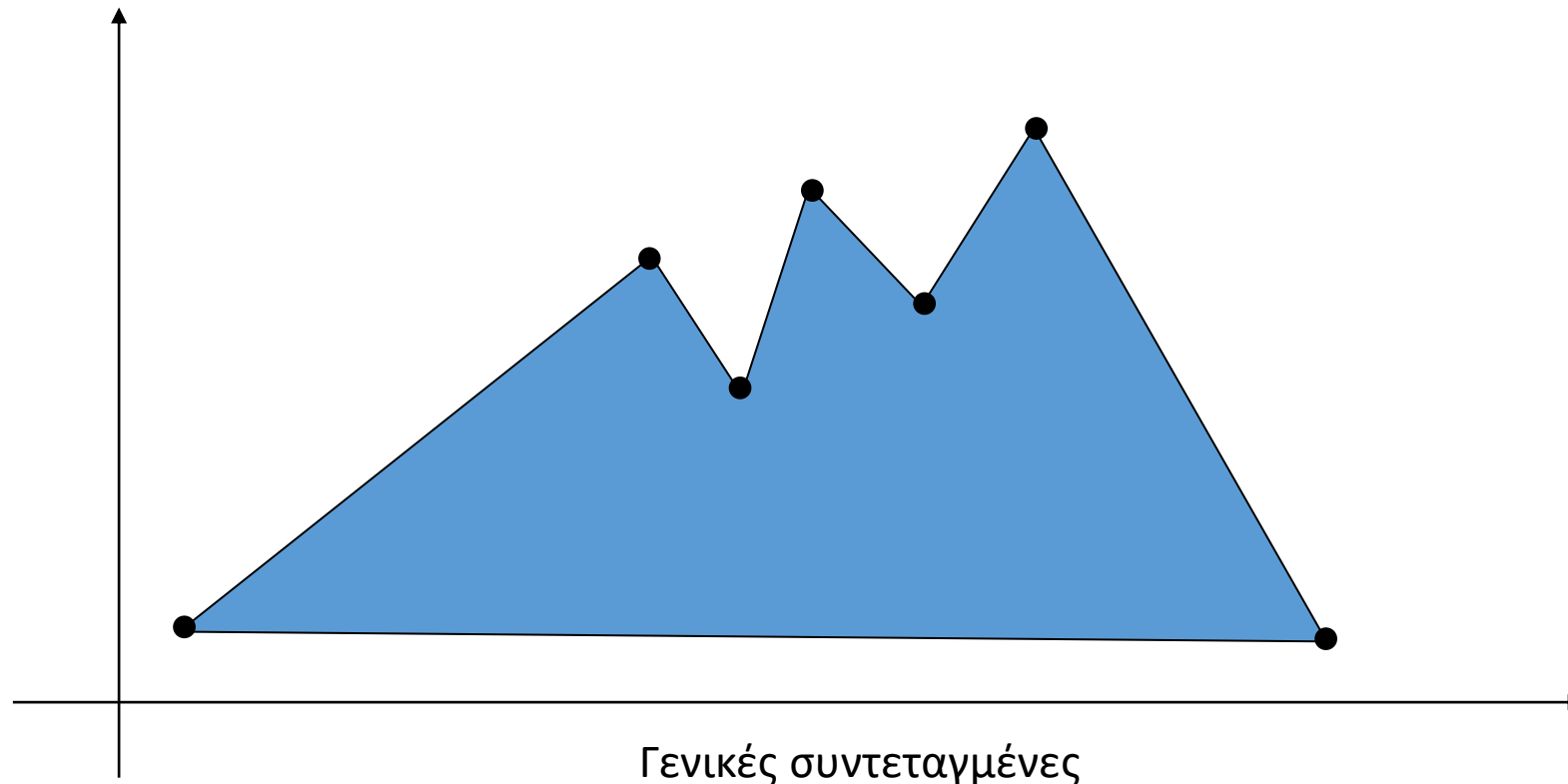


Γιατί γίνεται αποκοπή (Clipping);

- Γενικά είναι κακή ιδέα να δουλεύουμε έξω από τα όρια του buffer πλαισίου,
 - Γιατί;
- Μην χάνετε χρόνο να σαρώνετε και να μετατρέπετε pixel έξω από το viewing window

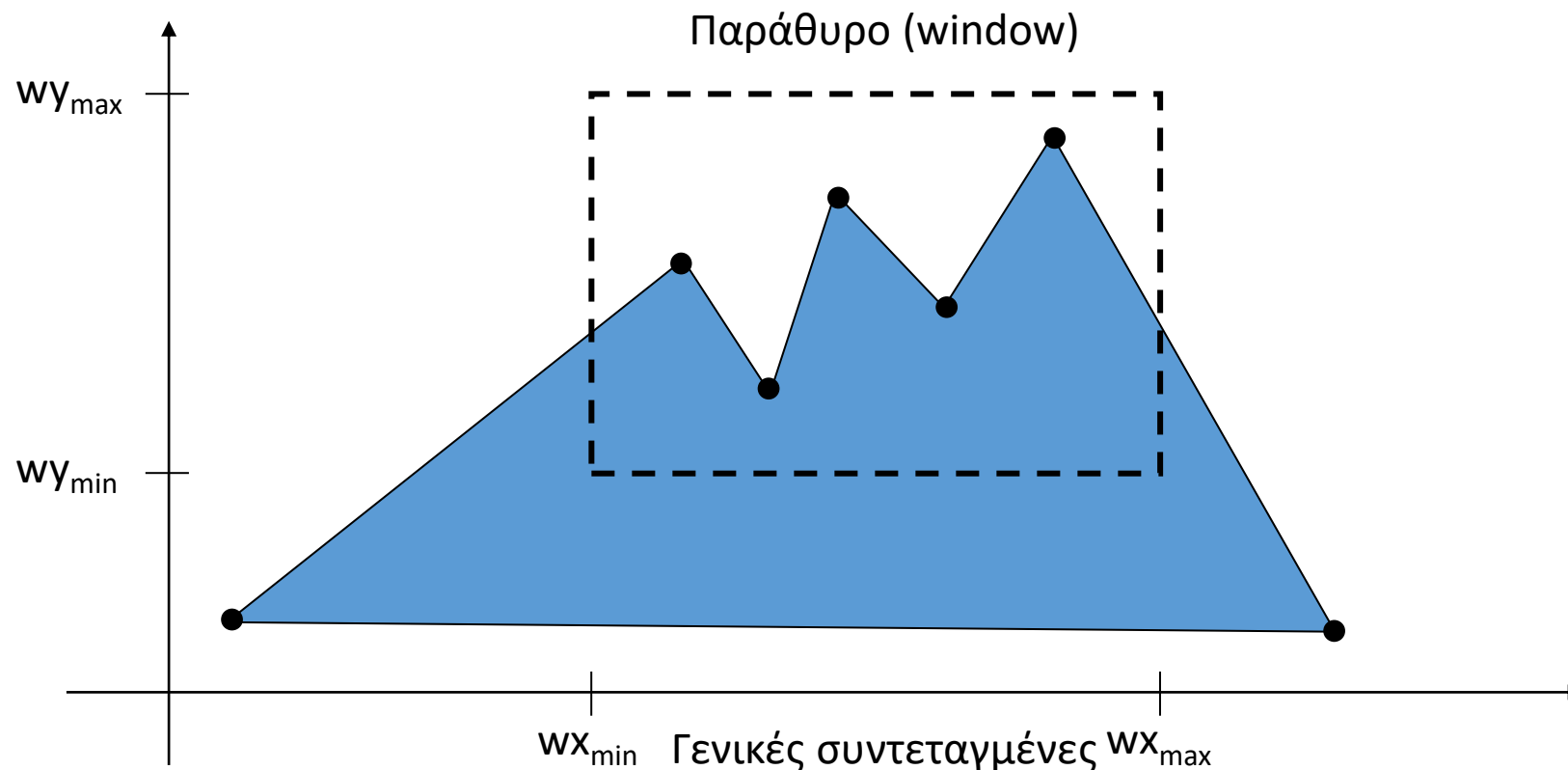
Windowing I

- Μια σκηνή αποτελείται από μια συλλογή αντικειμένων που προσδιορίζονται στις γενικές συντεταγμένες.



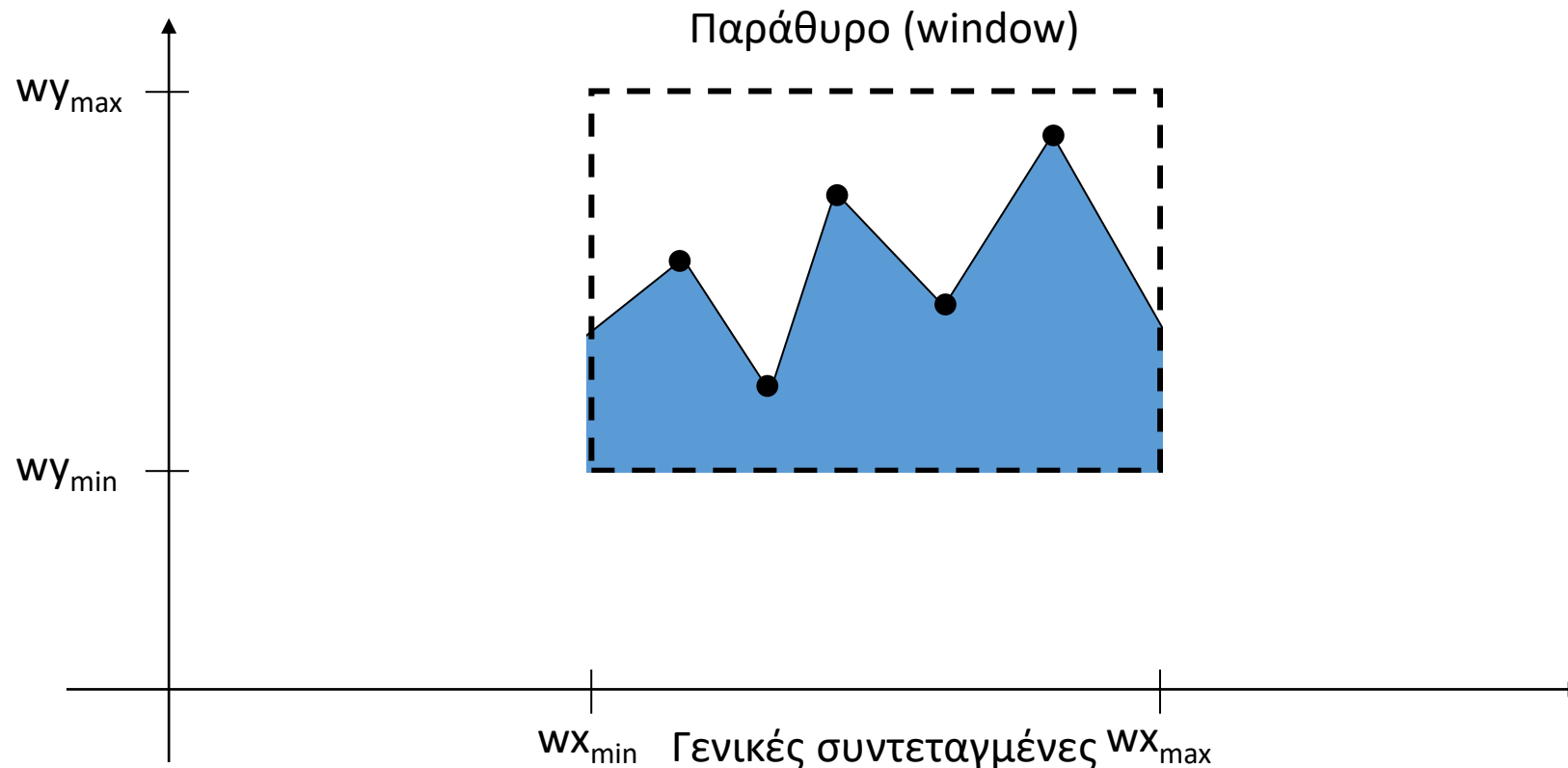
Windowing II

- Κατά την προβολή μιας σκηνής, εμφανίζονται μόνο αυτά τα αντικείμενα τα οποία βρίσκονται μέσα σε ένα συγκεκριμένο παράθυρο



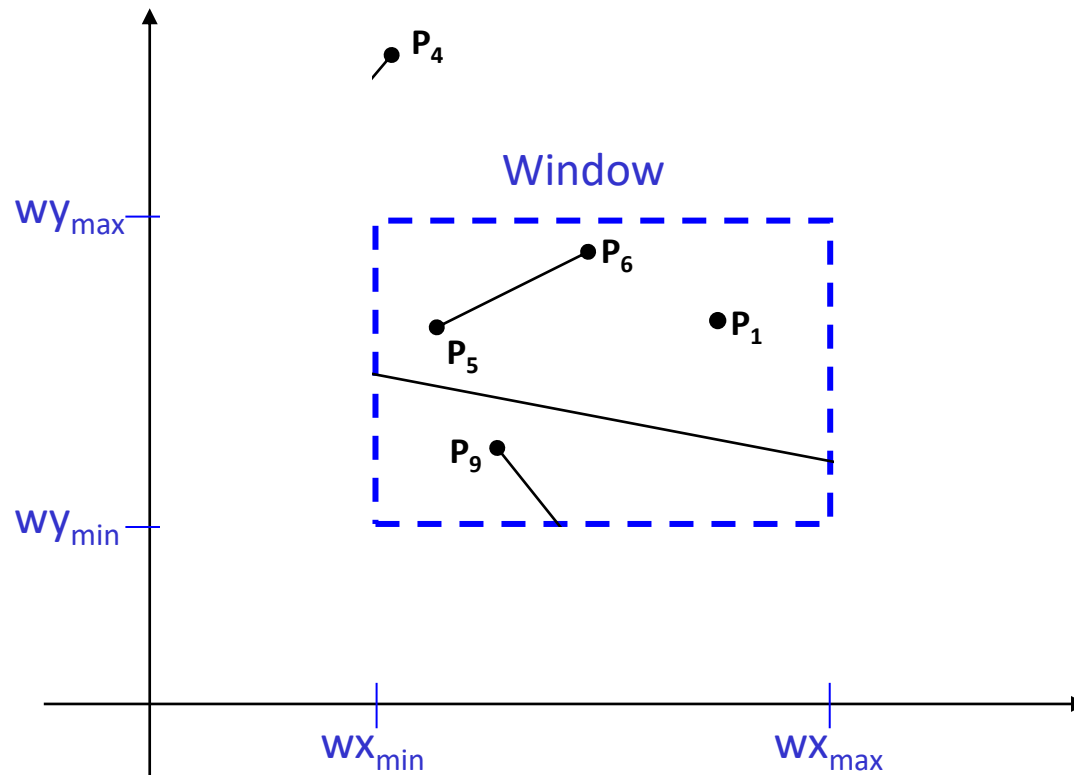
Windowing III

- Επειδή η σχεδίαση αντικειμένων σε μια οθόνη απαιτεί χρόνο, δεν ασχολούμαστε με τα αντικείμενα έξω από το παράθυρο



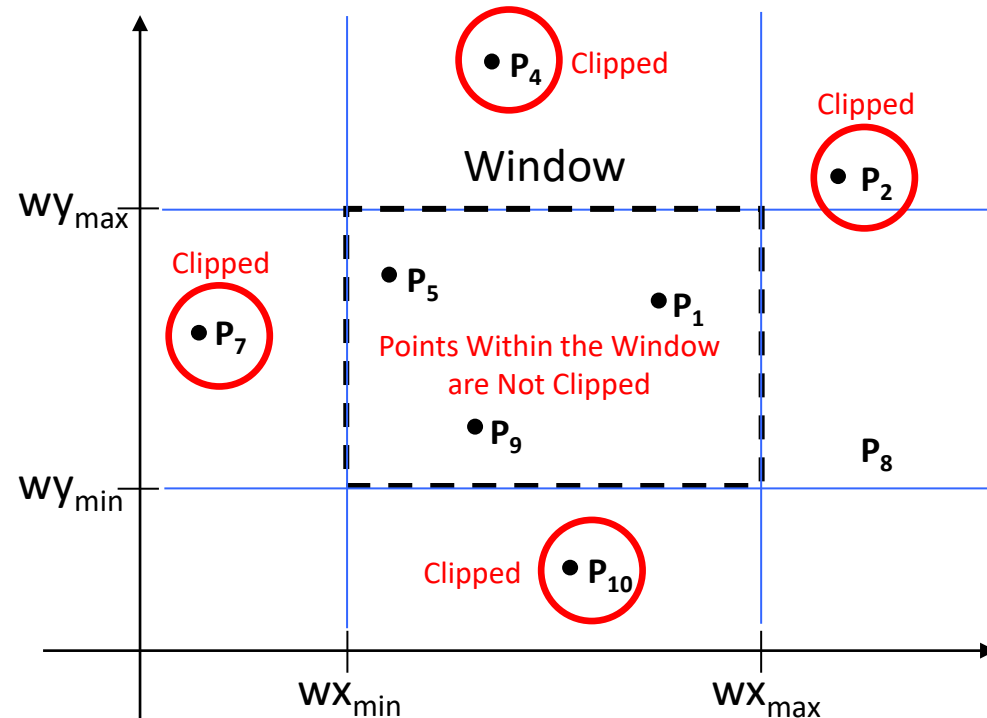
Αποκοπή (Clipping)

- Για την παρακάτω εικόνα προσδιορίστε ποιες γραμμές και σημεία πρέπει να διατηρηθούν, και ποια θα πρέπει να αποκοπούν



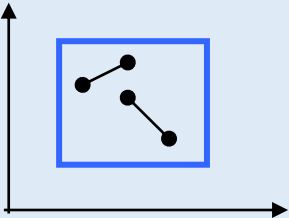
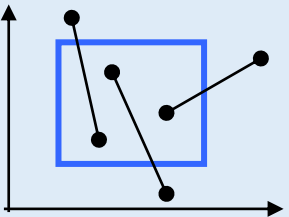
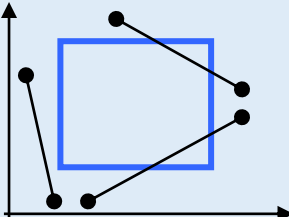
Αποκοπή Σημείου (Point Clipping)

- Εύκολο – ένα σημείο (x,y) δεν θα αποκοπεί αν:
- $wx_{min} \leq x \leq wx_{max}$ ΚΑΙ $wy_{min} \leq y \leq wy_{max}$
- αλλιώς θα αποκοπεί



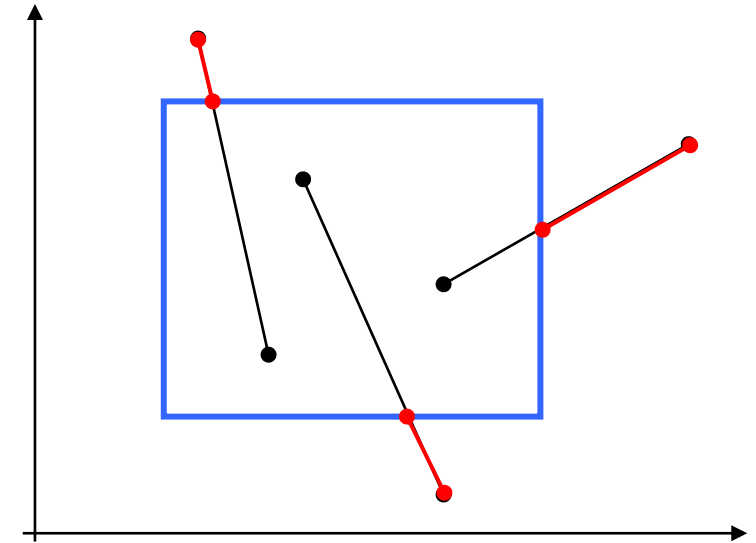
Αποκοπή Ευθείας Γραμμής (Line Clipping)

- Δύσκολο - να εξετάσετε τα τελικά σημεία κάθε γραμμής για να δείτε αν βρίσκονται εντός του παραθύρου ή όχι

Κατάσταση	Λύση	Παράδειγμα
Και τα δύο σημεία τέλους μέσα στο παράθυρο	Καμία αποκοπή	
Ένα τελικό σημείο μέσα στο παράθυρο, ένα έξω	Αποκοπή	
Και τα δύο σημεία τέλους έξω από το παράθυρο	Δεν γνωρίζουμε!	

Αποκοπή Γραμμής (Brute Force Line Clipping)

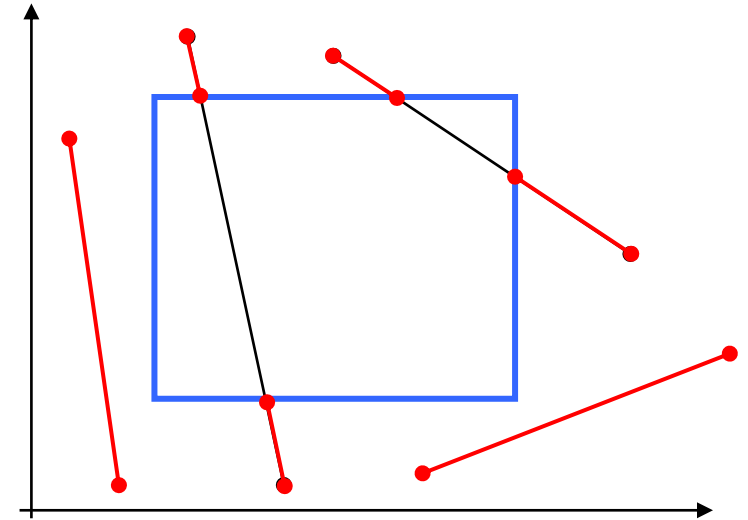
- Η αποκοπή μια γραμμής δύναμης μπορεί να εκτελεστεί ως εξής:
 - Να μην γίνεται αποκοπή γραμμών εάν και τα δύο σημεία τέλους βρίσκονται μέσα στο παράθυρο
 - Για γραμμές με ένα τελικό σημείο μέσα στο παράθυρο και ένα τελικό σημείο έξω, υπολογίστε το σημείο τομής (χρησιμοποιώντας την εξίσωση της γραμμής) και αποκόψτε από αυτό το σημείο και έξω



Αποκοπή Γραμμής (Brute Force Line Clipping)

- Η αποκοπή μια γραμμής μπορεί να εκτελεστεί ως εξής:
 - Για γραμμές με τα δύο τελικά σημεία έξω από το παράθυρο, ελέγξτε τη τομή της γραμμής με όλα τα όρια του παραθύρου, και αποκόψτε κατάλληλα

Ωστόσο, ο υπολογισμός των σημείων τομής γραμμής είναι υπολογιστικά ακριβό. Λαμβάνοντας υπόψη πως μια σκηνή μπορεί να περιέχει πολλές γραμμές, αυτή η προσέγγιση για αποκοπή είναι πολύ αργή...



Αλγόριθμος αποκοπής Cohen-Sutherland

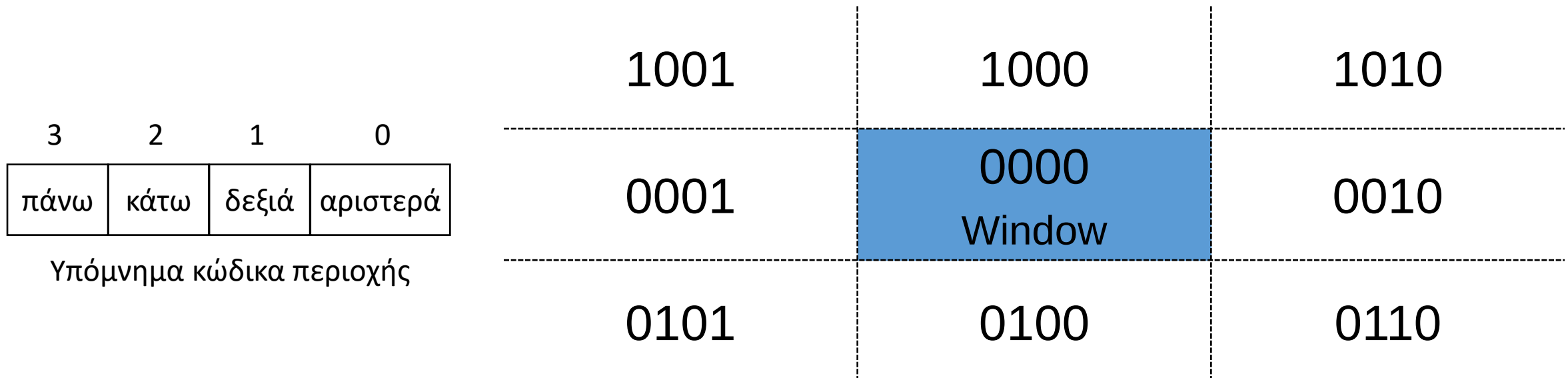
- Ένας αποδοτικός αλγόριθμος αποκοπής
- Το βασικό πλεονέκτημα του αλγορίθμου είναι ότι μειώνει σημαντικά τον αριθμό των γραμμών που πρέπει να υπολογίζονται



Dr. Ivan E. Sutherland co-developed the Cohen-Sutherland clipping algorithm. Sutherland is a graphics giant and includes amongst his achievements the invention of the head mounted display.

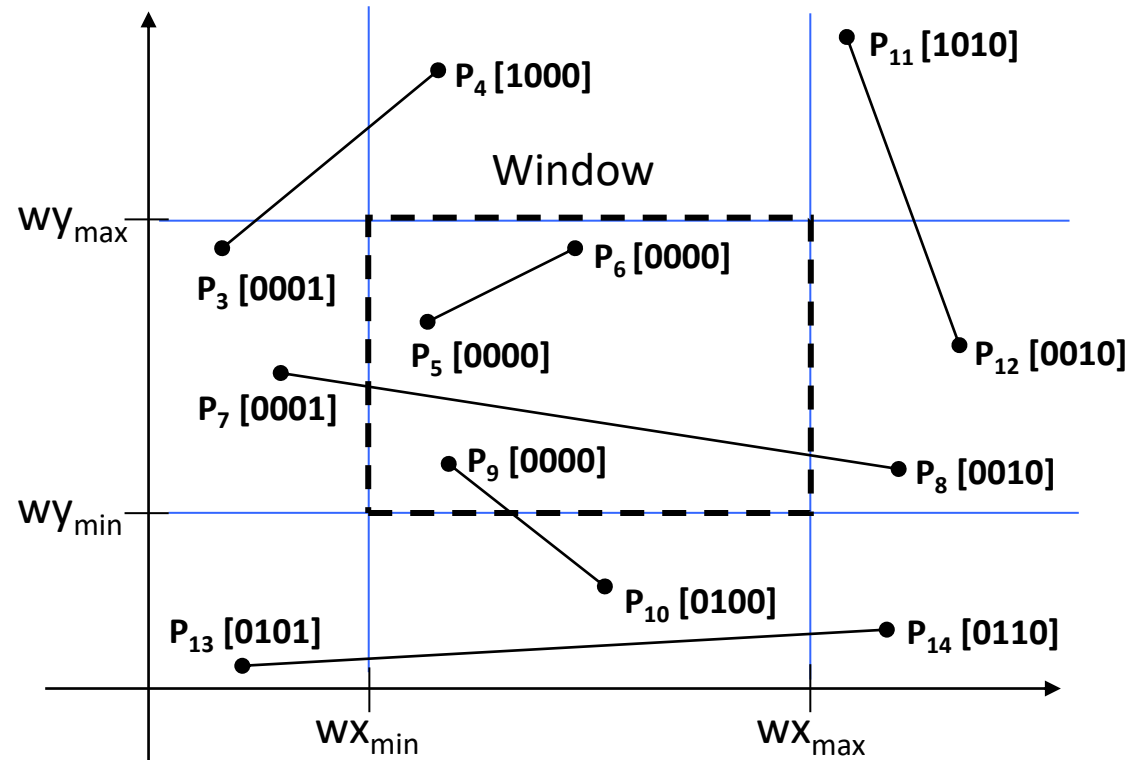
Cohen-Sutherland: Διαχωρισμός του κόσμου

- Ο παγκόσμιος χώρος χωρίζεται σε περιοχές με βάση τα όρια του παραθύρου
 - Κάθε περιοχή έχει έναν μοναδικό κωδικό περιοχής τεσσάρων bit
 - Οι κωδικοί περιοχής υποδεικνύουν τη θέση των περιφερειών σε σχέση με το παράθυρο
 - Μόνο η κεντρική περιοχή είναι ορατή



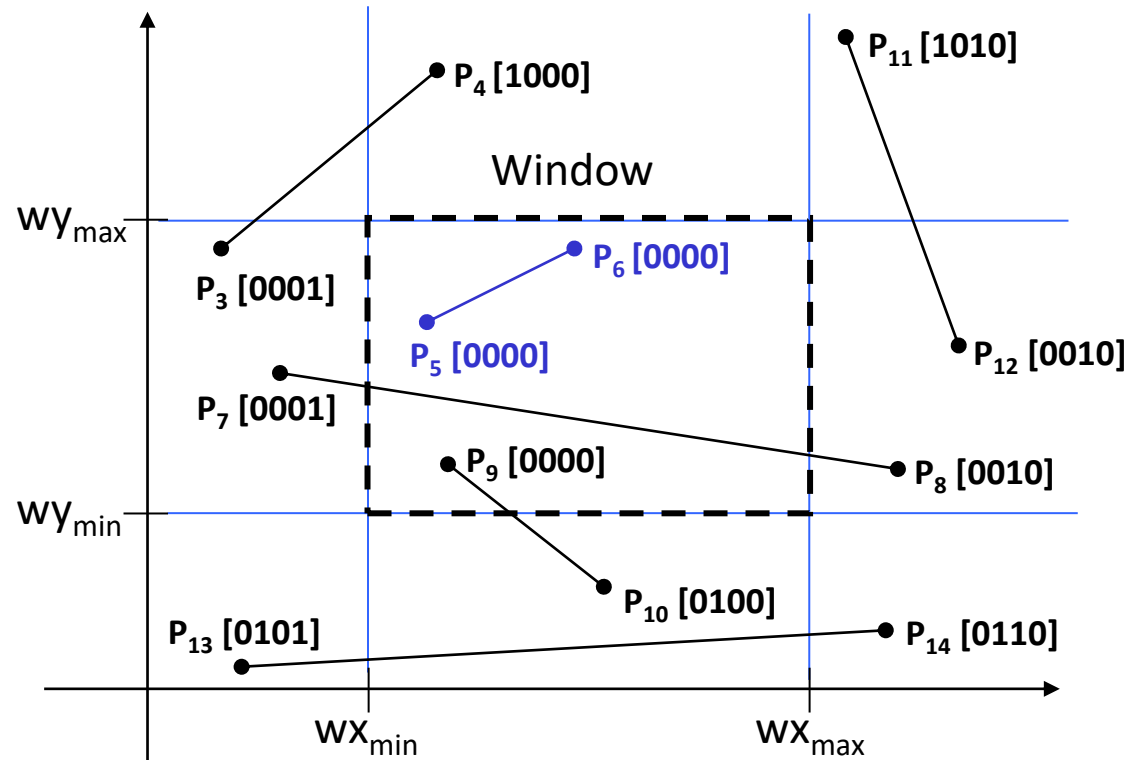
Cohen-Sutherland: Επισήμανση

- Κάθε τελικό σημείο επισημαίνεται με τον κατάλληλο κωδικό περιοχής



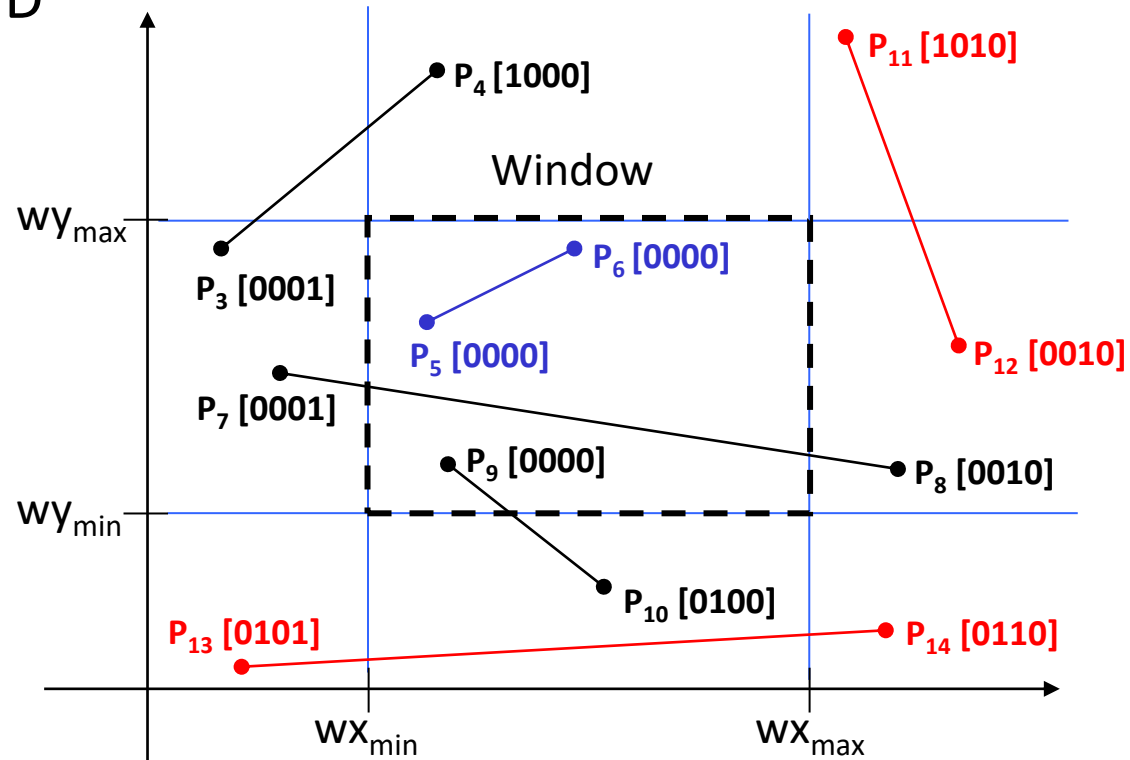
Cohen-Sutherland: Γραμμές μέσα στο παράθυρο

- Οι γραμμές που περιέχονται πλήρως στα όρια του παραθύρου έχουν κωδικό περιοχής [0000] και για τα δύο σημεία τέλους, επομένως δεν αποκόπτονται



Cohen-Sutherland: Γραμμές έξω από το παράθυρο

- Μπορείτε να περικόπτεται οποιαδήποτε γραμμή όταν υπάρχει τουλάχιστον ένα κοινό στοιχείο «1» στους κωδικούς περιοχής τους
 - Λογική πράξη AND



Cohen-Sutherland: Υπόλοιπες γραμμές

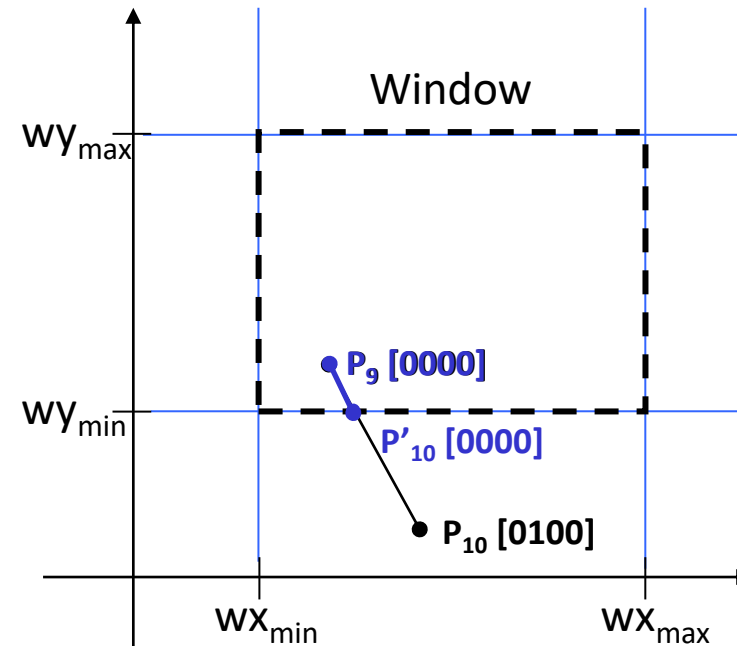
- Οι γραμμές που δεν μπορούν να αναγνωριστούν πλήρως εάν είναι μέσα ή έξω από το παράθυρο, μπορεί να διασχίζουν το εσωτερικό του παραθύρου
- Οι γραμμές αυτές επεξεργάζονται ως εξής:
 - Σύγκρινε ένα σημείο εκτός του κύκλου με τα πλαίσιο του παραθύρου (επέλεξε οποιοδήποτε όριο, πάνω κάτω, δεξιά, αριστερά), και προσδιόρισε σε ποιο βαθμό θα πρέπει να αποκοπεί
 - Εάν το υπόλοιπο της γραμμής είναι εντελώς μέσα ή έξω από το παράθυρο, κρατήστε το ή αποκόψτε το αντίστοιχα
 - Διαφορετικά, συγκρίνετε το υπόλοιπο της γραμμής με τα άλλα όρια του παραθύρου
 - Συνεχίστε μέχρι να απορριφθεί η γραμμή ή να βρεθεί ένα τμήμα μέσα στο παράθυρο

Cohen-Sutherland: Υπόλοιπες γραμμές

- Μπορούμε να χρησιμοποιήσουμε τους κωδικούς περιοχής για να καθορίσουμε ποια όρια παραθύρων πρέπει να εξεταστούν για πιθανή τομή
 - Για να ελέγξετε αν μια γραμμή διασχίζει ένα συγκεκριμένο όριο, συγκρίνετε τα κατάλληλα bits στους κωδικούς περιοχής των τελικών σημείων τους
 - Εάν το ένα από αυτά τα στοιχεία είναι 1 και το άλλο είναι 0, τότε η γραμμή διασχίζει το παράθυρο

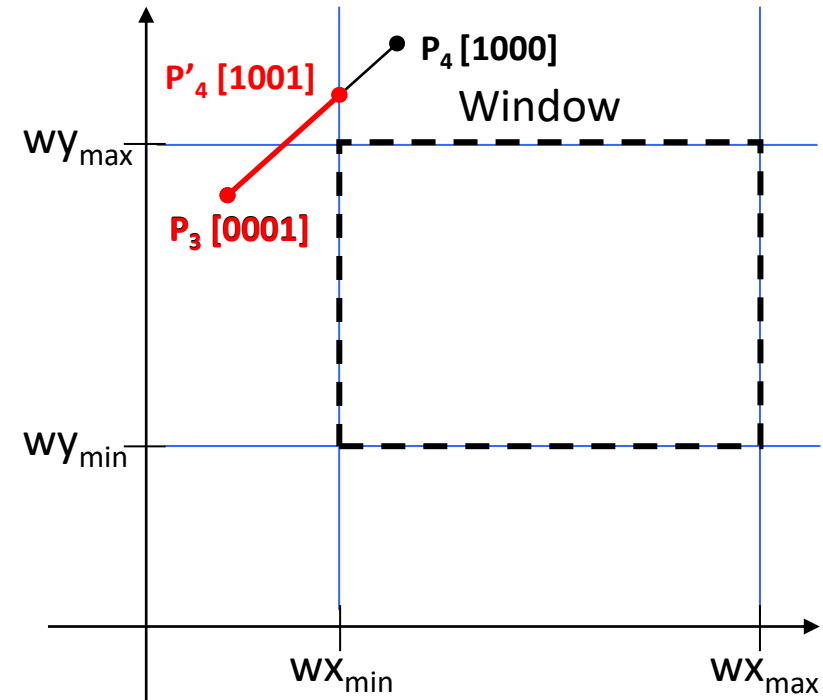
Cohen-Sutherland: Παραδείγματα

- Έστω η ευθεία από το σημείο P_9 στο P_{10}
 - Ξεκινήστε από το P_{10}
 - Από τους κωδικούς περιοχής των δύο τελικών σημείων γνωρίζουμε ότι η γραμμή **δεν διασχίζει** το αριστερό ή το δεξιό παράθυρο
 - Υπολογίστε την τομή της γραμμής με το κάτω όριο για να δημιουργήσετε ένα νέο σημείο P'_{10}
 - Η γραμμή P_9 το P'_{10} είναι εντελώς μέσα στο παράθυρο, έτσι δεν θα αποκοπεί



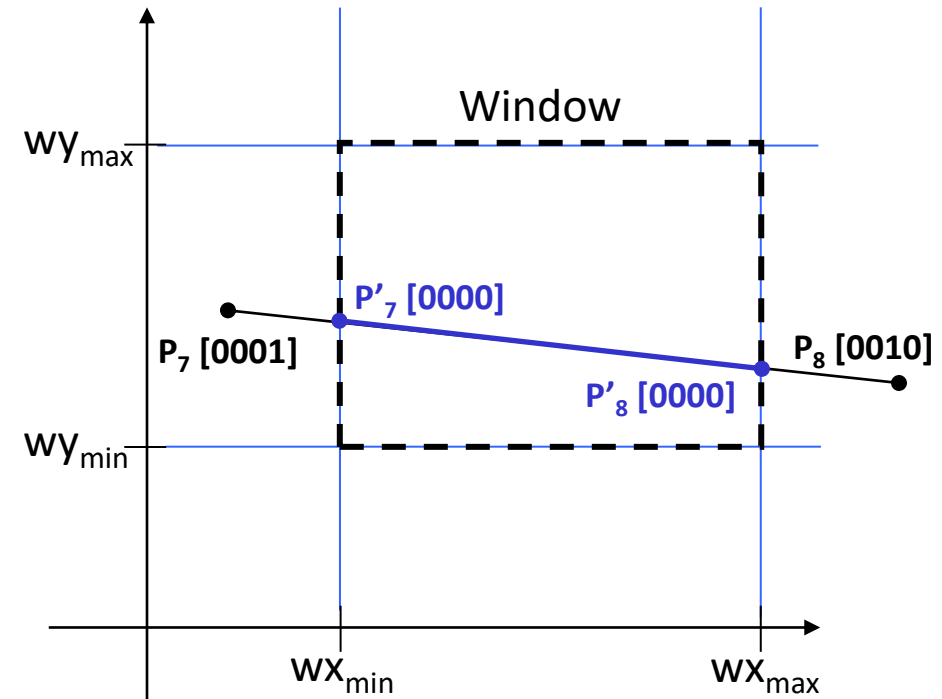
Cohen-Sutherland: Παραδείγματα

- Έστω η γραμμή από το σημείο P_3 στο P_4
 - Ξεκινήστε από το P_4
 - Από τους κωδικούς περιοχής των δύο τελικών σημείων γνωρίζουμε ότι η γραμμή **διασχίζει** το αριστερό παράθυρο στο P'_4
 - Η γραμμή P_3 στο P'_4 είναι εντελώς έξω από το παράθυρο έτσι αποκόπτεται
 - Το ίδιο και η γραμμή P_4 στο P'_4



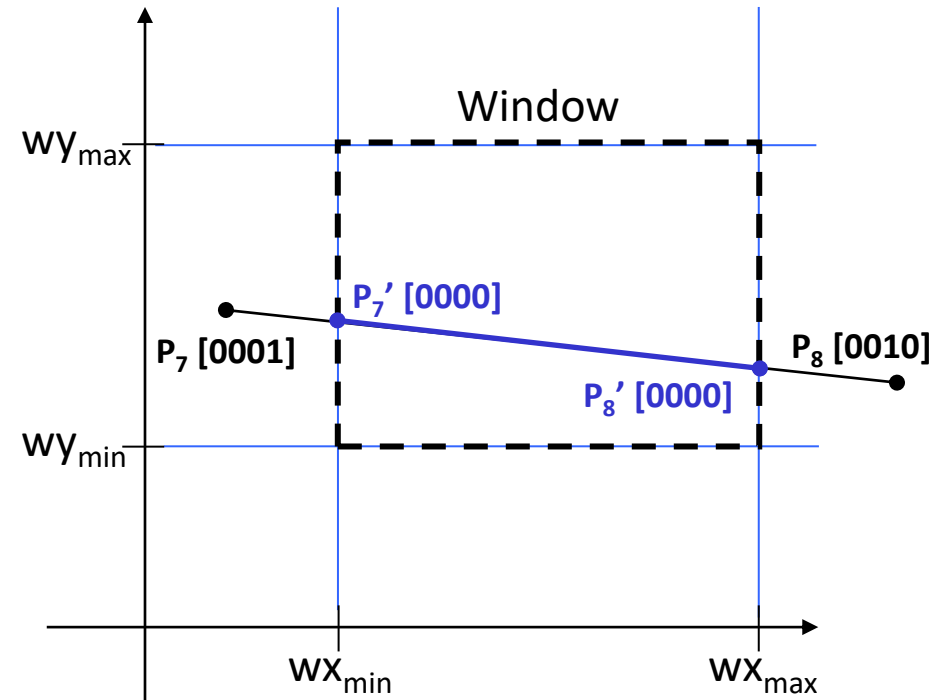
Cohen-Sutherland: Παραδείγματα

- Έστω η γραμμή από το σημείο P_7 στο P_8
 - Ξεκινήστε από το P_7
 - Από τους κωδικούς περιοχής των δύο τελικών σημείων γνωρίζουμε ότι η γραμμή **διασχίζει** το αριστερό παράθυρο στο P'_7



Cohen-Sutherland: Παραδείγματα

- Έστω η γραμμή από το σημείο P'_7 στο P_8
 - Ξεκινήστε από το P_8
 - Υπολογίστε τη τομή με το παράθυρο για να δημιουργήσετε P'_8
 - Τόσο το P'_7 όσο και το P'_8 είναι μέσα στο παράθυρο, έτσι δεν αποκόπτεται



Υπολογισμός της τομής των γραμμών

- Τα σημεία τομής με τα όρια του παραθύρου υπολογίζονται χρησιμοποιώντας τις γνωστές παραμέτρους της εξίσωσης γραμμών
- Έστω μια γραμμή με τα τελικά σημεία (x_1, y_1) και (x_2, y_2)
 - Η y -συντεταγμένη της τομής μιας γραμμής με ένα κατακόρυφο όριο παραθύρου μπορεί να υπολογιστεί με τη χρήση της εξίσωσης:

$$y = y_1 + m (x_{boundary} - x_1)$$

όπου $x_{boundary}$ μπορεί να οριστεί σαν το wx_{min} ή το wx_{max}

Υπολογισμός της τομής των γραμμών

- Αντίστοιχα, Η x-συντεταγμένη της τομής μιας γραμμής με ένα οριζόντιο όριο παραθύρου μπορεί να υπολογιστεί με τη χρήση της εξίσωσης

$$x = x_1 + (y_{\text{boundary}} - y_1) / m$$

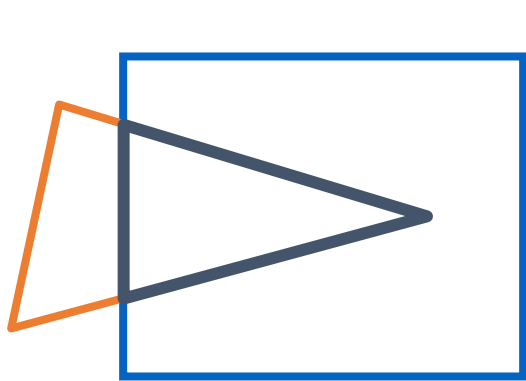
όπου y_{boundary} μπορεί να οριστεί σαν το $wy_{\min}$ ή $wy_{\max}$

- m είναι η κλίση της εν λόγω γραμμής και μπορεί να υπολογιστεί σαν

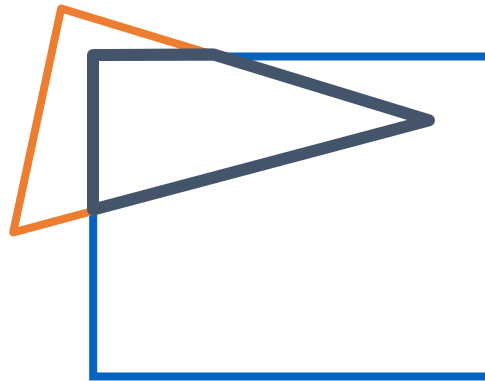
$$m = (y_2 - y_1) / (x_2 - x_1)$$

Γιατί η αποκοπή είναι δύσκολή;

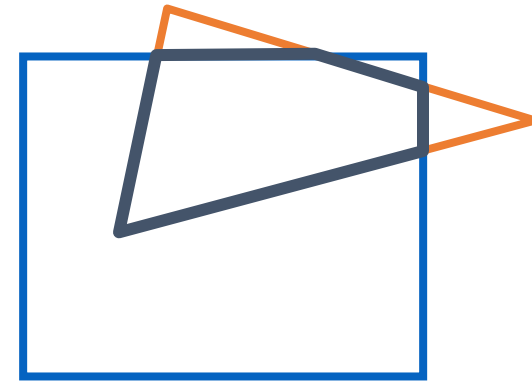
- Τι μπορεί να συμβεί σε ένα τρίγωνο κατά την αποκοπή;
- Πιθανά αποτελέσματα:



triangle-to-triangle



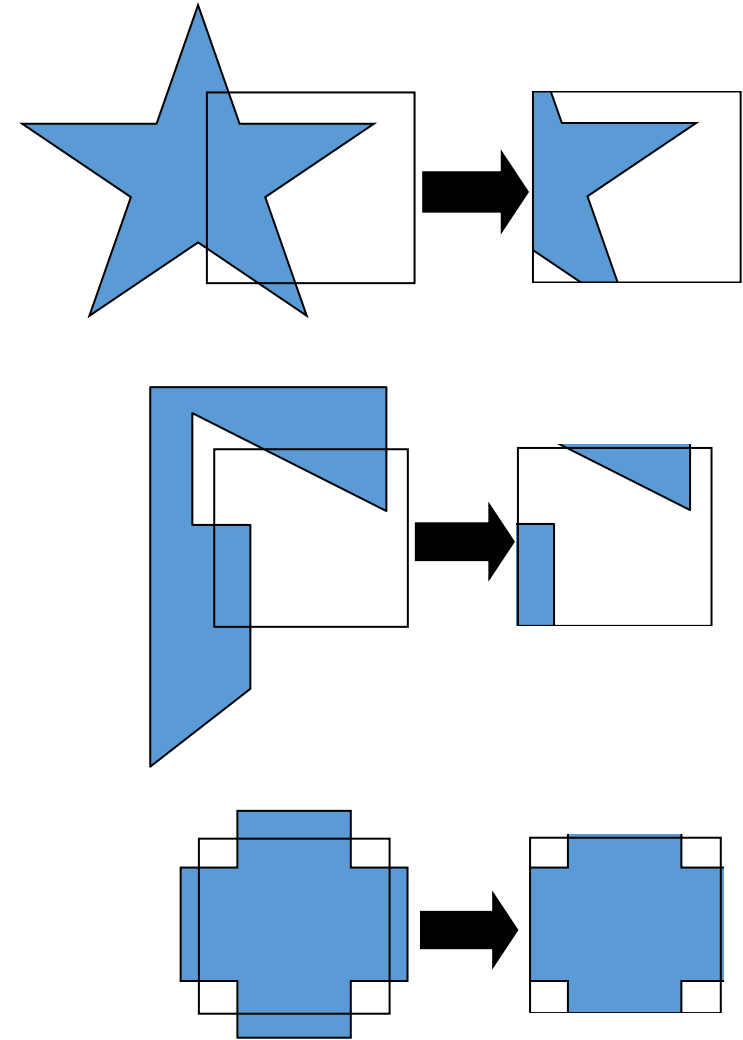
triangle-to-quad



triangle-to-5-gon

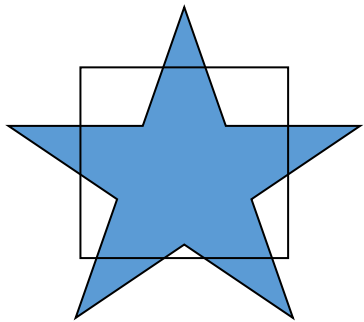
Αποκοπή περιοχής

- Παρόμοια με τις γραμμές, οι περιοχές πρέπει να αποκοπούν σε ένα όριο παραθύρου
- Πρέπει να ληφθούν υπόψη τα τμήματα της περιοχής που πρέπει να αποκοπούν

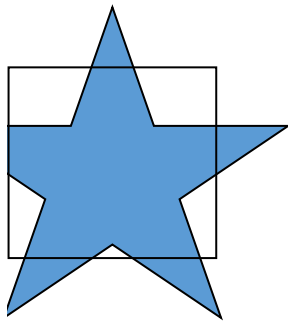


Sutherland-Hodgman: Αλγόριθμος αποκοπής περιοχής

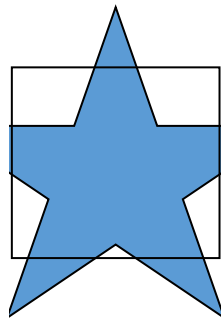
- Οι Sutherland & Hodgman ανέπτυξαν μια νέα τεχνική για τις αποκοπή περιοχών
- Τα πολύγωνα αποκόπτονται εξετάζοντας κάθε όριο παράθυρου στη σειρά



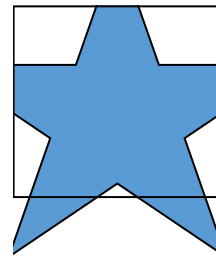
Original Area



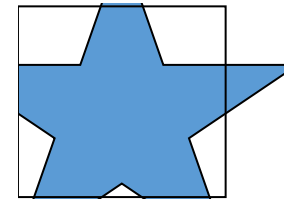
Clip Left



Clip Right



Clip Top



Clip Bottom

Sutherland turns up again. This time with



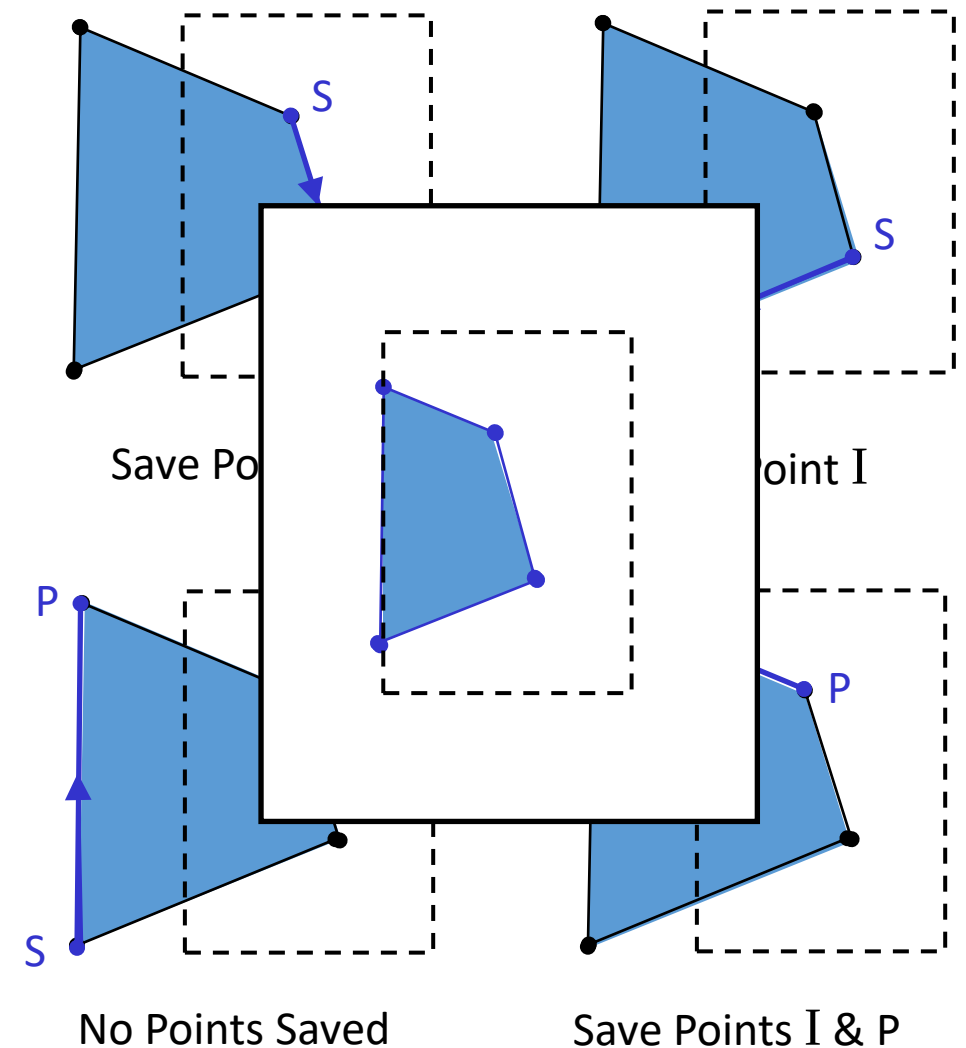
Gary Hodgman with whom he worked at the first ever graphics company Evans & Sutherland

Sutherland-Hodgman: Αλγόριθμος αποκοπής περιοχής

- Για να αποκόψετε μια περιοχή σε ένα όριο παραθύρου:
 - Εξετάστε κάθε κορυφή, με τη σειρά, έναντι του ορίου
 - Οι κορυφές που είναι μέσα στο όριο αποθηκεύονται για αποκοπή στην επόμενη τομή με το όριο
 - Οι κορυφές έξω από το όριο περικόπτονται
 - Αν προχωρήσουμε από ένα σημείο μέσα στο όριο σε ένα σημείο έξω, η τομή της γραμμής τους με το όριο αποθηκεύεται
 - Αν περάσουμε από το εξωτερικό στο εσωτερικό του παραθύρου, τότε το σημείο τομής θα αποθηκευτεί

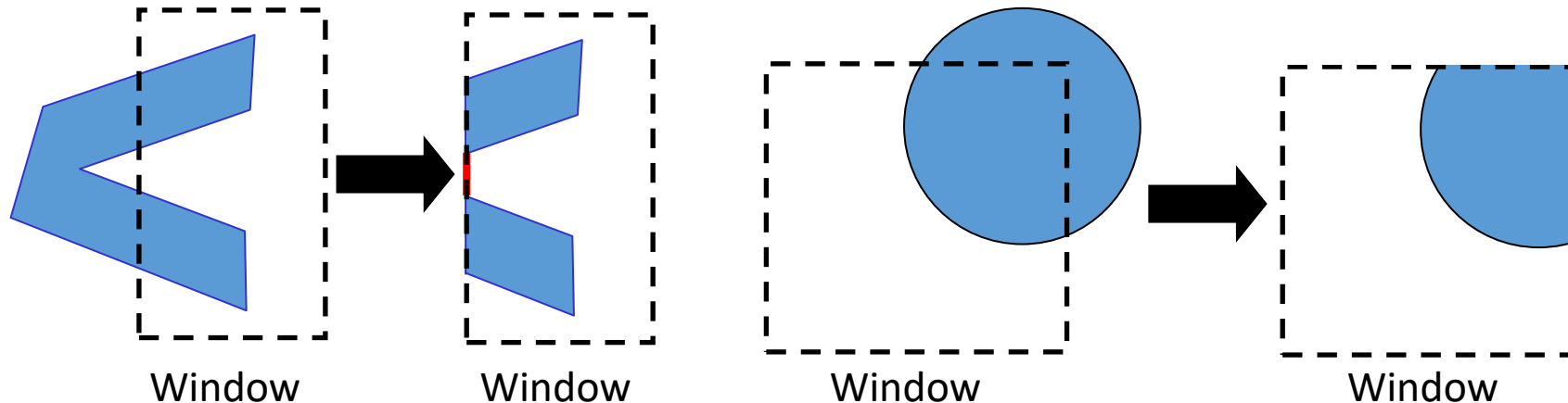
Sutherland-Hodgman: Παράδειγμα

- Κάθε παράδειγμα δείχνει το σημείο που γίνεται επεξεργασία (P) και το προηγούμενο σημείο (S)
- Τα αποθηκευμένα σημεία καθορίζουν την περιοχή που αποκόπτεται στο συγκεκριμένο όριο



Αποκοπή σε άλλα σχήματα

- Η αποκοπή σε τέτοιες περιοχές είναι λίγο πιο δύσκολη καθώς συχνά περισεύουν γραμμές που θα πρέπει να αποκοπούν
 - Όταν έχουμε μη-κυρτά πολύγωνα τότε ο πιο πάνω αλγόριθμος μπορεί να παράγει πολύγωνα με συμπίπτων ακμές
 - Αυτό είναι εντάξει για το σχεδιασμό (rendering) αλλά ίσως όχι για άλλες εφαρμογές (π.χ. σκιές)
 - Ο Weiler-Atherton αλγόριθμος παράγει ξεχωριστά πολύγωνα για κάθε ορατό κομμάτι πολυγώνου



- Η αποκοπή σε κοίλα σχήματα θέλει περισσότερη δουλειά
 - π.χ., για κύκλους, πρέπει να βρούμε την τομή κύκλου με ευθεία

Αποκοπή πολυγώνων σε 3D

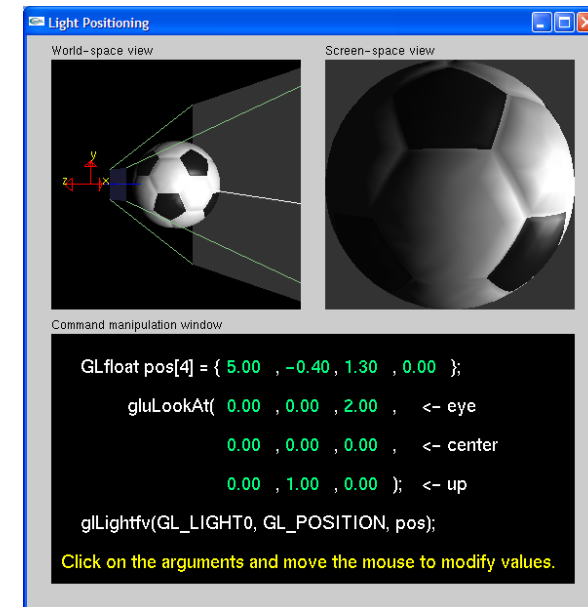
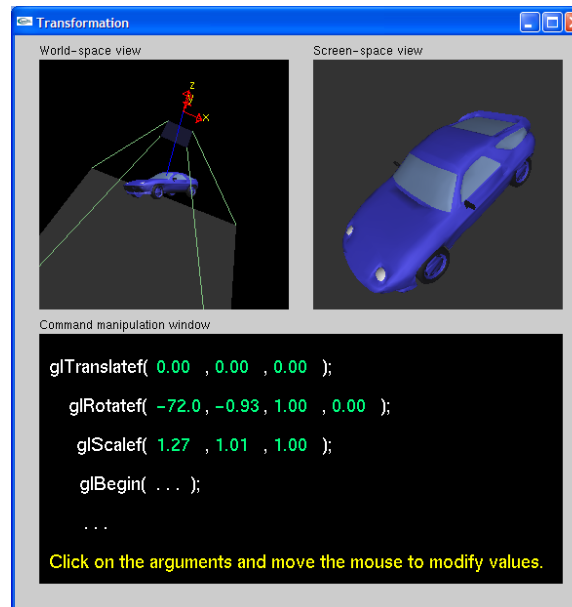
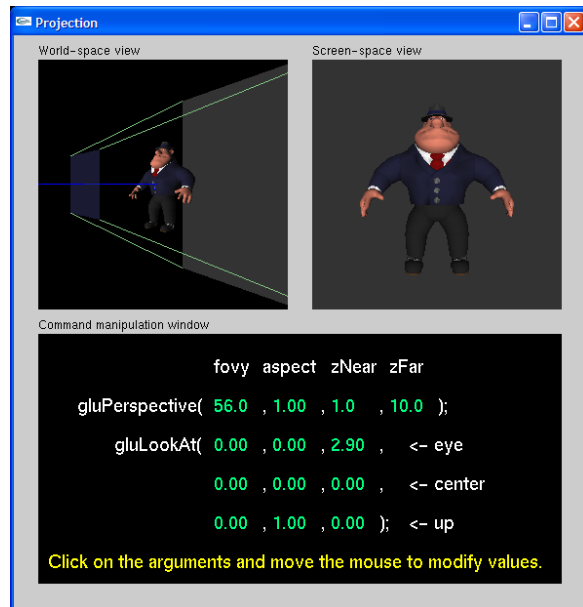
- Ο αλγόριθμος Sutherland-Hodgman μπορεί εύκολα να επεκταθεί σε 3D
 - τα όρια αποκοπής είναι 6 επίπεδα αντί 4 γραμμών
 - η εύρεση τομής γίνεται συγκρίνοντας μια ακμή προς ένα επίπεδο αντί σύγκρισης ακμής προς ακμή

Περίληψη

- Τα αντικείμενα μέσα σε μια σκηνή πρέπει να αποκοπούν για να εμφανιστεί μόνο η «χρήσιμη» σκηνή σε ένα παράθυρο
- Επειδή υπάρχουν πολλά αντικείμενα σε μια σκηνή, οι αλγόριθμοι αποκοπής πρέπει να είναι εξαιρετικά αποδοτικοί.
- Ο αλγόριθμος Cohen-Sutherland μπορεί να χρησιμοποιηθεί για αποκοπή γραμμής
- Ο αλγόριθμος Sutherland-Hodgman μπορεί να χρησιμοποιηθεί για αποκοπή μιας περιοχής

Nate Robins' OpenGL Tutorials

- Nate Robins έχει μια σειρά από φροντιστήρια (tutorials) για χρήση της βιβλιοθήκης OpenGL



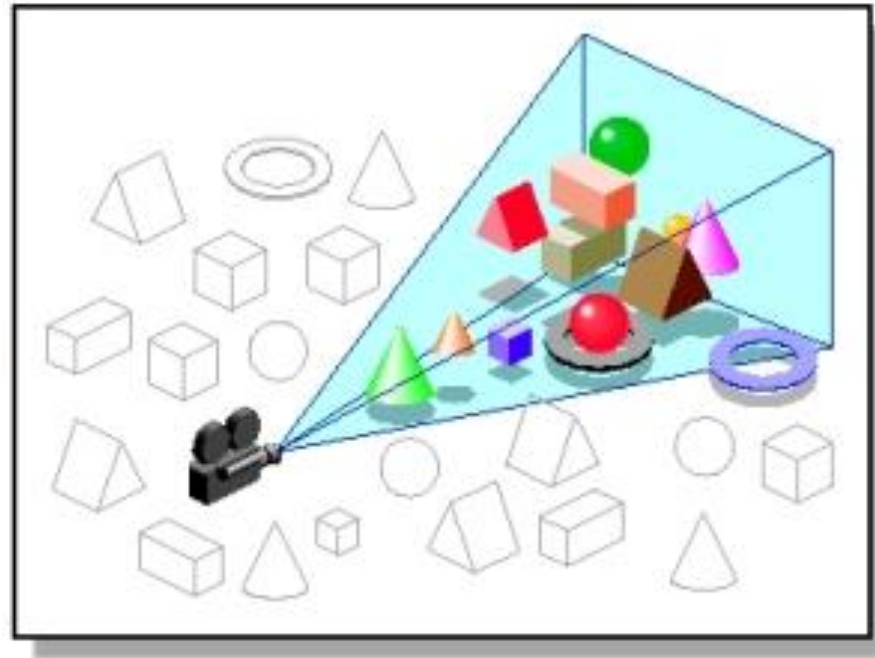
Nate Robin's OpenGL Tutorials available at: <http://www.xmission.com/~nate/tutors.html>

3Δ Αποκοπή

- Ακριβώς όπως και στην περίπτωση των δύο διαστάσεων, η αποκοπή αφαιρεί αντικείμενα που δεν θα είναι ορατά στη σκηνή
 - Σκοπός είναι να μειωθεί ο χρόνος επεξεργασίας
- Η 3Δ αποκοπή επιτυγχάνεται σε 2 φάσεις
 - Αποκοπή/Απόρριψη αντικειμένων που δεν είναι ορατά στην κάμερα
 - δηλαδή αντικείμενα που βρίσκονται πίσω από την κάμερα, έξω από το οπτικό πεδίο, ή είναι πολύ μακριά
 - Αποκοπή αντικειμένων που τέμνονται με οποιοδήποτε από τα επίπεδα αποκοπής

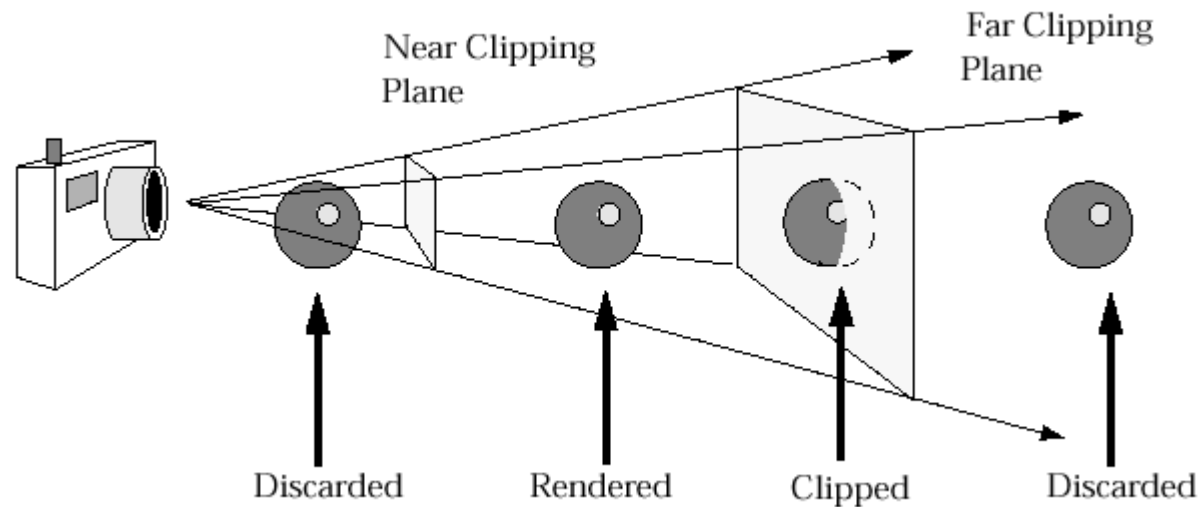
Αποκοπή/Απόρριψη Αντικειμένων

- Η απόρριψη αντικειμένων που δεν είναι ορατά περιλαμβάνει τη σύγκριση τους με ένα πλαίσιο που οριοθετεί τις διαστάσεις του οπτικού πεδίου (την πυραμίδα προβολής)
 - Μπορεί να γίνει πριν ή και μετά την προβολή



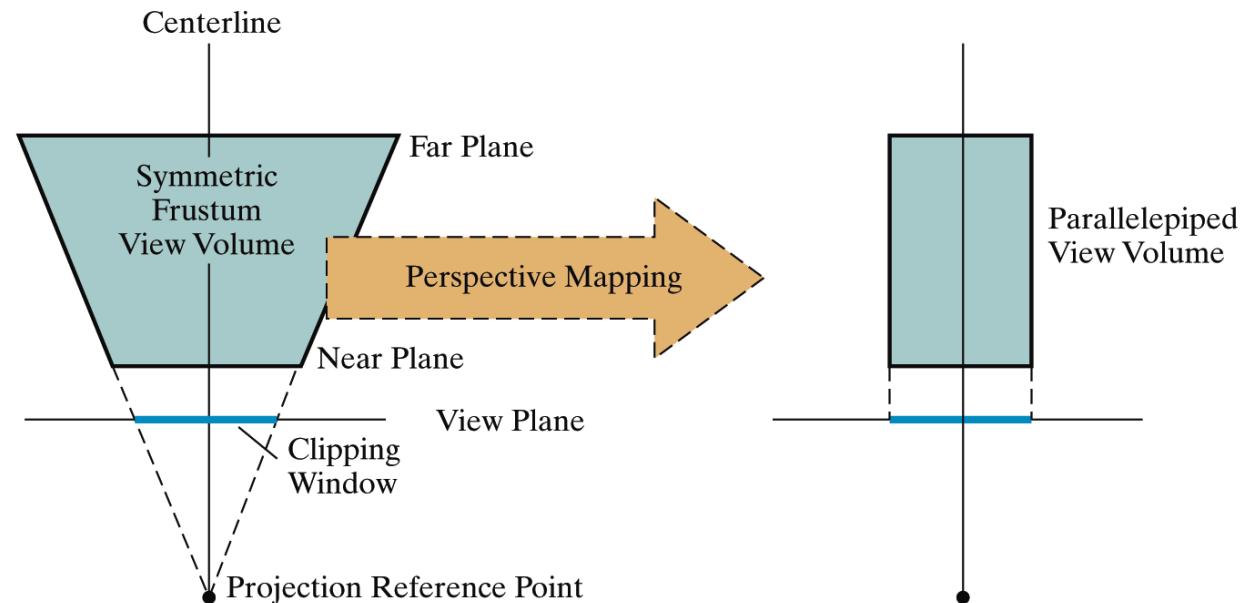
Αποκοπή αντικειμένων

- Τα αντικείμενα που είναι μερικώς εντός του οπτικού πεδίου πρέπει να αποκόπτονται, όπως αντίστοιχα και στην περίπτωση των 2Δ



Ο όγκος αποκοπής (the clipping volume)

- Αφού εφαρμοστεί προβολή με προοπτική, το σχήμα του οπτικού πεδίου (όγκος) θα έχει μετατραπεί σε παραλληλόγραμμα - θυμηθείτε ότι διατηρήσαμε όλες τις πληροφορίες για τις z-συντεταγμένες (βάθος)



Πότε κάνουμε αποκοπή;

- Εκτελούμε την αποκοπή αφού ολοκληρωθεί ο μετασχηματισμός προβολής και η κανονικοποίηση (στον αρχικό άξονα συντεταγμένων) των δεδομένων μας
- Έτσι, έχουμε τα εξής:

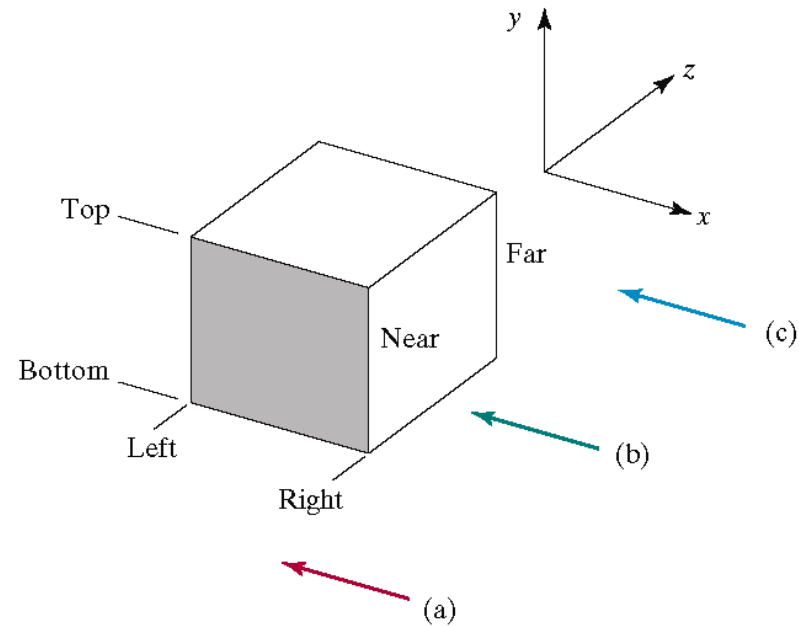
$$\begin{bmatrix} x_h \\ y_h \\ z_h \\ h \end{bmatrix} = M \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Διαίρεση του κόσμου

- Παρόμοια με την περίπτωση των 2Δ, χωρίζουμε τον κόσμο σε περιοχές
- Αυτή τη φορά χρησιμοποιούμε 6 bit για κάθε κωδικό περιοχής, οπότε έχουμε 27 διαφορετικούς κωδικούς περιοχής
- Οι κωδικοί bits για αυτές τις περιοχές είναι οι εξής:

bit 6 Far	bit 5 Near	bit 4 Top	bit 3 Bottom	bit 2 Right	bit 1 Left
--------------	---------------	--------------	-----------------	----------------	---------------

Κωδικοί περιοχής



011001	011000	011010
010001	010000	010010
010101	010100	010110

Region Codes
In Front of Near Plane
(a)

001001	001000	001010
000001	000000	000010
000101	000100	000110

Region Codes
Between Near and Far Planes
(b)

101001	101000	101010
100001	100000	100010
100101	100100	100110

Region Codes
Behind Far Plane
(c)

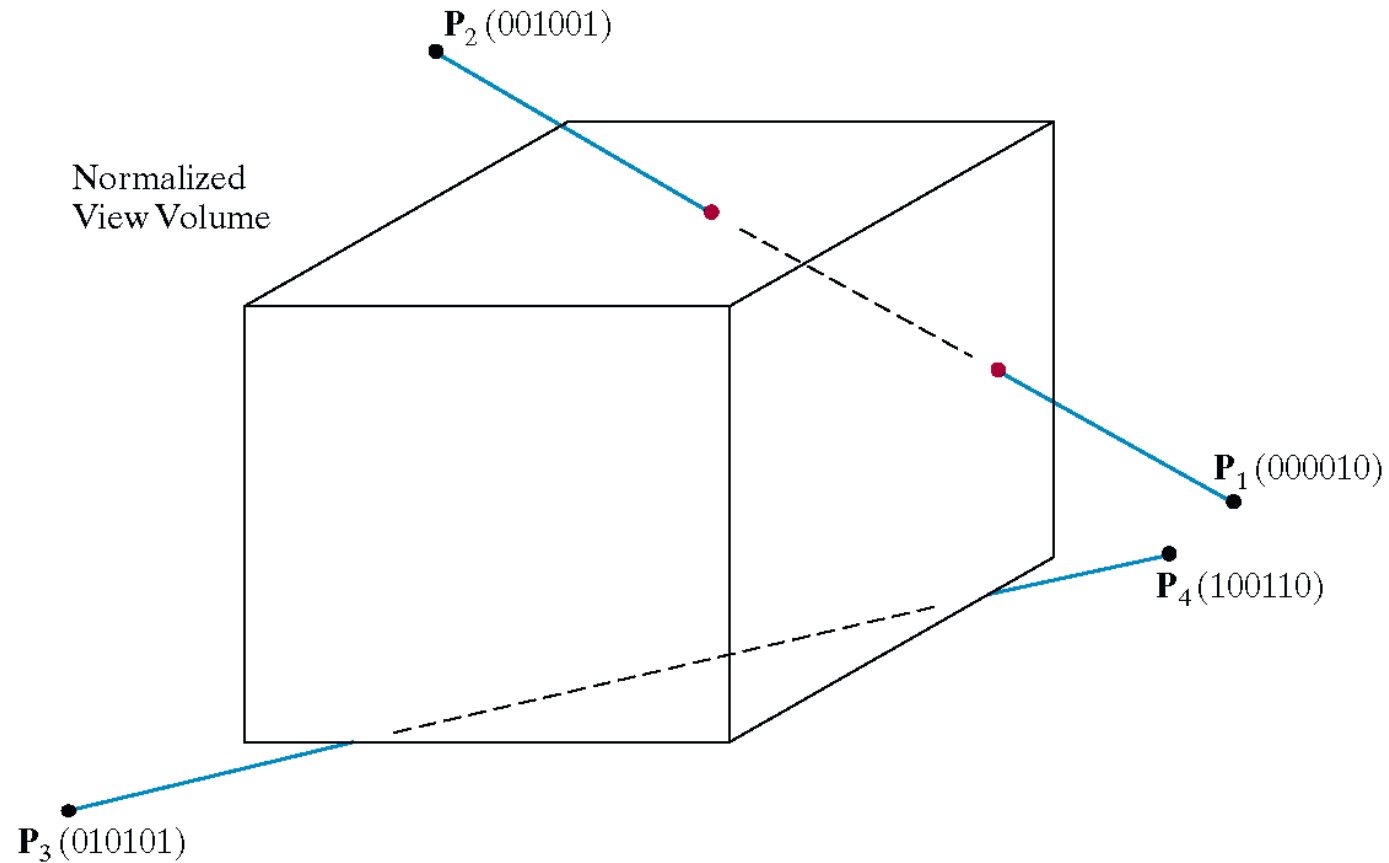
Αποκοπή σημείου

- Η αποκοπή σημείου είναι πολύ εύκολη
 - Αν το σημείο ανήκει στο $[000000]$, τότε είναι ορατό

Αποκοπή ευθείας γραμμής

- Για αποκοπή ευθείας γραμμής, πρώτα ορίζουμε τα σημεία αρχής και τέλους μιας γραμμής με τον κωδικό περιοχής τους
- Μπορούμε εύκολα να πούμε πως όλες οι γραμμές με τα δύο τελικά τους σημεία στο [000000] δεν αποκόπτονται
- Ομοίως με την περίπτωση των 2Δ, αποκόπτουμε όλες τις γραμμές των οποίων τα τελικά σημεία τους μοιράζονται κοινά bit «1» σε οποιαδήποτε θέση
 - Στο παράδειγμα που ακολουθεί, η γραμμή από το $P_3[010101]$ στο $P_4[100110]$ θα αποκοπεί

Αποκοπή ευθείας γραμμής



Η εξίσωση γραμμής για 3D αποκοπή

- Για αποκοπή γραμμών σε 3D χρειαζόμαστε τις παραμετρικές εξισώσεις.
- Για ένα τμήμα γραμμής με τελικά σημεία $P_1(x_{1,h}, y_{1,h}, z_{1,h}, h_{1,})$ και $P_2(x_{2,h}, y_{2,h}, z_{2,h}, h_{2,})$ η παραμετρική εξίσωση που περιγράφει οποιοδήποτε σημείο στη γραμμή είναι:

$$P = P_1 + (P_2 - P_1)u \quad 0 \leq u \leq 1$$

Η εξίσωση γραμμής για 3D αποκοπή

- Από αυτή την παραμετρική εξίσωση, μπορούμε να δημιουργήσουμε τις εξισώσεις για ομοιογενείς συντεταγμένες:

$$x_h = x_{1,h} + (x_{2,h} - x_{1,h})u$$

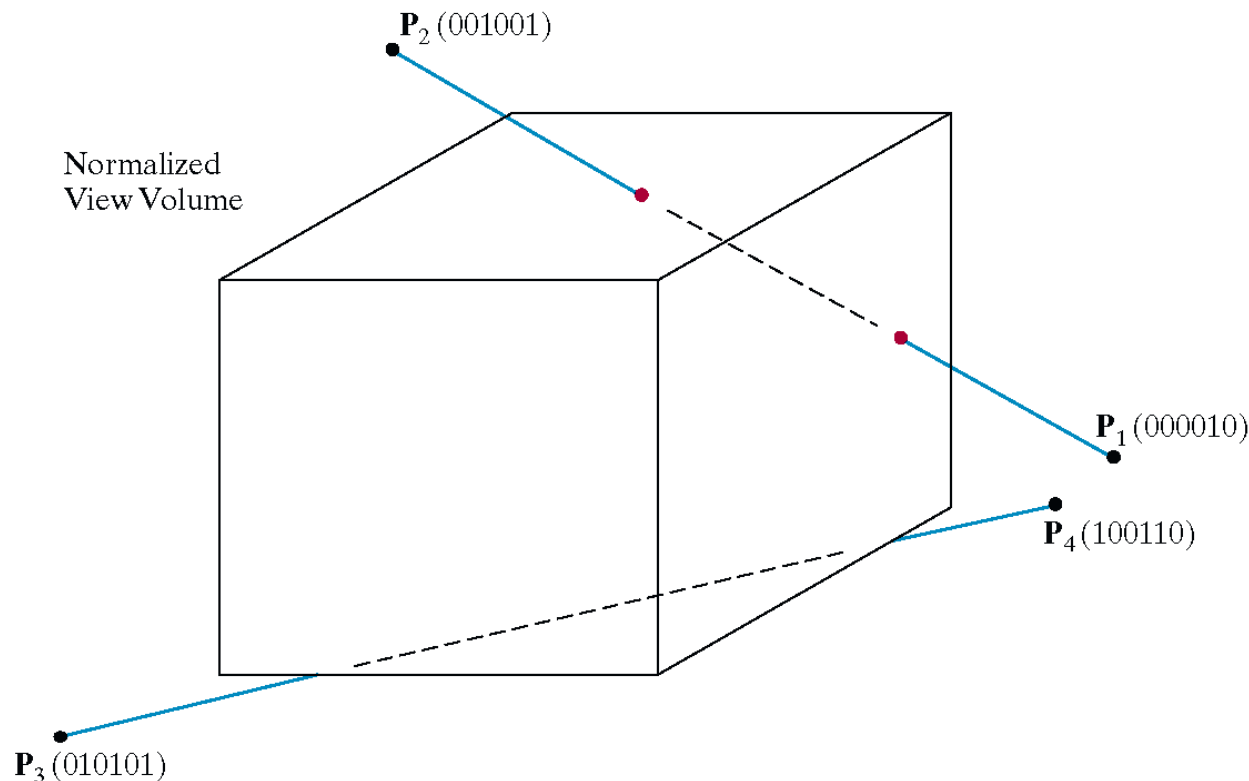
$$y_h = y_{1,h} + (y_{2,h} - y_{1,h})u$$

$$z_h = z_{1,h} + (z_{2,h} - z_{1,h})u$$

$$h = h_1 + (h_2 - h_1)u$$

Παράδειγμα 3Δ αποκοπής

- Έστω η ευθεία από το $P_1[000010]$ στο $P_2[001001]$
- Επειδή τα σημεία έχουν διαφορετικές τιμές στο 2^ο bit γνωρίζουμε ότι η γραμμή διασχίζει το δεξιό όριο



Παράδειγμα 3Δ αποκοπής

- Δεδομένου ότι το δεξιό όριο είναι το $X = 1$, ισχύει το εξής:

$$x_p = \frac{x_h}{h} = \frac{x_{1,h} + (x_{2,h} - x_{1,h})u}{h_1 + (h_2 - h_1)u} = 1$$

- το οποίο λύνουμε ως προς το u :

$$u = \frac{x_{1,h} - h_1}{(x_{1,h} - h_1) - (x_{2,h} - h_2)}$$

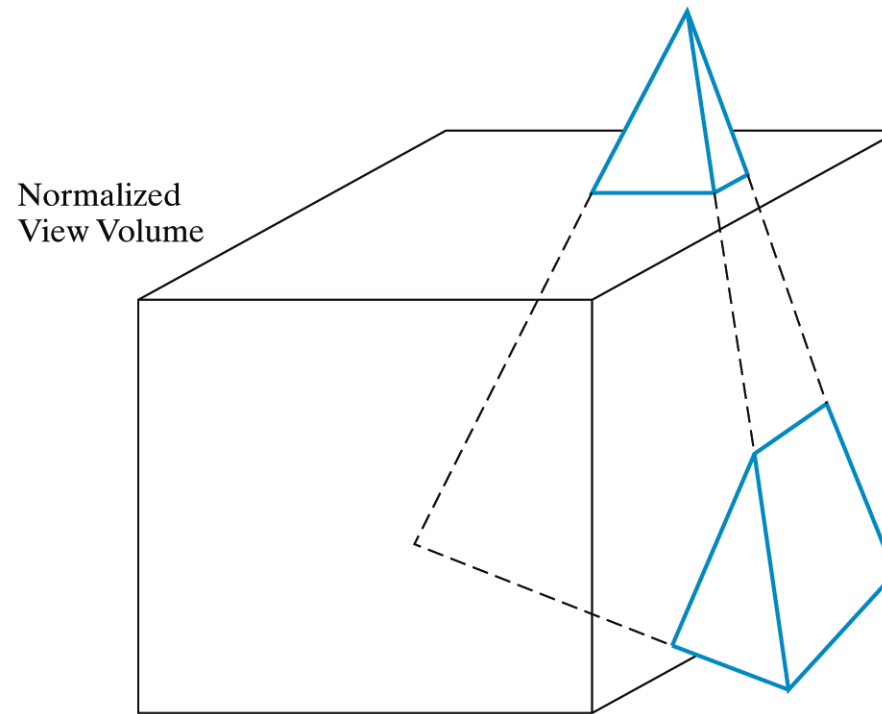
- Χρησιμοποιώντας αυτή την τιμή για το u μπορούμε να βρούμε τα y_p και z_p

Παράδειγμα 3D αποκοπής

- Συνεχίσουμε την ίδια τακτική για κάθε σημείο, και κάθε γραμμή.

3Δ αποκοπή πολυγώνων

- Ωστόσο, η πιο συνηθισμένη περίπτωση 3Δ αποκοπής είναι αποκοπή αντικειμένων που αποτελείται από πολύγωνα



3Δ αποκοπή πολυγώνων

- Σε αυτή την περίπτωση, πρώτα αποκόπτουμε τα πολύγωνα που είναι εξ ολοκλήρου εκτός του οπτικού πεδίου
- Στη συνέχεια εφαρμόζουμε τον αλγόριθμο του Sutherland-Hodgman για κάθε 2Δ πολύγωνο, όπως μάθαμε νωρίτερα

Αποκοπή: Παιχνίδια και επίπεδα

- ▶ **Πρόβλημα:** κάποτε στα παιχνίδια μπορεί να τοποθετήσεις την κάμερα σε ένα σημείο όπου το αντικείμενο πλησίον της κάμερας πρέπει να αποκοπεί για να δεις μέσα από αυτό.
- ▶ Τα παιχνίδια χρησιμοποιούν διάφορες τεχνικές για να αποφύγουν αυτό το πρόβλημα. Μια τεχνική είναι τα αντικείμενα που είναι πολύ κοντά να ξεθωριάζουν πριν αποκοπούν, όπως μπορείτε να δει πιο κάτω



Screenshots from the game, *Okami*

Αποκοπή: Παιχνίδια και επίπεδα

- ▶ **Πρόβλημα:** έχετε ποτέ παίξει ένα βίντεο παιχνίδι και ξαφνικά κάποιο αντικείμενο ξεπροβάλλει στο παρασκήνιο (π.χ., ένα δέντρο σε ένα παιχνίδι αγώνων); Αυτό είναι αντικείμενο που έρχεται μέσα από την αποκοπή far clip.
- ▶ Μια τεχνική είναι να προστεθεί **ομίχλη** σε αντικείμενα που βρίσκονται σε απόσταση. Ένα κλασικό παράδειγμα αυτού του εφέ είναι από το παιχνίδι *Turok: Dinosaur Hunter*
- ▶ Η σύγχρονη λύση είναι η δημιουργία ενός δυναμικού επιπέδου λεπτομέρειας, οι λεπτομέρειες αυξάνονται καθώς πλησιάζουμε σε αυτό το αντικείμενο.



- ▶ Σε αυτό το παράδειγμα το μόνο που παρατηρείται είναι ομίχλη.
- ▶ Τώρα τα πράγματα έχουν βελτιωθεί!