

Melihat & Memotong Dalam 2D

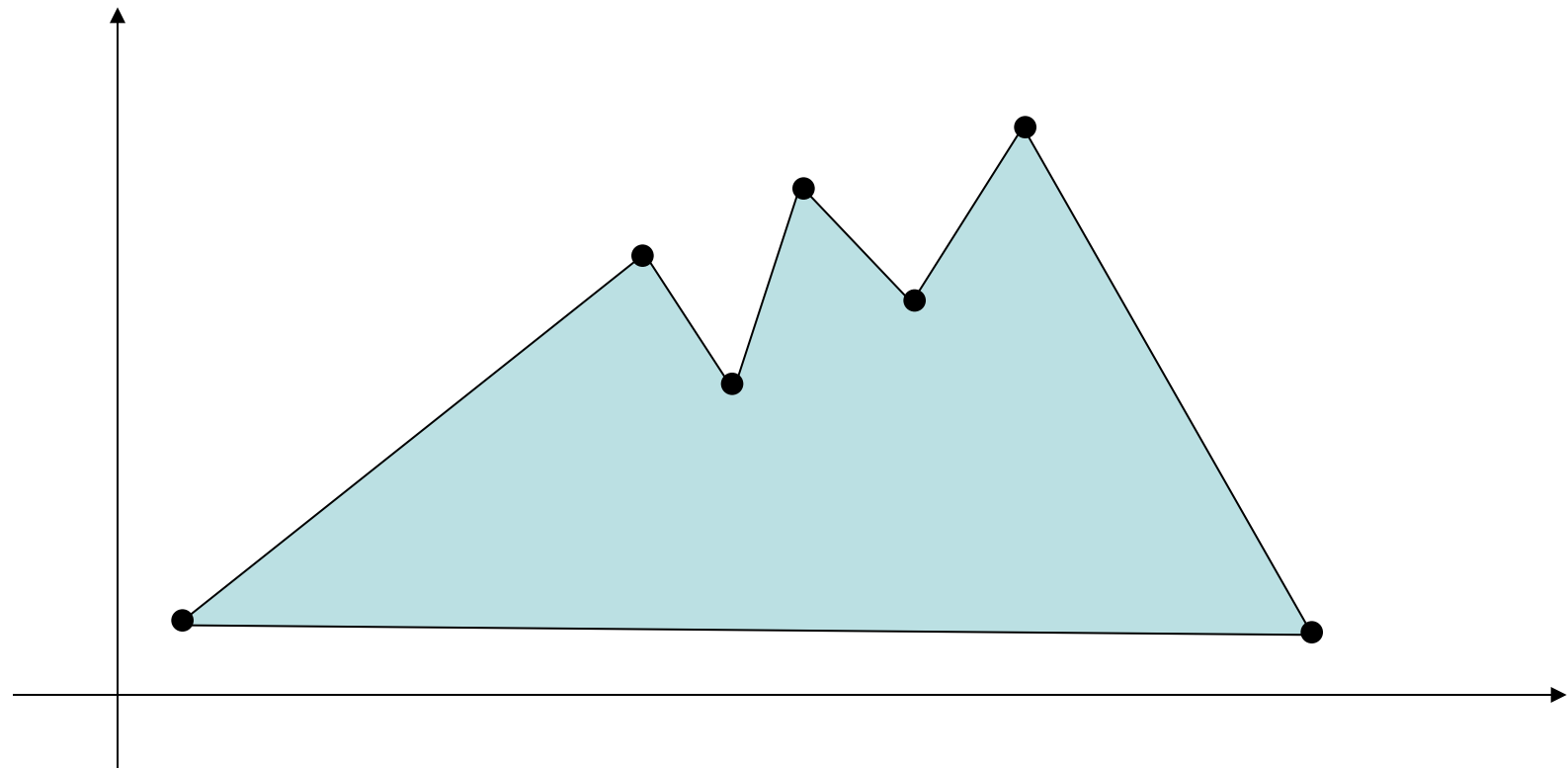
Konsep Jendela Kliping

- pengantar
- Kasar
- Kliping Algoritma Area Kliping

Cohen-Sutherland

- Algoritma Kliping Area Sutherland-Hodgman

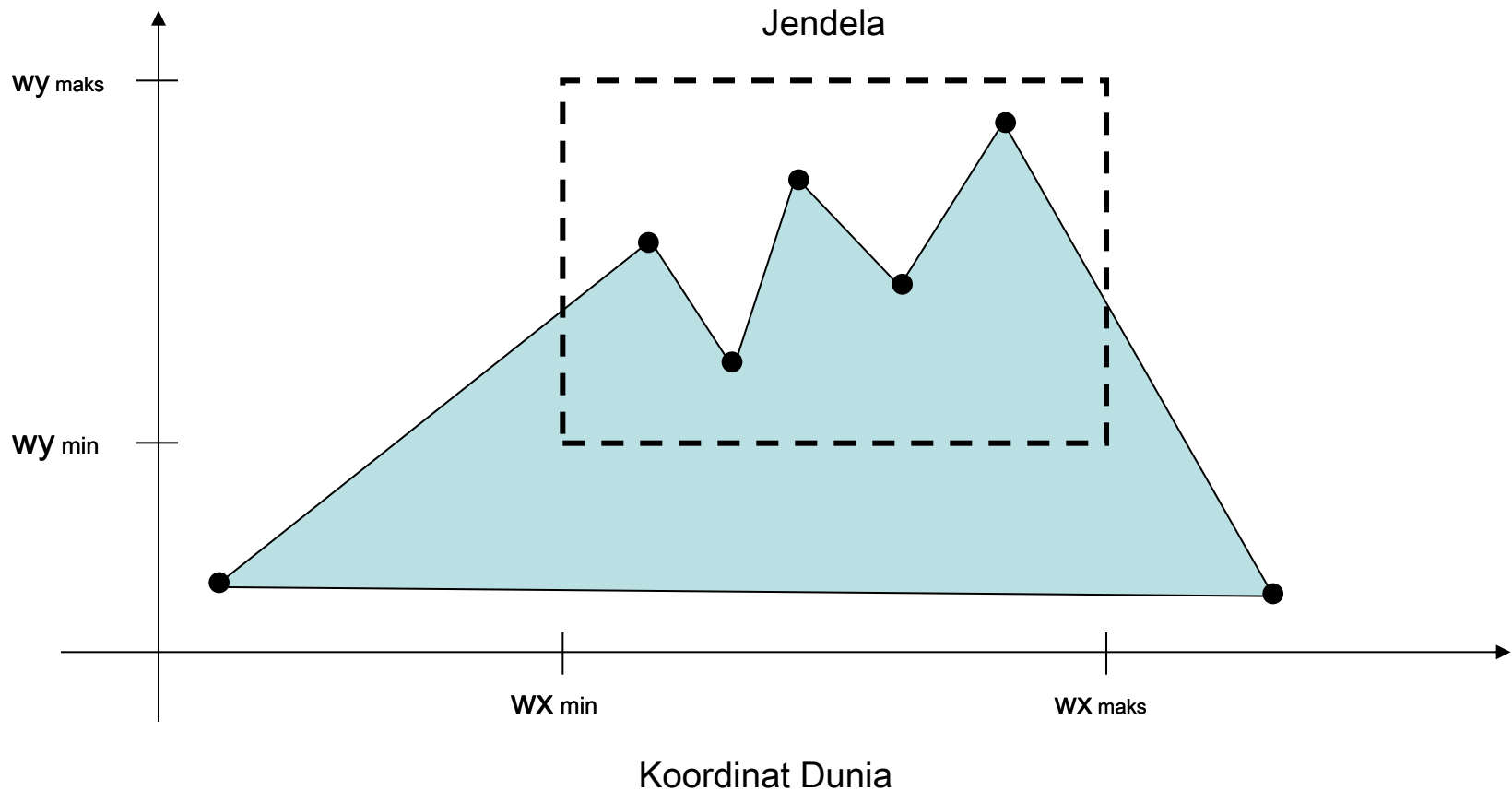
Sebuah adegan terdiri dari kumpulan objek yang ditentukan dalam koordinat dunia



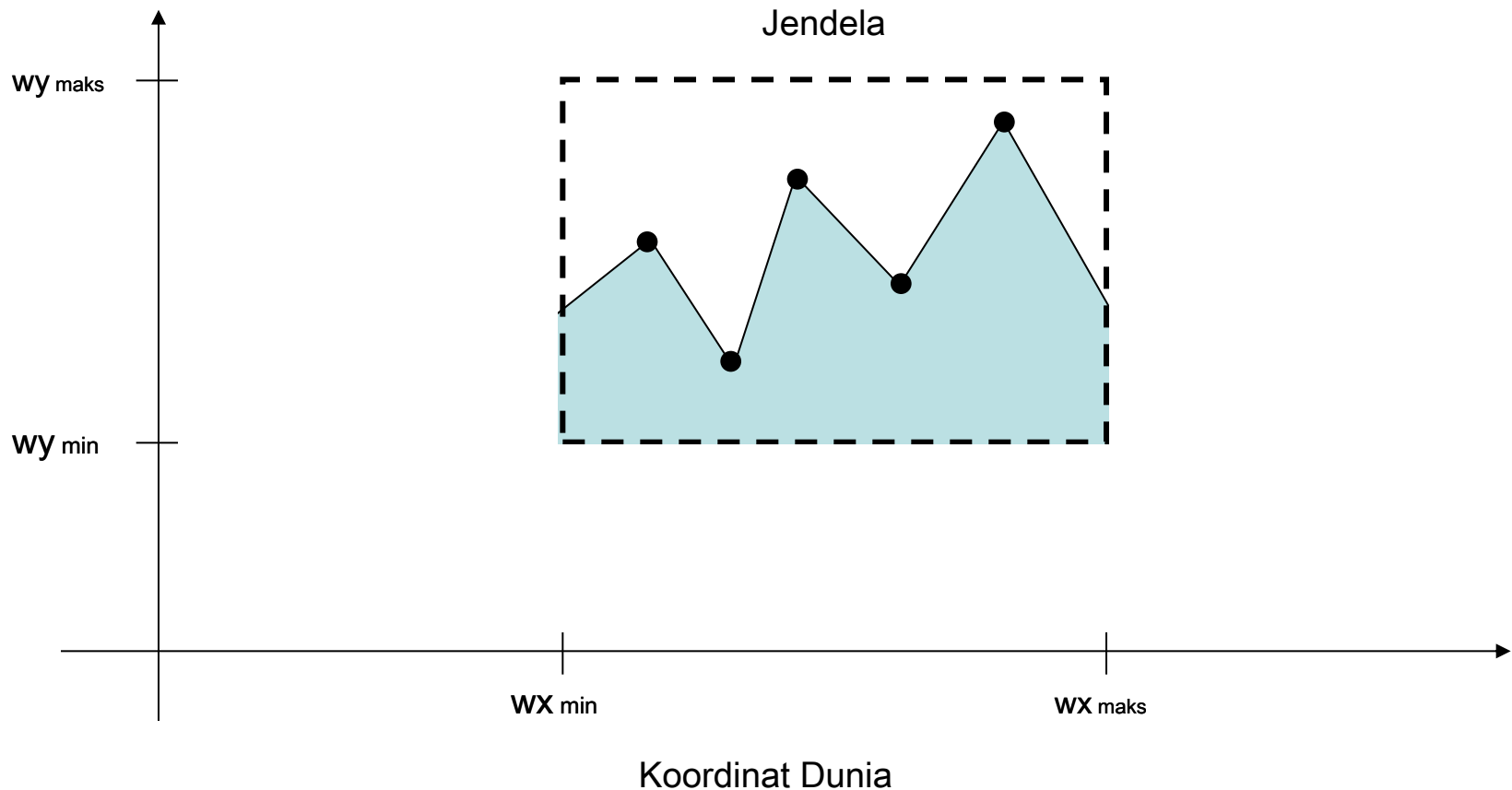
Koordinat Dunia

Windowing II

Ketika kami menampilkan adegan, hanya objek-objek tersebut di dalam jendela tertentu yang ditampilkan



Karena menggambar sesuatu untuk dipajang membutuhkan waktu kita *klip* semuanya di luar jendela



Jelaskan istilah "Window" dan "Viewport". Tulis bentuk umum jendela untuk melihat persamaan transformasi koordinat.

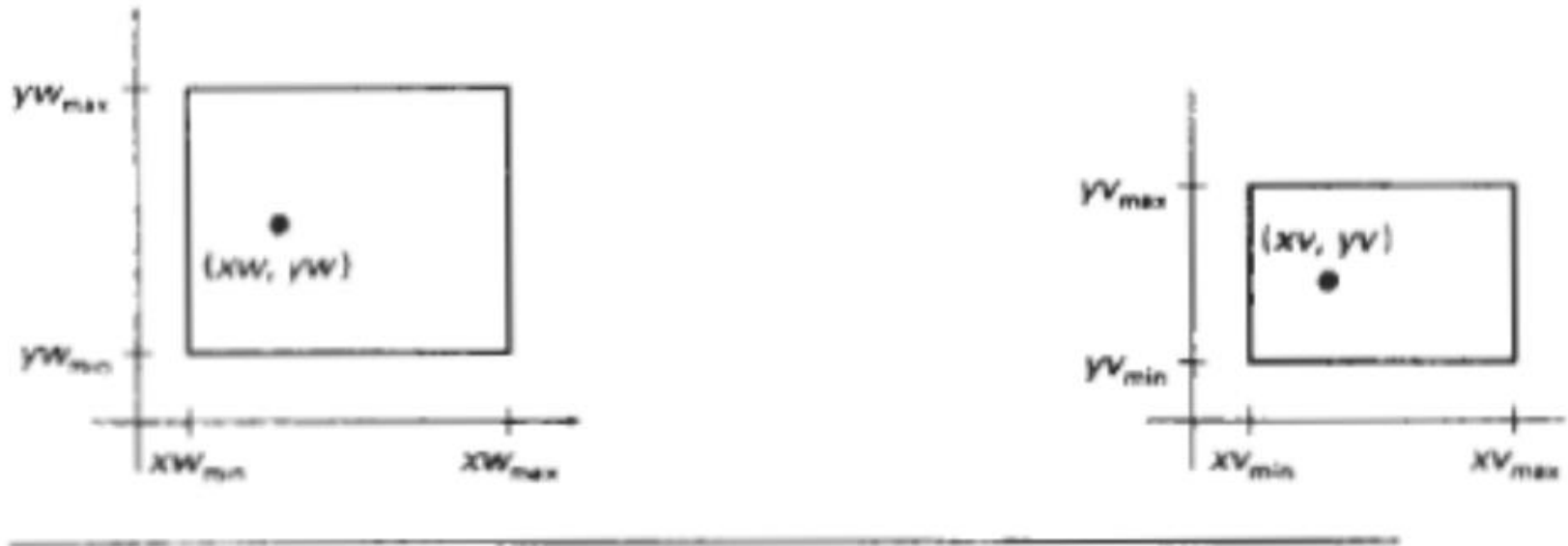
Area koordinat dunia yang dipilih untuk tampilan disebut jendela.

Area pada perangkat tampilan yang dipetakan jendela disebut viewport.

Jendela menentukan apa yang harus dilihat, viewport menentukan di mana ia akan ditampilkan.

Dalam terminologi grafis komputer, jendela istilah awalnya merujuk ke area gambar yang dipilih untuk dilihat.

TRANSFORMASI KOORDINASI WINDOW-TO-VIEWPORT



Gambar menggambarkan pemetaan window-to-viewport.

SEBUAH titik pada posisi (xw , yw) di jendela dipetakan ke posisinya (xv , yv) di viewport terkait. Untuk mempertahankan penempatan relatif yang sama di viewport seperti di jendela, kami mengharuskan itu

$$\frac{xv - xv_{\min}}{xv_{\max} - xv_{\min}} = \frac{yw - yw_{\min}}{yw_{\max} - yw_{\min}}$$

$$\frac{yv - yv_{\min}}{yv_{\max} - yv_{\min}} = \frac{yw - yw_{\min}}{yw_{\max} - yw_{\min}}$$

Solving these expressions for the viewport position (xv, yv) , we have

$$xv = xv_{min} + (xw - xw_{min})sx$$

$$yv = yv_{min} + (yw - yw_{min})sy$$

where the scaling factors are

$$sx = \frac{xv_{max} - xv_{min}}{xw_{max} - xw_{min}}$$

$$sy = \frac{yv_{max} - yv_{min}}{yw_{max} - yw_{min}}$$

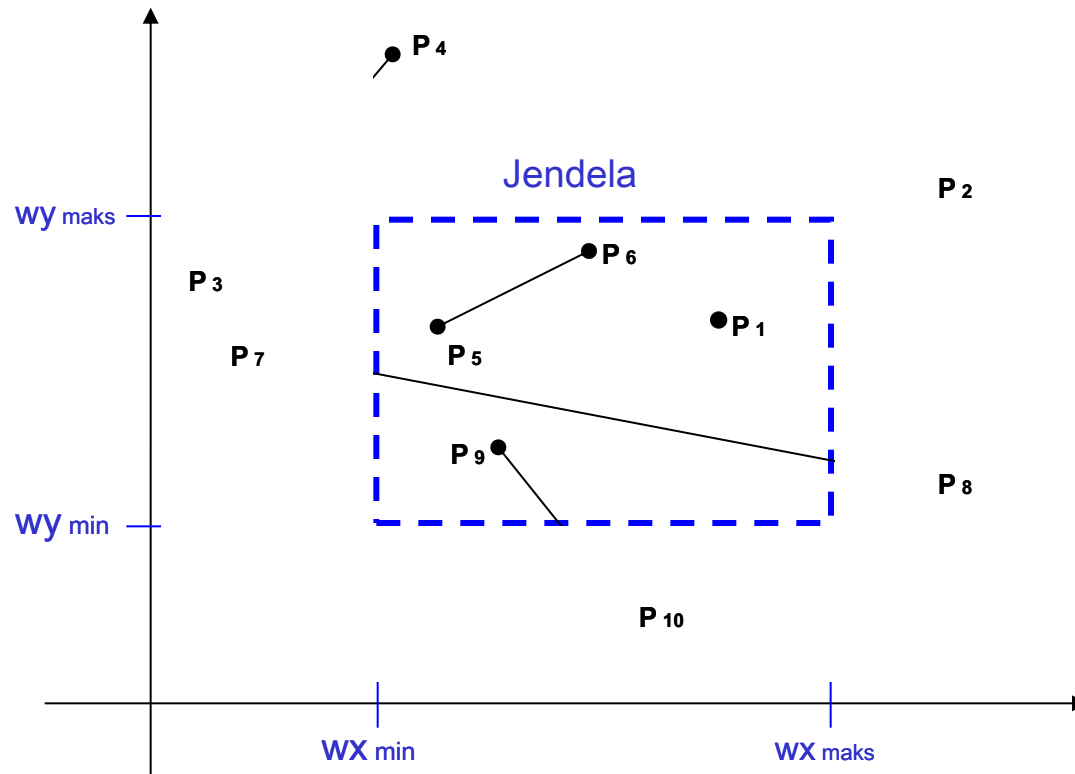
Konversi dari area jendela ke area viewport dilakukan dengan urutan transformasi berikut.

1. Lakukan transformasi penskalaan menggunakan posisi titik tetap ($x_{w \min}$, $y_{w \min}$) yang mengukur area jendela dengan ukuran viewport.
2. Terjemahkan area jendela yang diskalakan ke posisi viewport.

Setiap prosedur yang mengidentifikasi bagian-bagian gambar yang berada di dalam atau di luar wilayah tertentu dari ruang disebut sebagai algoritma kliping atau hanya kliping.

Wilayah tempat objek dijepit disebut jendela klip.

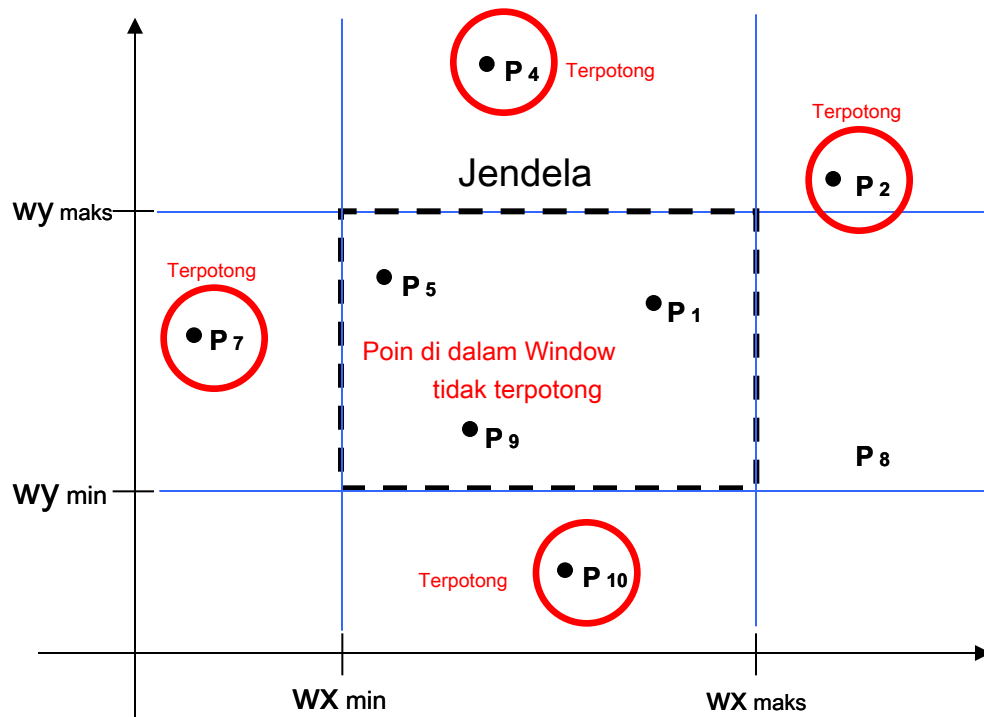
Untuk gambar di bawah ini pertimbangkan garis dan titik mana yang harus dijaga dan mana yang harus dipotong



Mudah - suatu titik (x, y) tidak terpotong jika:

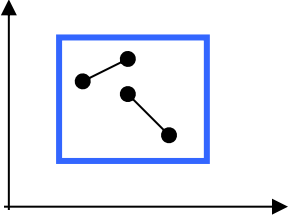
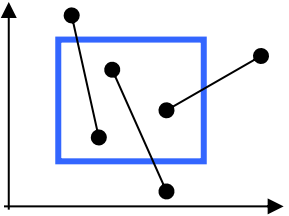
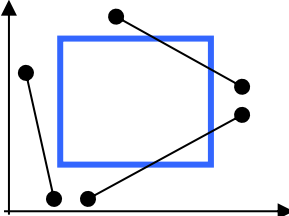
$$WX_{min} \leq x \leq WX_{maks} \text{ DAN } wy_{min} \leq y \leq wy_{maks}$$

jika tidak maka akan terpotong



Klipping Baris

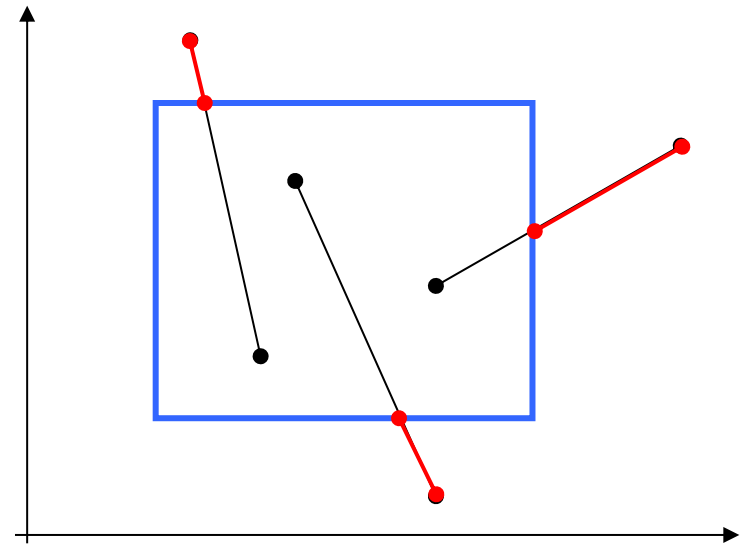
Sulit - memeriksa titik akhir dari setiap baris untuk melihat apakah mereka ada di jendela atau tidak

| Situasi | Larutan | Contoh |
|---|-------------|---|
| Kedua titik akhir di dalam jendela | Jangan klip |  |
| Satu titik akhir di dalam jendela, satu di luar | Harus klip |  |
| Kedua titik akhir di luar jendela | Tidak tahu! |  |

Kliping Garis Brute Force

Kliping garis kasar dapat dilakukan sebagai berikut:

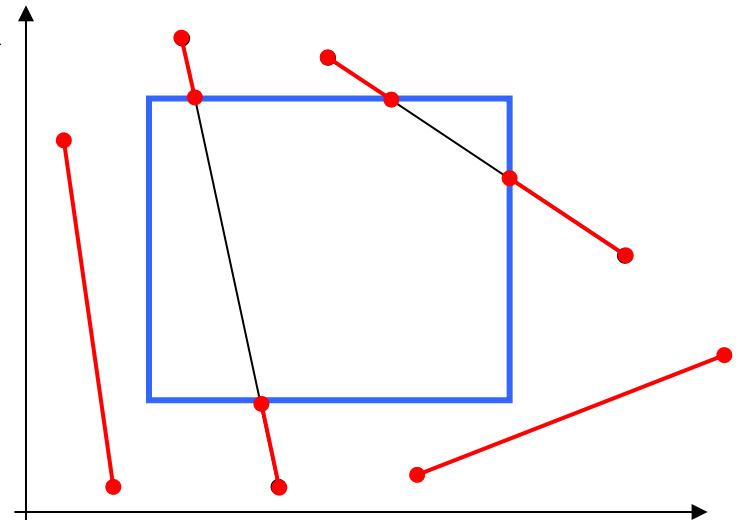
- Jangan memotong garis dengan kedua titik akhir di dalam jendela
- Untuk garis dengan satu titik akhir di dalam jendela dan satu titik akhir di luar, hitung



titik persimpangan (menggunakan persamaan garis)
dan klip dari titik ini

Klipping Brute Force Line (lanjutan ...)

- Untuk garis dengan kedua titik akhir di luar jendela uji garis untuk persimpangan dengan semua batas jendela, dan klip dengan tepat



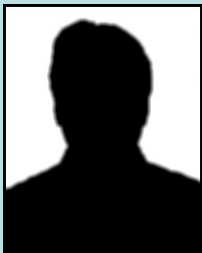
Namun, menghitung persimpangan garis secara komputasi mahal

Karena sebuah adegan dapat mengandung begitu banyak garis, pendekatan brute force untuk memotong terlalu lambat

Algoritma Kliping Cohen-Sutherland

Algoritme kliping garis yang efisien

Keuntungan utama dari algoritma ini adalah ia sangat mengurangi jumlah persimpangan garis yang harus dihitung



Cohen adalah suatu misteri - adakah yang bisa mengetahui siapa dia?



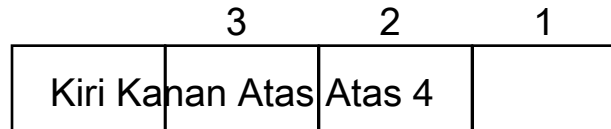
Ivan E. Sutherland bersama-sama mengembangkan algoritma Cohen-Sutherland dan clipping. Sutherland adalah raksasa grafis dan termasuk di antara prestasinya pada penemuan kepala mount diputar.

Algoritma Kliping Cohen-Sutherland

- Salah satu algoritma paling awal dengan banyak variasi dalam penggunaan.
- Waktu pemrosesan dikurangi dengan melakukan lebih banyak tes sebelum melanjutkan ke perhitungan persimpangan.
- Pada awalnya, setiap titik akhir garis diberi nilai biner empat digit yang disebut kode wilayah, dan setiap bit digunakan untuk menunjukkan apakah titik berada di dalam atau di luar salah satu batas jendela kliping.

Algoritma Kliping Cohen-Sutherland

- Kita dapat mereferensikan tepi jendela dalam urutan apa pun, dan berikut adalah satu kemungkinan.



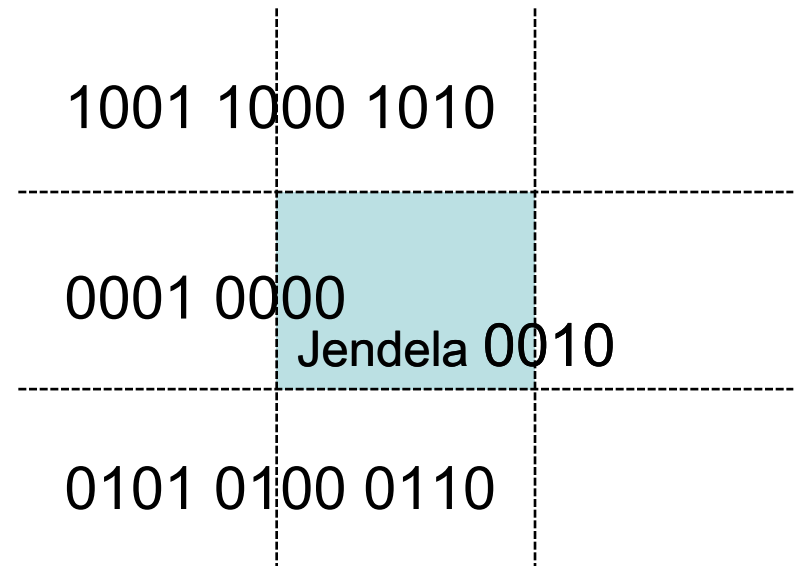
- Untuk pemesanan ini, (bit 1) referensi batas kiri, dan (bit 4) referensi yang teratas.
- Nilai 1 (benar) dalam posisi bit mana pun menunjukkan bahwa titik akhir berada di luar batas itu.
- Nilai 0 (false) menunjukkan bahwa titik akhir berada di dalam atau di perbatasan itu.

Cohen-Sutherland: Divisi Dunia

- Perbatasan empat jendela membuat sembilan wilayah
- Gambar di bawah mencantumkan nilai untuk kode biner di masing-masing wilayah ini.

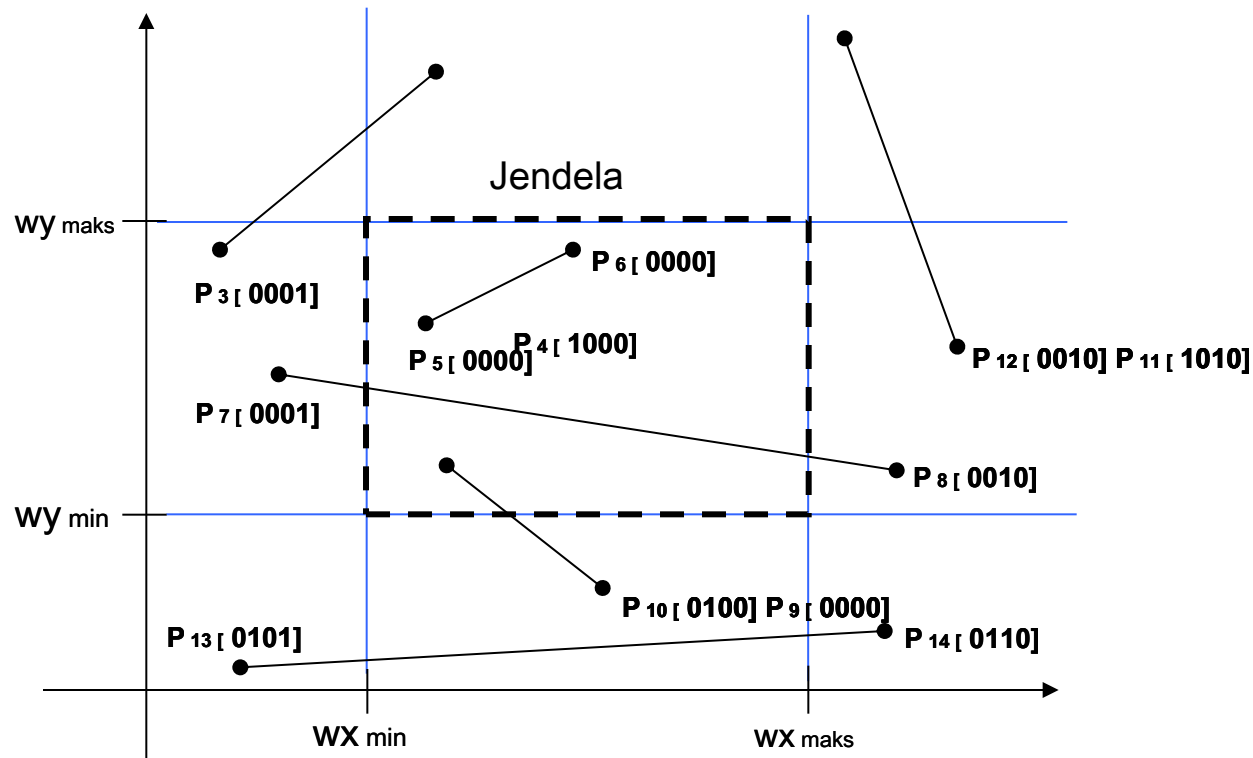
Dengan demikian, titik akhir yang di bawah dan di sebelah kiri jendela kliping diberikan wilayah (0101).

Kode wilayah untuk titik akhir apa pun di dalam jendela kliping adalah (0000).

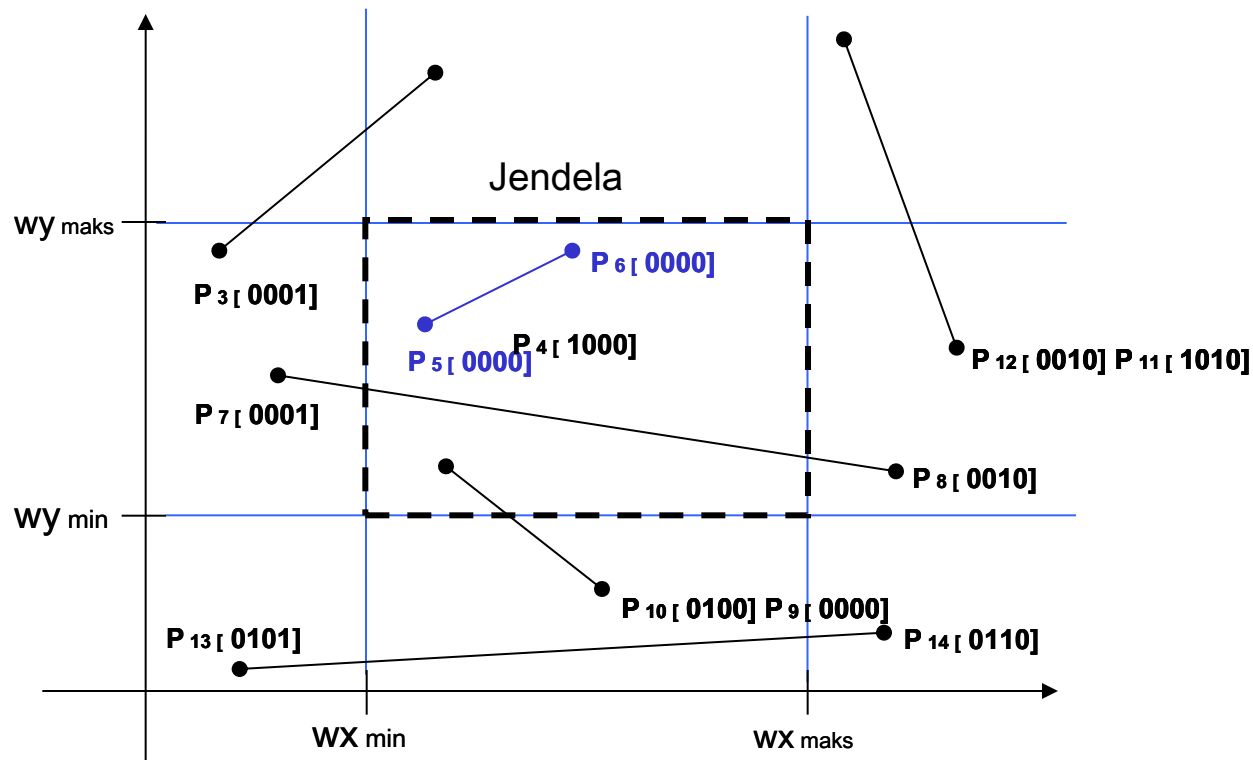


Cohen-Sutherland: Pelabelan

Setiap titik akhir dilabeli dengan kode wilayah yang sesuai

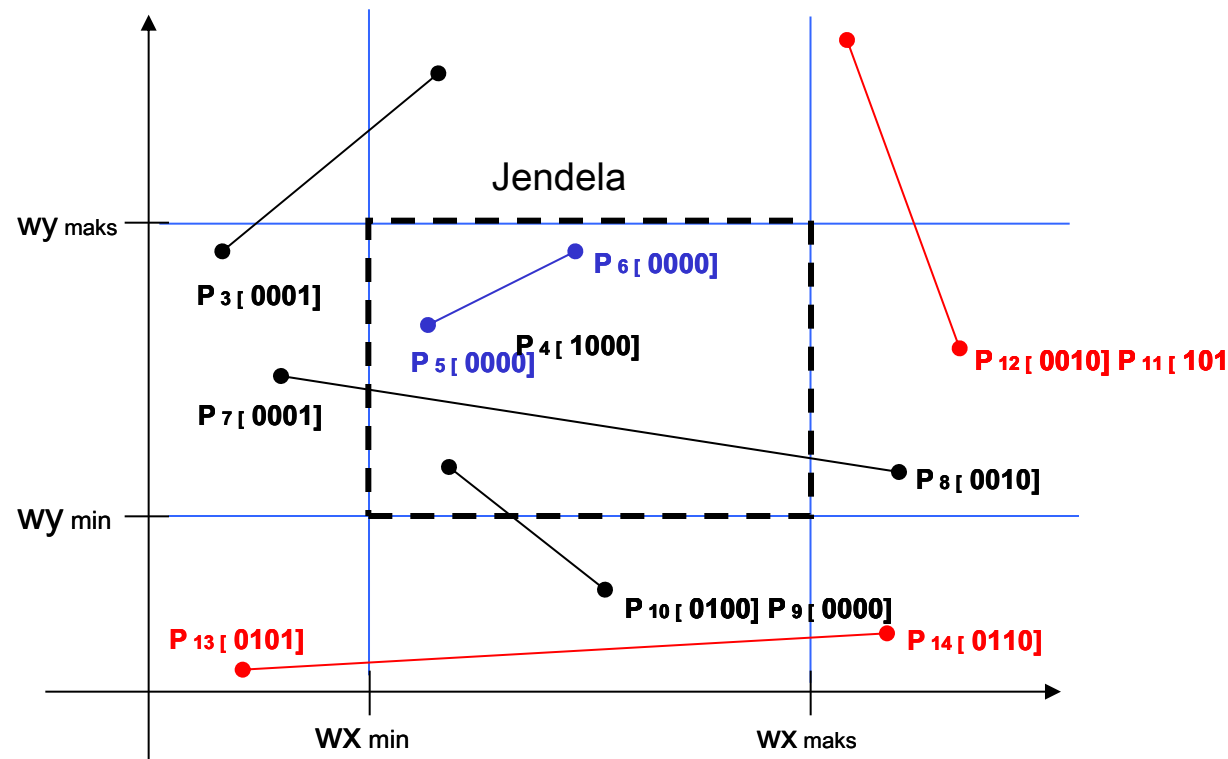


Garis-garis yang sepenuhnya terkandung dalam batas-batas jendela memiliki kode wilayah [0000] untuk kedua titik akhir sehingga tidak terpotong



Setiap garis dengan 1 pada posisi bit yang sama untuk kedua titik akhir benar-benar di luar dan harus dipotong.

Misalnya sebuah baris dengan kode 1010 untuk satu titik akhir dan 0010 untuk yang lain (baris P11, P12) sepenuhnya berada di sebelah kanan jendela kliping.



- Kita dapat melakukan pengujian di dalam / luar untuk saluran menggunakan operator logis.
- Ketika **atau** operasi antara dua kode titik akhir adalah false (0000), garis ada di dalam jendela kliping, dan kami menyimpannya.
- Ketika **dan** operasi antara dua kode titik akhir adalah benar (bukan 0000), garisnya benar-benar di luar jendela kliping, dan kita dapat menghilangkannya.

Cohen-Sutherland: Baris Lain

Garis yang tidak dapat diidentifikasi sepenuhnya di dalam atau di luar jendela mungkin atau mungkin tidak melintasi interior jendela

Baris-baris ini diproses sebagai berikut:

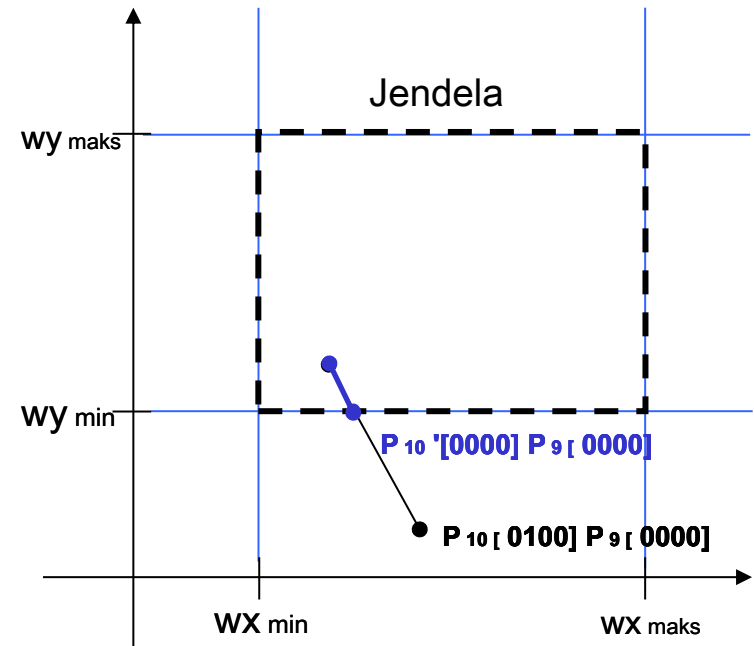
- Bandingkan titik akhir di luar jendela dengan batas (pilih urutan apa pun untuk mempertimbangkan batas misalnya kiri, kanan, bawah, atas) dan tentukan berapa banyak yang dapat dibuang.
- Jika sisa baris seluruhnya di dalam atau di luar jendela, pertahankan atau klipkan masing-masing

- Jika tidak, bandingkan sisa garis dengan batas jendela lainnya
- Lanjutkan sampai garis ditemukan atau segmen di dalam jendela ditemukan. Kita dapat menggunakan kode wilayah untuk menentukan batas jendela mana yang harus dipertimbangkan untuk persimpangan
- Untuk memeriksa apakah suatu garis melewati batas tertentu, kami membandingkan bit yang sesuai dalam kode wilayah titik akhir
- Jika salah satu dari ini adalah 1 dan yang lainnya adalah 0 maka garis melewati batas

Contoh Cohen-Sutherland

Pertimbangkan baris P_9 ke P_{10} di bawah

- Mulai dari P_{10}
- Dari kode wilayah dari dua titik akhir kita tahu garis tidak melewati batas kiri atau kanan
- Hitung persimpangan garis dengan batas bawah untuk menghasilkan titik P_{10}'
- Garis P_9 ke P_{10}' Benar-benar di dalam jendela jadi dipertahankan

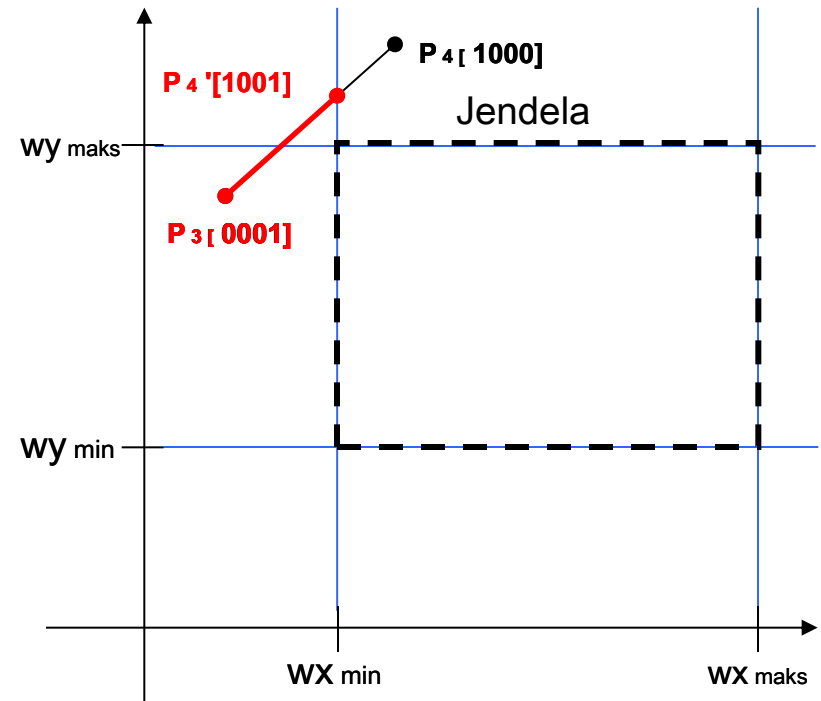


Cohen-Sutherland Examples (cont...)

Pertimbangkan baris P_3 ke P_4 di bawah

- Mulai dari P_4
- Dari kode wilayah dari dua titik akhir kita tahu garis melintasi kiri

batas jadi hitung titik persimpangan untuk menghasilkan P_4'



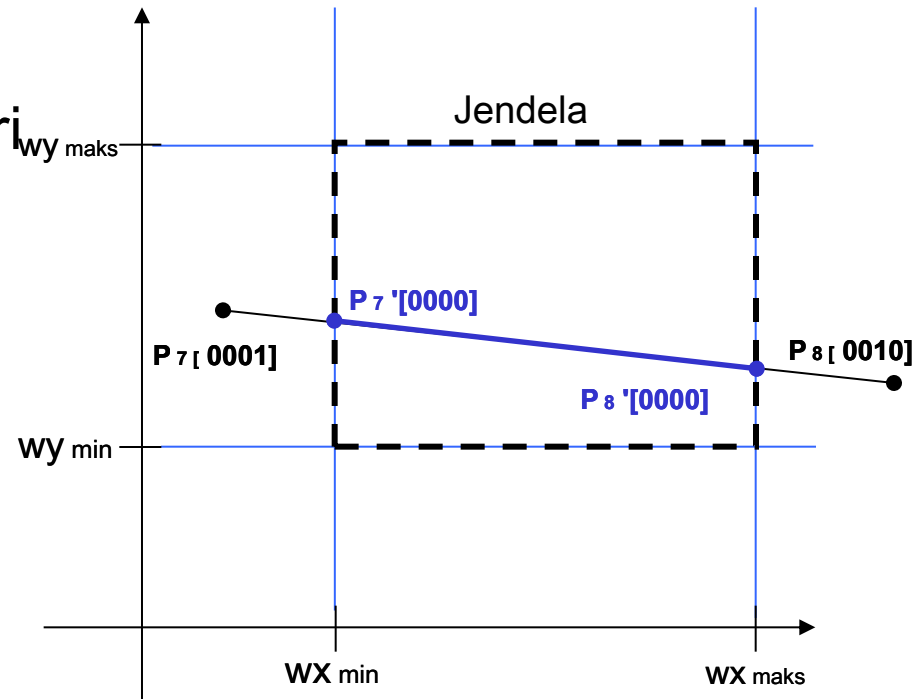
- Garis P_3 ke P_4' Benar-benar di luar jendela jadi terpotong

Cohen-Sutherland Examples (cont...)

Pertimbangkan baris P_7 ke P_8 di bawah

- Mulai dari P_7
- Dari dua kode wilayah dari dua titik akhir kita tahu garis melintasi batas kiri jadi hitung

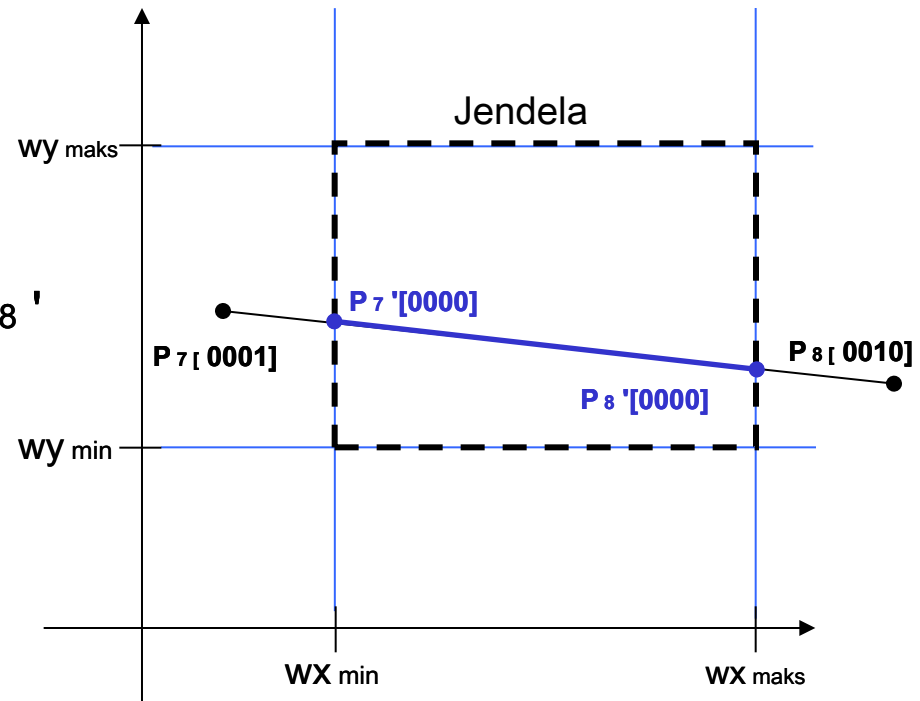
titik persimpangan untuk menghasilkan P_7'



Cohen-Sutherland Examples (cont...)

Pertimbangkan baris P_7 ke P_8

- Mulai dari P_8
- Hitung persimpangan dengan batas kanan untuk menghasilkan P_8'
- P_7 ke P_8' Ada di dalam jendela jadi dipertahankan

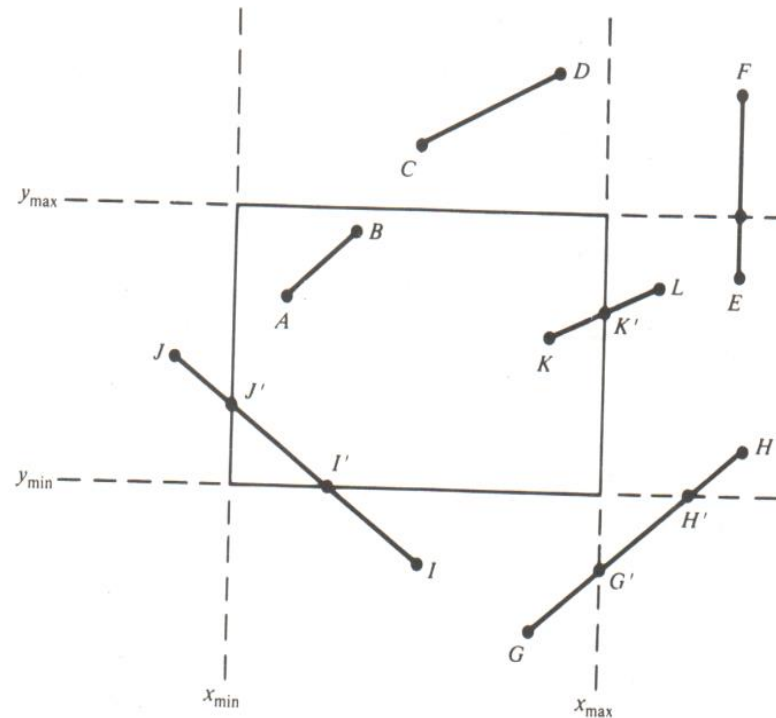


Cohen-Sutherland Examples (cont...)

Q.1: Jelaskan algoritma Sutherland-Cohen yang memberikan semua kemungkinan kasus kliping baris dengan bantuan diagram?

Cohen-Sutherland Examples (cont...)

T.2: Jelaskan algoritma Cohen-Sutherland dan buat daftar kategori kliping untuk segmen garis yang diberikan dalam gambar berikut.



Cohen-Sutherland Examples (cont...)

R.3: Misalkan R adalah jendela segi empat yang sudut kiri bawahnya di L (-3, 1) dan sudut kanan atas berada di R (2, 6).
Temukan kode titik akhir untuk baris berikut menggunakan algoritma Sutherland-Cohen dan tulis kategori kliping untuk baris AB, CD, EF, GH, IJ: A (-4, 3)

B (-1, 9) C

(-2, 5)

D (4, 8) E

(-2, 3)

F (1, 2) G

(-1, -2) H (3, 3) I (-4, 7)

J (-2, 10)

Cohen-Sutherland Examples (cont...)

The endpoint codes for point (x, y) are set according to the scheme

$$\text{Bit 1} = \text{sign}(y - y_{\max}) = \text{sign}(y - 6)$$

$$\text{Bit 3} = \text{sign}(x - x_{\max}) = \text{sign}(x - 2)$$

$$\text{Bit 2} = \text{sign}(y_{\min} - y) = \text{sign}(1 - y)$$

$$\text{Bit 4} = \text{sign}(x_{\min} - x) = \text{sign}(-3 - x)$$

Here

$$\text{Sign } a = \begin{cases} 1 & \text{if } a \text{ is positive or zero} \\ 0 & \text{otherwise} \end{cases}$$

So

$$A(-4, 2) \rightarrow 0001$$

$$F(1, 2) \rightarrow 0000$$

$$B(-1, 7) \rightarrow 1000$$

$$G(1, -2) \rightarrow 0100$$

$$C(-1, 5) \rightarrow 0000$$

$$H(3, 3) \rightarrow 0010$$

$$D(3, 8) \rightarrow 1010$$

$$I(-4, 7) \rightarrow 1001$$

$$E(-2, 3) \rightarrow 0000$$

$$J(-2, 10) \rightarrow 1000$$

Cohen-Sutherland Examples (cont...)

Kategori 1 (terlihat): EF karena kedua kode titik akhir adalah 0000

Kategori 2 (tidak terlihat): IJ sejak $(1001) \text{ AND } (1000) = 1000$ (yang bukan 0000)

Kategori 3 (kandidat untuk kliping): AB, CD dan GH (karena logika DAN titik akhir baris sama dengan 0000)

Menghitung Interseksi Garis

Titik-temu dengan batas-batas jendela dihitung dengan menggunakan parameter persamaan garis

- Pertimbangkan garis dengan titik akhir (x_1, y_1) dan (x_2, y_2)
- Koordinat-y dari persimpangan dengan batas jendela vertikal dapat dihitung menggunakan:

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

dimana x_{batas} dapat diatur ke salah satu wx_{min} atau

wx_{maks}

Calculating Line Intersections (cont...)

- Koordinat x dari persimpangan dengan batas jendela horizontal dapat dihitung menggunakan:

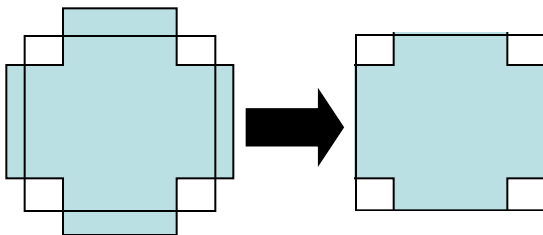
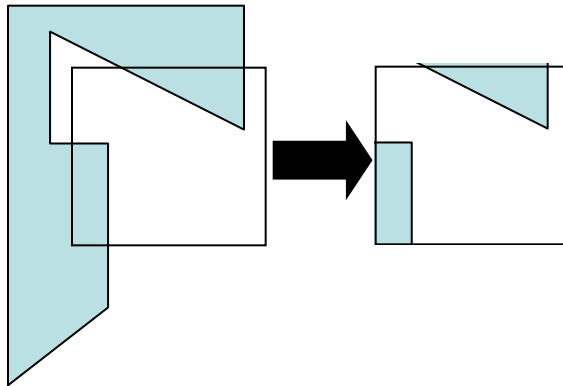
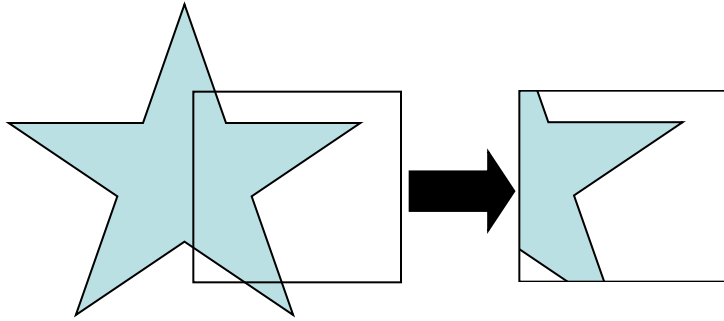
$$x = x_1 + (y_{\text{batas}} - y_1) / m$$

dimana y_{batas} dapat diatur ke salah satu y_{min} atau

y_{maks}

- m adalah kemiringan garis yang dipermasalahkan dan dapat dihitung sebagai $m = (y_2 - y_1) / (x_2 - x_1)$

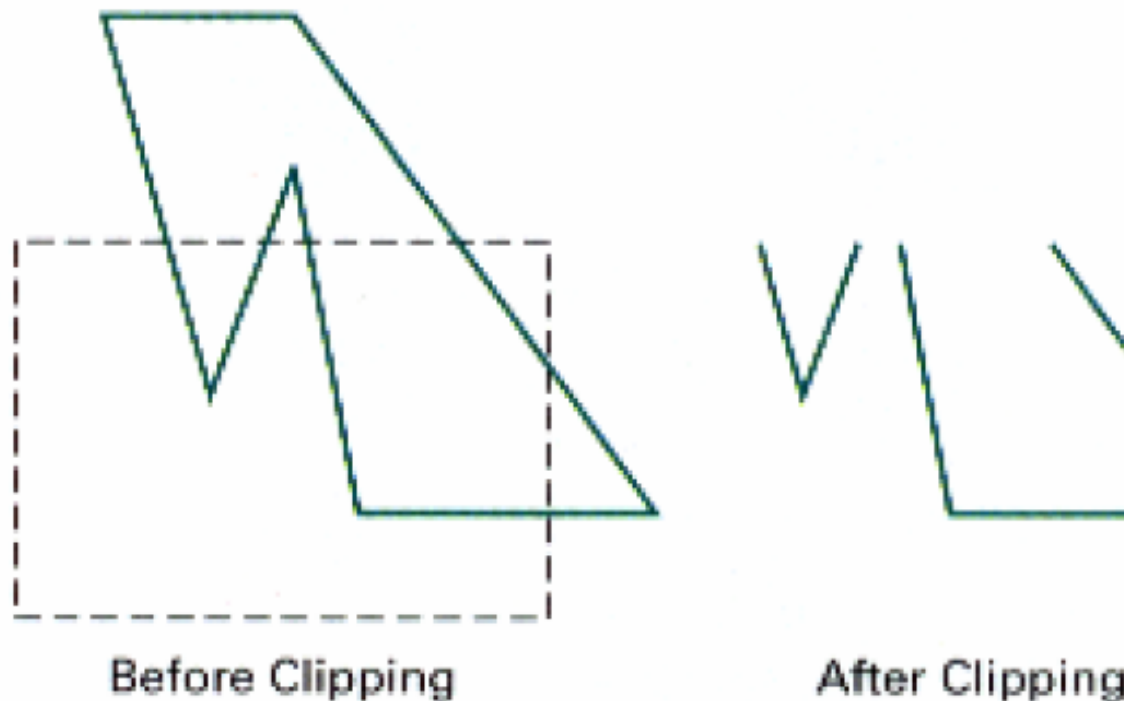
Kliping Area



Demikian pula dengan garis, area harus dijepitkan ke batas jendela
Pertimbangan harus diambil untuk bagian mana dari area yang harus dipotong

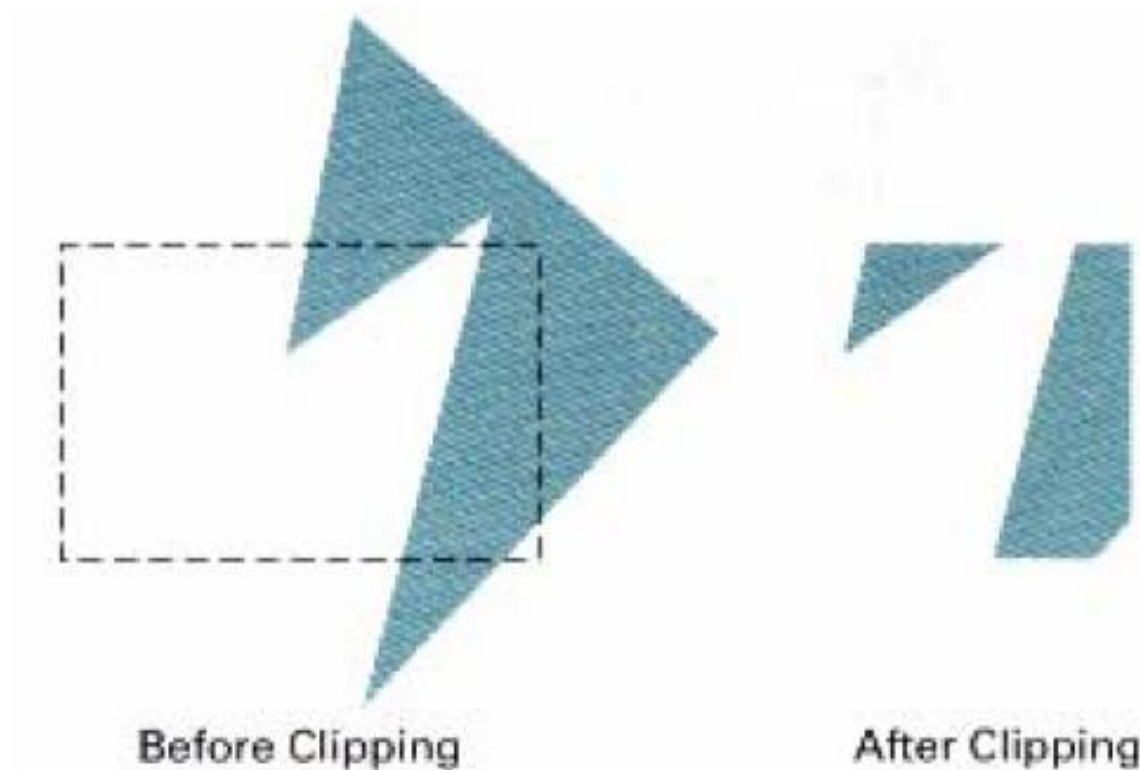
To clip a polygon:

- Modify the line-clipping procedures.
- Polygon boundary processed with a line clipper may be displayed as a series of unconnected line segments as in the following figure:



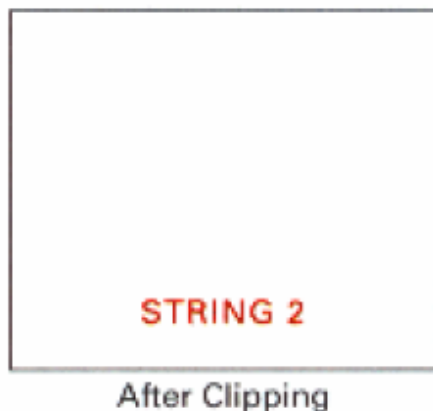
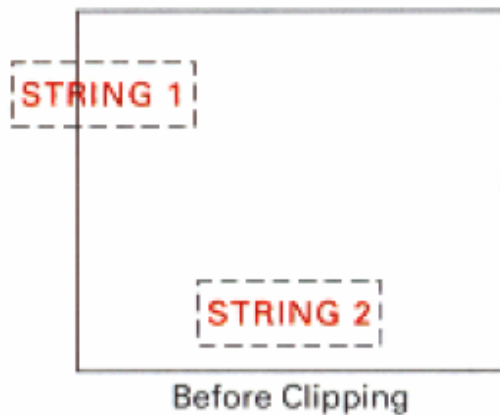
Kliping Area

What really required is a bounded area after clipping as in the following figure:

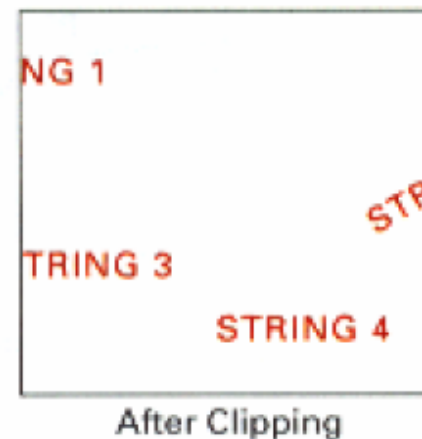
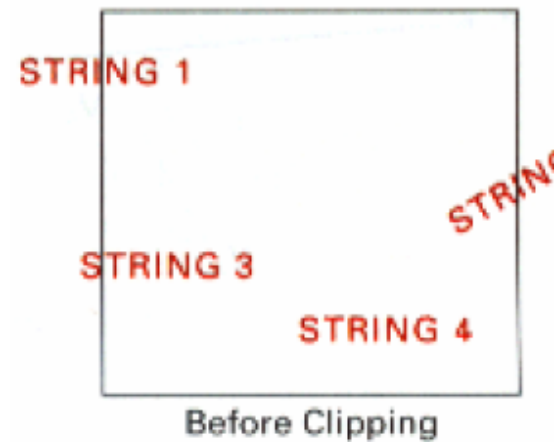


Kliping Teks

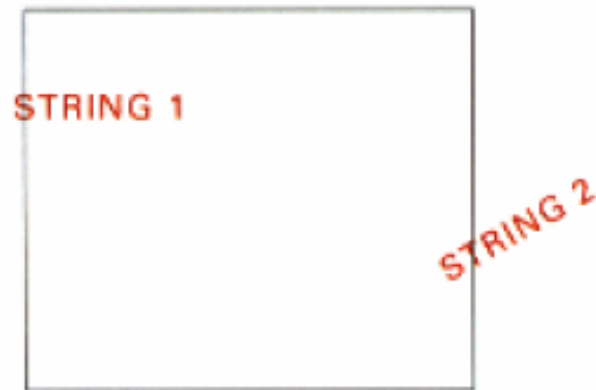
All-or-none String-clipping



All-or-none Character-clipping



Clipping on components of individual Characters



Before Clipping

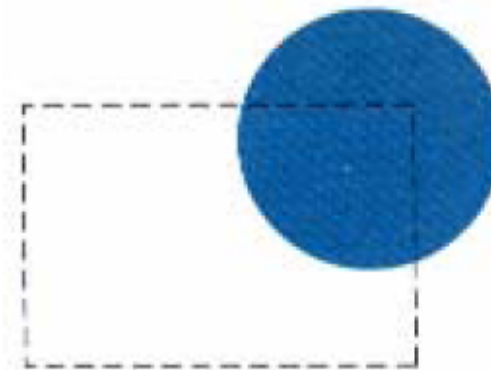


After Clipping

Klipping Kurva

Areas with curved boundaries can be clipped with methods similar to those discussed in the previous sections. Curve-clipping procedures will involve nonlinear equations, however, and this requires more processing than for objects with linear boundaries.

Example



Before Clipping



After Clipping