



PERTEMUAN 06

VIEWING 2D DAN CLIPPING

ADHI PRAHARA

Teknik Informatika. Fakultas Teknologi Industri

Universitas Ahmad Dahlan

CAPAIAN PEMBELAJARAN

Mahasiswa mampu:

- Menjelaskan tentang konsep viewing
- Menjelaskan tentang viewing dengan koordinat referensi
- Menjelaskan tentang viewing 2D
- Menjelaskan tentang clipping 2D dan 3D

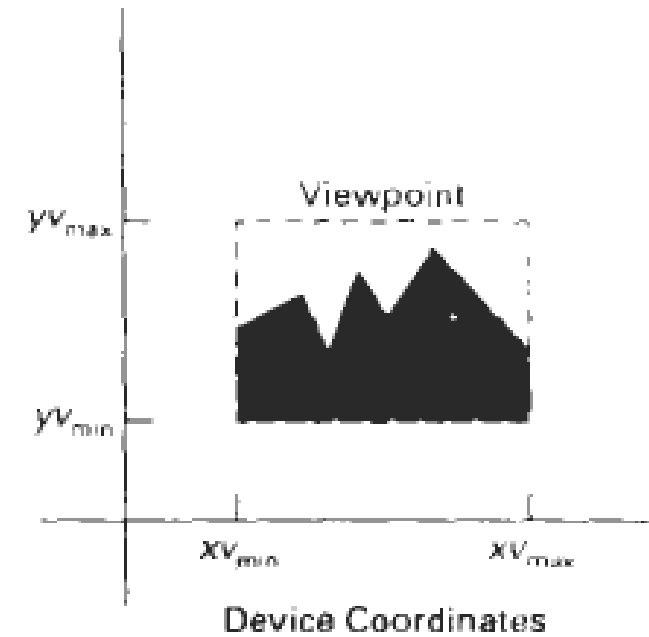
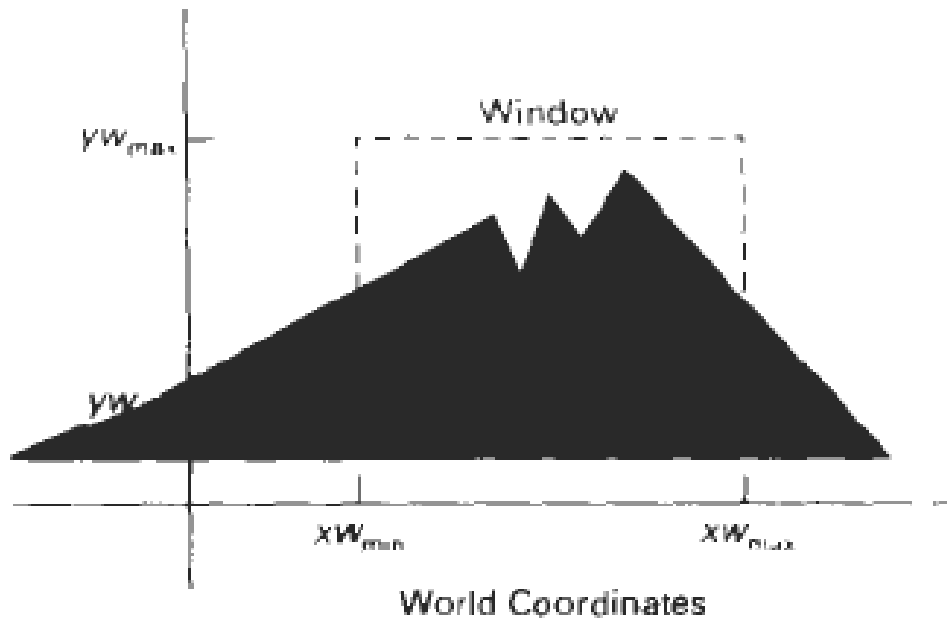
KONSEP VIEWING

- Menentukan bagian mana pada obyek / gambar yang akan ditampilkan di layar
- Dilakukan dengan mentransformasikan world coordinates ke device coordinates
- Konversi world coordinates ke device coordinates melibatkan proses translasi, rotasi, scaling dan clipping

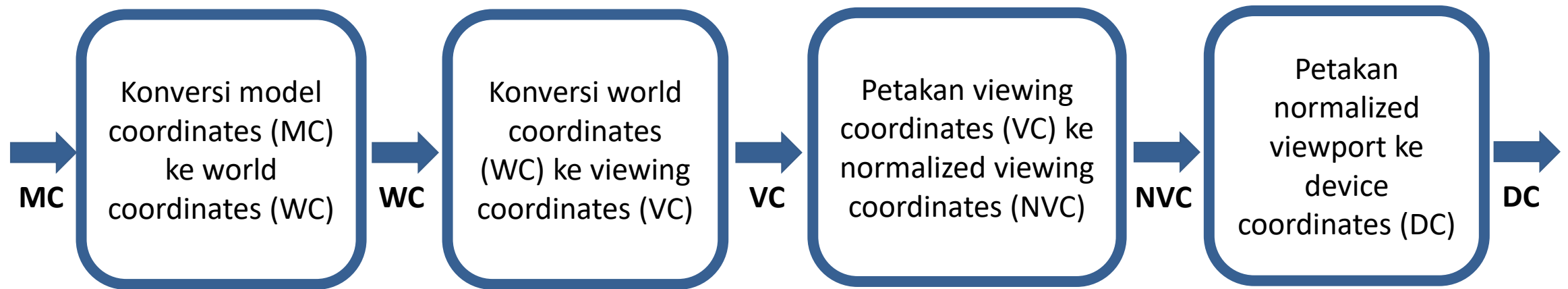
2D VIEWING

- **Window (jendela)** : daerah pada world coordinates yang dipilih sebagai layar tampilan
- **Viewport (daerah pandang)** : daerah pada layar tampilan dimana jendela dipetakan ke daerah tersebut
- Perbedaan :
 - Window mendefinisikan apa yang akan ditampilkan
 - Viewport mendefinisikan dimana akan ditampilkan
- Viewing transformation = window-to-viewport transformation

2D VIEWING



2D VIEWING PIPELINE



2D VIEWING PIPELINE

Langkah 1 :

- Buat ruangan pada world coordinates untuk diletakkan obyek
- Konversi model coordinates ke world coordinates

Langkah 2 :

- Set 2D viewing coordinates pada bidang world coordinates
- Definisikan jendela (window) pada sistem viewing coordinates tersebut
- Viewing coordinates reference frame digunakan untuk mengatur orientasi dari jendela

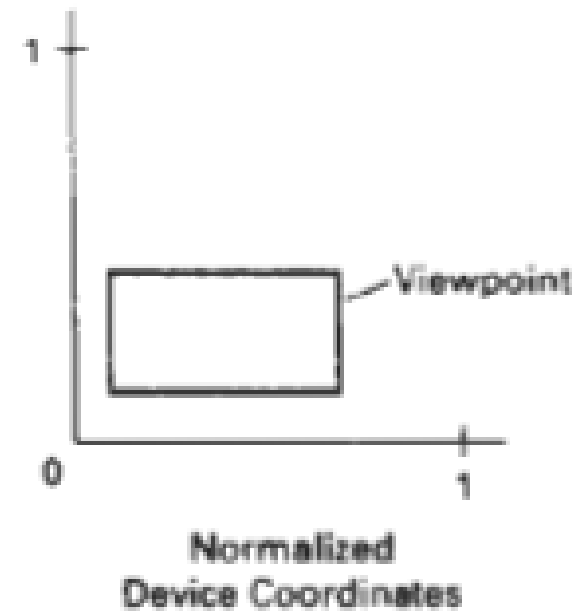
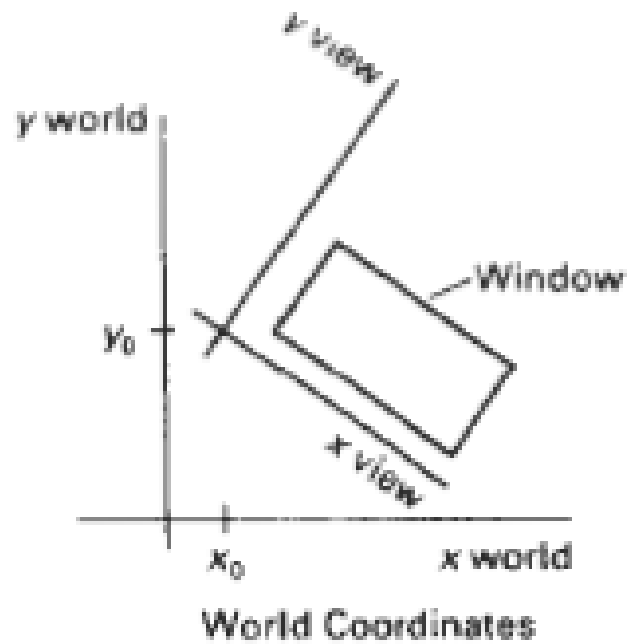
Langkah 3 :

- Definisikan normalized viewing coordinates ($0 - 1$)
- Petakan viewing coordinates ke normalized viewing coordinates

Langkah 4 :

- Lakukan clipping pada bagian obyek diluar daerah pandang
- Isi dari daerah pandang ditransfer ke device coordinates

NORMALIZED VIEWPORT



TUJUAN NORMALIZED VIEWPORT

- Membuat jendela viewing menjadi device independent
- Transformasi pada viewport menjadi device independent
- Mudah dipetakan ke semua device display karena sudah ternormalisasi

TRANSFORMASI ANTAR SISTEM KOORDINAT

- Transformasi dapat dilakukan antara coordinate reference pada bidang kartesius dengan sistem koordinat kartesius yang lain
- Misal satu sistem koordinat mempunyai titik origin $(0,0)$ dan sistem yang lain mempunyai titik origin (x_0, y_0) dengan sudut orientasi θ antara sumbu x dan sumbu x' .
- Langkah transformasi sistem koordinat xy ke sistem koordinat $x'y'$:
 - Melakukan translasi sehingga titik origin (x_0, y_0) dari sistem koordinat $x'y'$ dipindah ke titik asal sistem xy di $(0,0)$
 - Melakukan rotasi sumbu x' ke sumbu x

TRANSFORMASI ANTAR SISTEM KOORDINAT

- Matriks Translasi

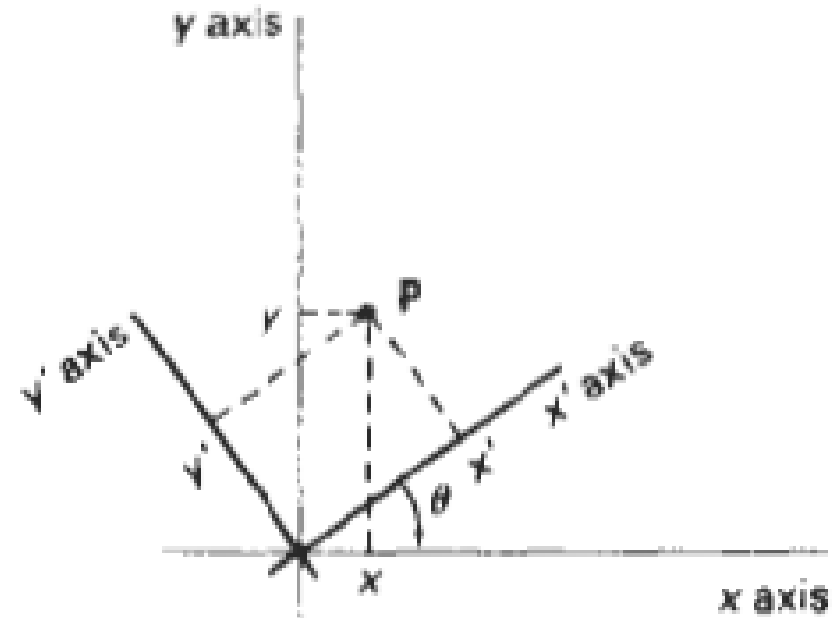
- $T(-x_0, -y_0) = \begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1 \end{bmatrix}$

- Matriks Rotasi

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

- Matriks komposisinya

- $M_{xy,x'y'} = R(-\theta) \cdot T(-x_0, -y_0)$



VIEWING COORDINATE REFERENCE FRAME

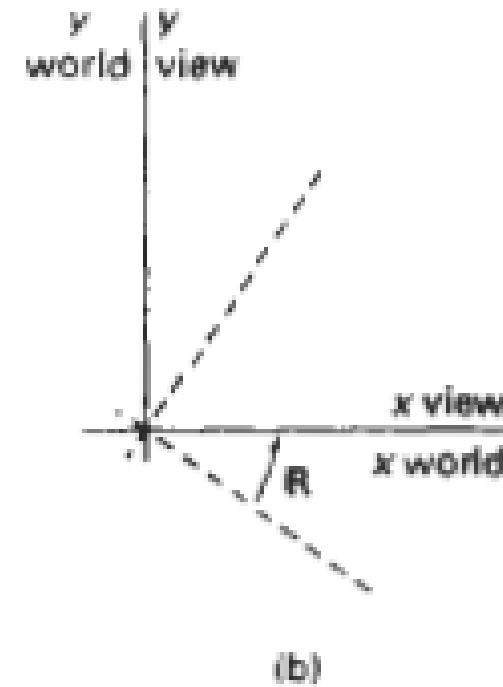
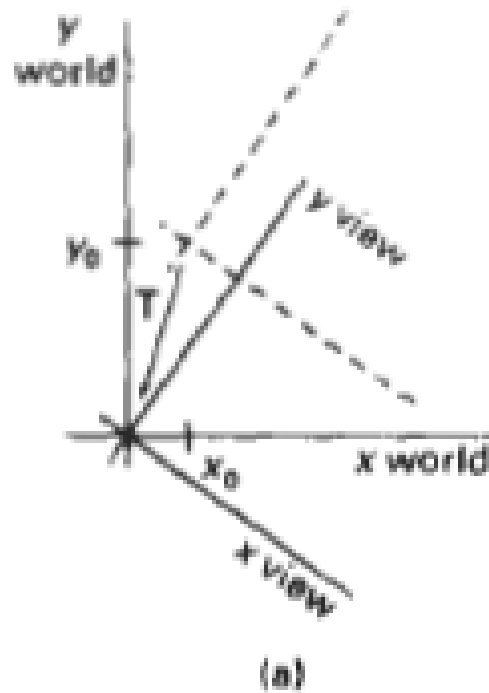
- Sistem koordinat menyediakan reference frame dalam membuat jendela pada world coordinates
- Langkah :
- Titik asal viewing coordinates dipilih pada world coordinates misalnya

$$P_0 = (x_0, y_0)$$

- Mengikuti langkah pada transformasi antar sistem koordinat yaitu ditranslasikan kemudian dirotasikan

$$M_{WC,VC} = R \cdot T$$

VIEWING COORDINATE REFERENCE FRAME



TRANSFORMASI WINDOW-TO-VIEWPORT

- Posisi titik (x_w, y_w) pada window dipetakan ke posisi (x_v, y_v) pada viewport dengan :

$$\frac{x_v - x_{v_{min}}}{x_{v_{max}} - x_{v_{min}}} = \frac{x_w - x_{w_{min}}}{x_{w_{max}} - x_{w_{min}}}$$

$$\frac{y_v - y_{v_{min}}}{y_{v_{max}} - y_{v_{min}}} = \frac{y_w - y_{w_{min}}}{y_{w_{max}} - y_{w_{min}}}$$

TRANSFORMASI WINDOW-TO-VIEWPORT

- Sehingga (xv, yv) dapat dihitung dengan :

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

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

- Dimana

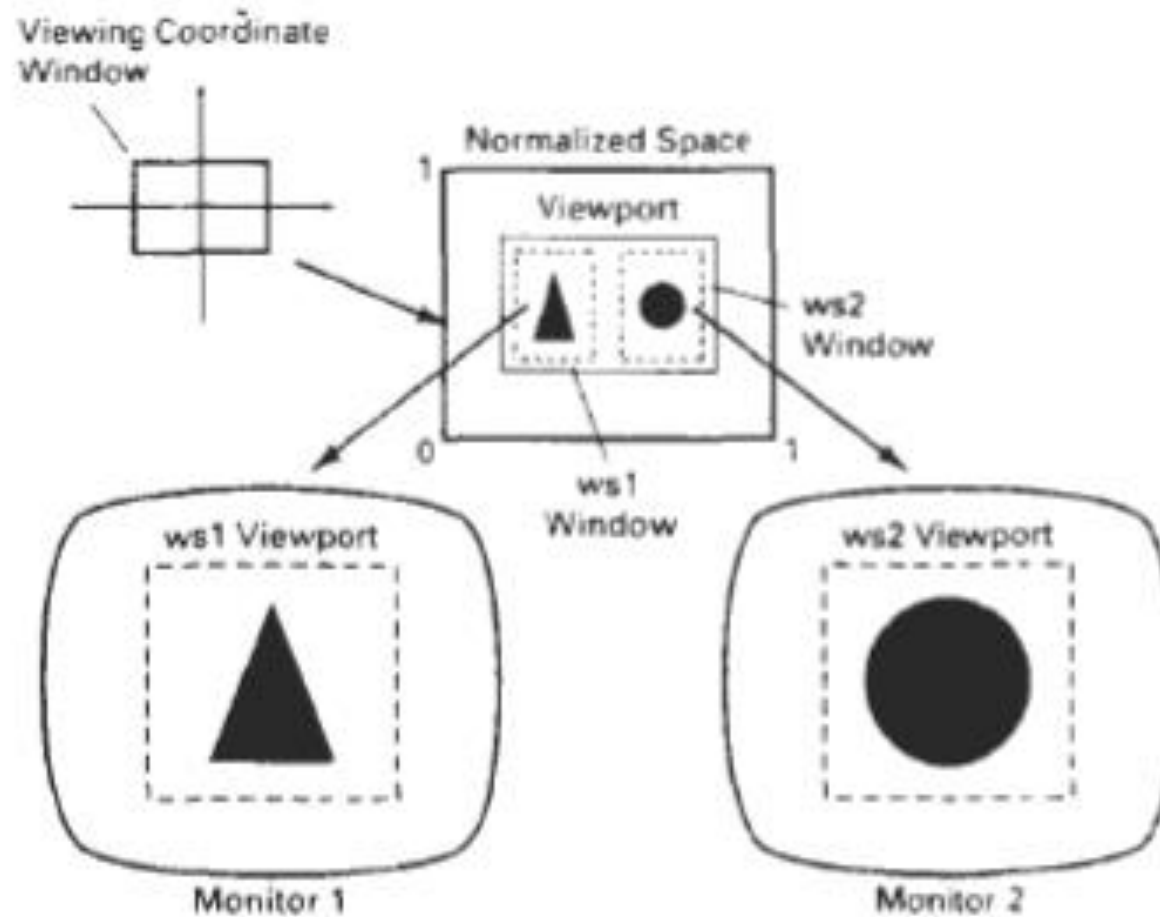
$$sx = \frac{xv_{max} - xv_{min}}{xw_{max} - xw_{min}}$$

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

TRANSFORMASI WINDOW-TO-VIEWPORT

- Langkah :
 - Melakukan scaling menggunakan titik tetap (fixed point) $(x_{w_{min}}, y_{w_{min}})$ yang men-scaling area jendela ke ukuran viewport
 - Mentranslasikan jendela tersebut ke posisi viewport
- Dari normalized viewport ini kemudian ditransformasikan ke device coordinate
- Pemetaan tersebut dinamakan workstation transformation
- Dilakukan dengan memilih area jendela pada normalized viewing coordinate dan memilih viewport area pada layar tampilan

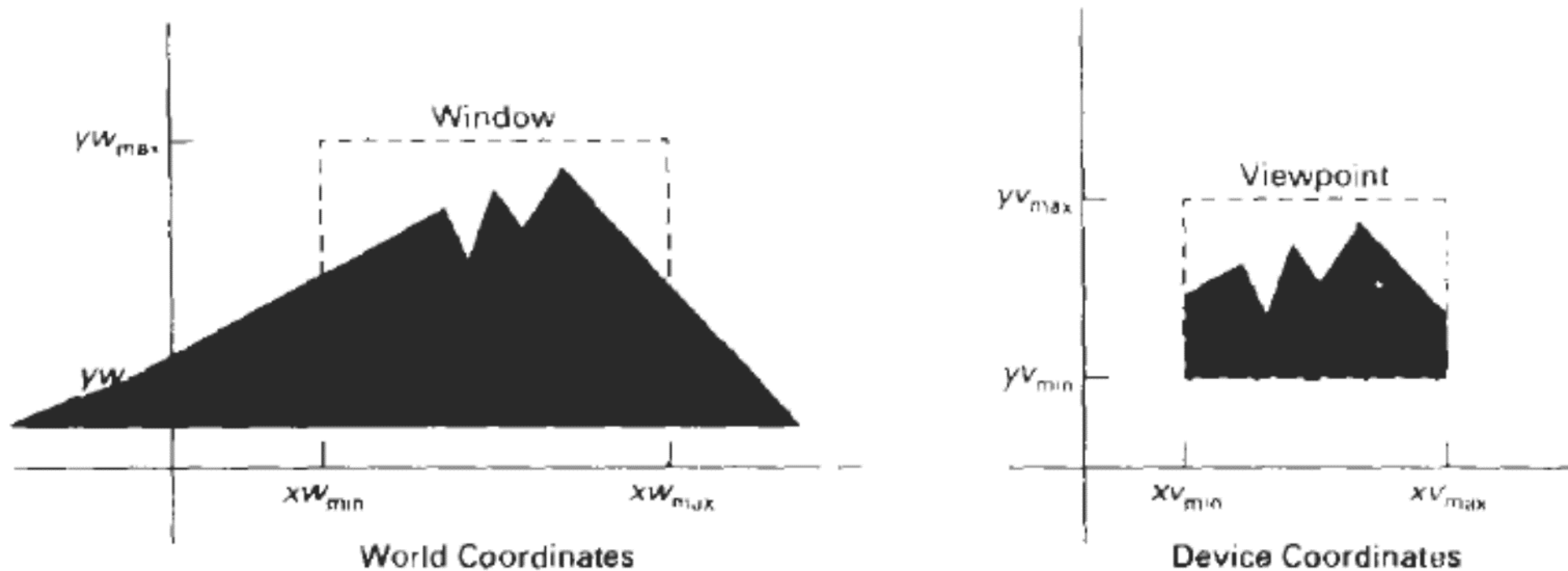
WORKSTATION TRANSFORMATION



CLIPPING

- Prosedur untuk menentukan bagian mana pada citra yang berada didalam atau diluar area yang ditentukan
- Areanya disebut jendela clipping (clipping window)
- Jendela clipping bisa berupa poligon atau kurva
- Biasanya berupa jendela kotak

JENDELA CLIPPING



OPERASI CLIPPING

- Point Clipping
- Line Clipping
- Polygon Clipping
- Curve Clipping
- Text Clipping
- Exterior Clipping

POINT CLIPPING

- Bila jendela clipping berbentuk kotak
- $xw_{min} \leq x \leq xw_{max}$
- $yw_{min} \leq y \leq yw_{max}$
- $(xw_{min}, xw_{max}, yw_{min}, yw_{max})$ = tepian dari jendela clipping
- Titik yang ada diluar batasan jendela ini tidak akan tampil di layar

LINE CLIPPING

Prosedur line clipping :

- Cek apakah keseluruhan garis ada didalam jendela clipping
- Cek apakah keseluruhan garis ada diluar jendela clipping
- Cek perpotongan garis dengan jendela clipping

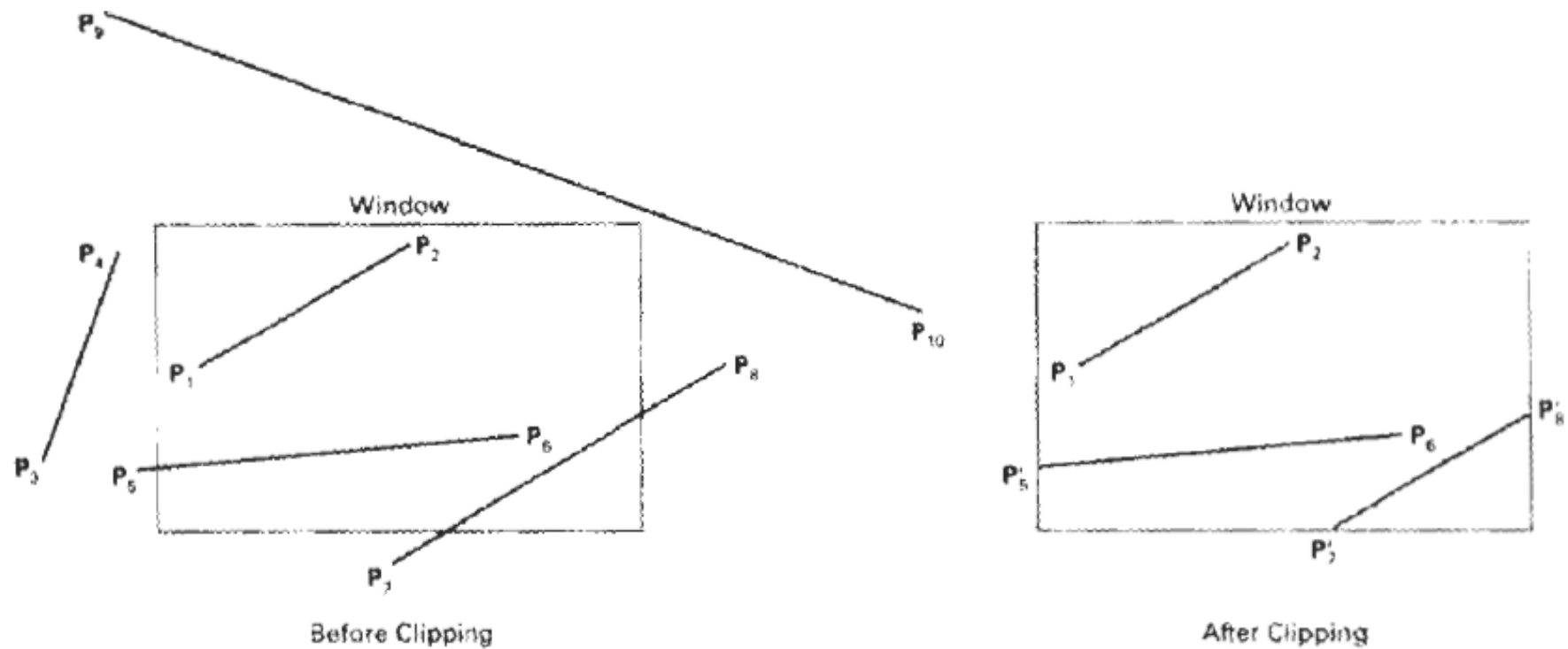
Bila titik ujung garis (x_1, y_1) dan (x_2, y_2)

$$xw = x_1 + u(x_2 - x_1)$$

$$yw = y_1 + u(y_2 - y_1)$$

- Dengan $0 \leq u \leq 1$ dan (xw, yw) adalah batasan jendela clipping
- Bila diuji hasil u diluar batas $0 \leq u \leq 1$ maka garis tidak memasuki jendela clipping di batasan itu

LINE CLIPPING



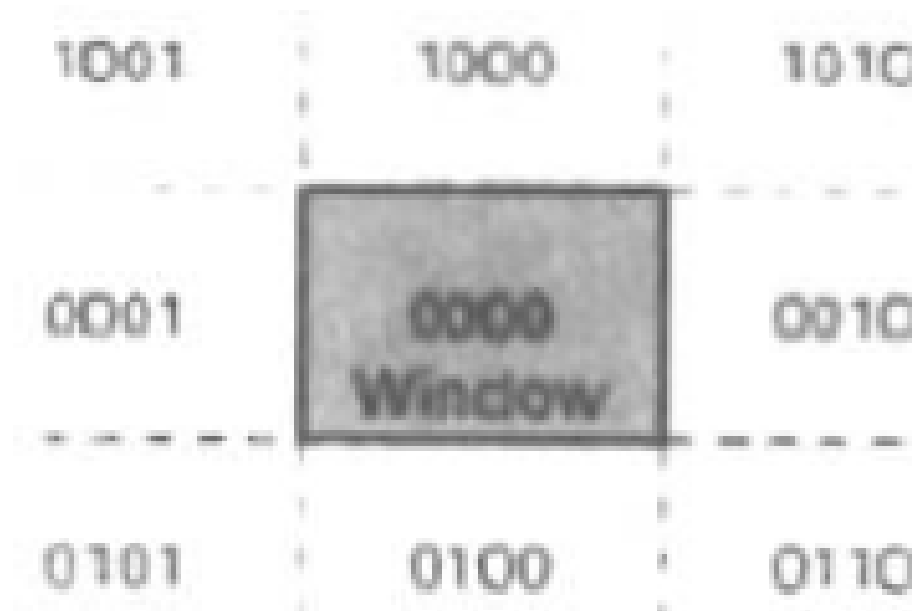
METODE LINE CLIPPING

- Cohen-Sutherland
- Liang-Barsky
- Nicholl-Lee-Nicholl

COHEN-SUTHERLAND

- Menggunakan 4 digit kode biner
- Bit 1 : kiri : $x - xW_{min}$
- Bit 2 : kanan : $xW_{max} - x$
- Bit 3 : bawah : $y - yW_{min}$
- Bit 4 : atas : $yW_{max} - y$
- Titik ujung garis dibandingkan dengan batasan jendela clipping
- Misal bit 1 bernilai 1 bila $x - xW_{min} < 0$
- Bila kode 0000 maka titik ada didalam jendela clipping
- Bila kode 0101 maka titik ada di kiri bawah jendela clipping

COHEN-SUTHERLAND



COHEN-SUTHERLAND

- Untuk garis yang berada sebagian di jendela clipping
- Dihitung perpotongannya dengan garis batas jendela

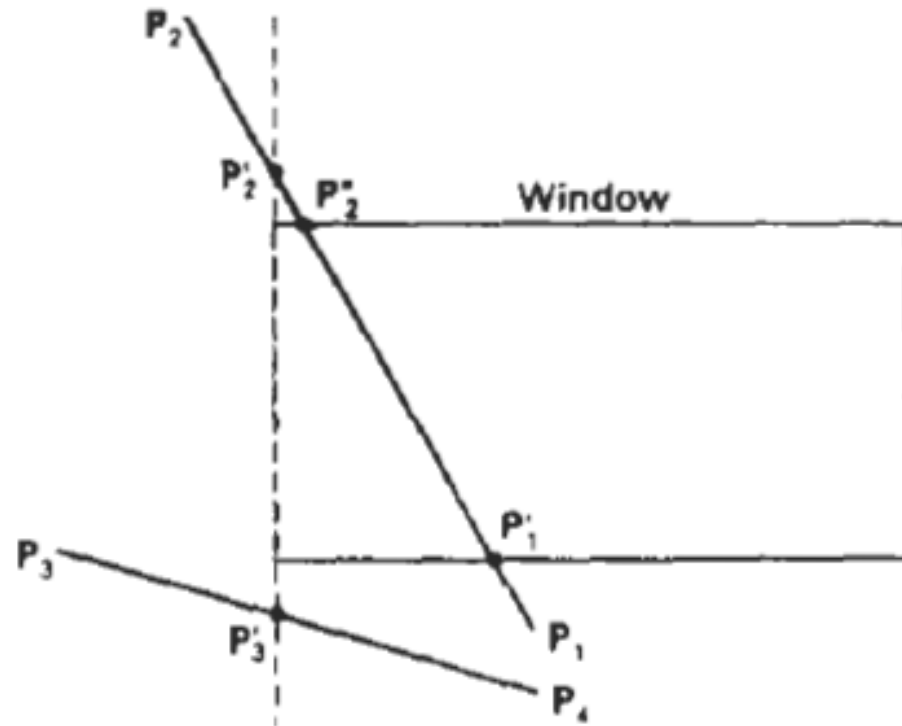
$$y = y_1 + m(xw - x_1)$$

- dengan $xw = xw_{min}$ atau $xw = xw_{max}$ tergantung garis berpotongan dengan garis batas jendela xw_{min} atau xw_{max}

$$x = x_1 + \frac{(yw - y_1)}{m}$$

- dengan $yw = yw_{min}$ atau $yw = yw_{max}$ tergantung garis berpotongan dengan garis batas jendela yw_{min} atau yw_{max}

LINE CLIPPING



LIANG-BARSKY

- Line clipping didefinisikan dengan rumus :
- $xw = x_1 + u(x_2 - x_1) = x_1 + u\Delta x$
- $yw = y_1 + u(y_2 - y_1) = y_1 + u\Delta y$
- Dengan $0 \leq u \leq 1$
- Liang-Barsky memodifikasi persamaan diatas menjadi :
- $xw_{min} \leq x_1 + u\Delta x \leq xw_{max}$
- $yw_{min} \leq y_1 + u\Delta y \leq yw_{max}$

LIANG-BARSKY

- Atau dapat dimisalkan
- $up_k \leq q_k$ untuk $k = 1,2,3,4$
- Dimana :

- | | | |
|---------------------|------------------------|---------|
| • $p_1 = -\Delta x$ | $q_1 = x_1 - xw_{min}$ | (kiri) |
| • $p_2 = \Delta x$ | $q_2 = xw_{max} - x_1$ | (kanan) |
| • $p_3 = -\Delta y$ | $q_3 = y_1 - yw_{min}$ | (bawah) |
| • $p_4 = \Delta y$ | $q_4 = yw_{max} - y_1$ | (atas) |

LIANG-BARSKY

- Bila titik potong garis pada jendela adalah u_1 dan u_2
- Bila $p_k < 0$ maka update u_1 dengan $\max\left(0, \frac{q_k}{p_k}\right)$
- Bila $p_k > 0$ maka update u_2 dengan $\min\left(1, \frac{q_k}{p_k}\right)$
- Bila $u_1 > u_2$ maka garis dibuang
- Bila tidak maka cek :
- Bila $p_k = 0$ dan $q_k < 0$ maka garis dibuang
- Bila tidak maka titik ujung garis diambil dari titik potong dengan jendela

NICHOLL-LEE-NICHOLL (NLN)

Kelebihan :

- Lebih sedikit persyaratannya

Kekurangan :

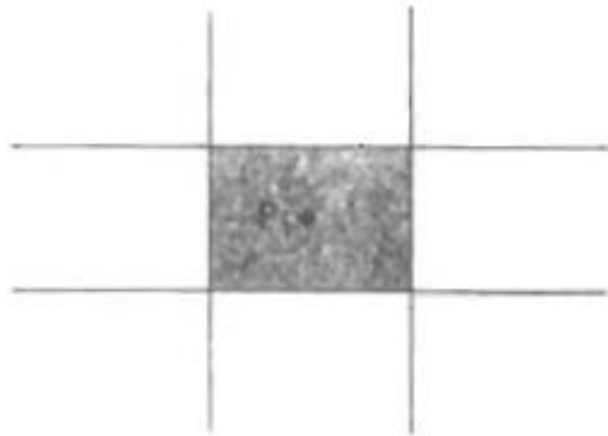
- Hanya bisa diterapkan pada 2D clipping

NICHOLL-LEE-NICHOLL (NLN)

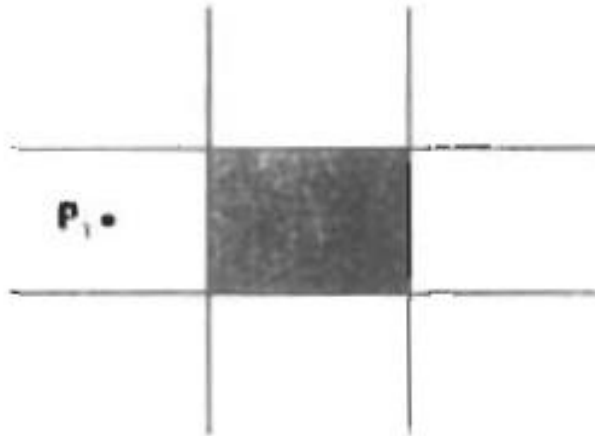
Langkah :

- Bagi area disekeliling jendela menjadi 9 bagian
- Cek letak area titik ujung P_1
- Pindahkan titik ujung P_1 bila ada selain di 3 daerah (didalam jendela, ditepi dan di sudut)
- Tentukan posisi P_2 berdasarkan P_1 dengan menggunakan slope (m)
- Hitung perpotongan dengan jendela clipping

NICHOLL-LEE-NICHOLL (NLN)



P_1 in Window



P_1 in Edge Region



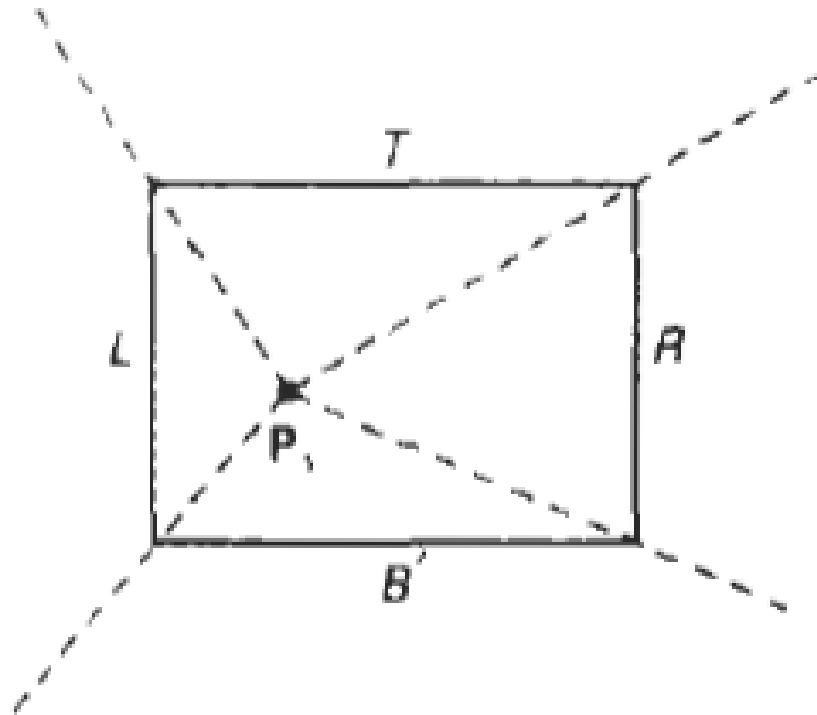
P_1 in Corner Region

NICHOLL-LEE-NICHOLL (NLN)

Aturan bila P_1 ada didalam jendela clipping

- Buat 4 daerah yang merupakan perpotongan P_1 dengan titik sudut jendela clipping beri nama L, T, R, B
- Tentukan daerah P_2
 - Bila didaerah T maka garis diclipping sampai perpotongan P_1 dengan batas atas jendela clipping
 - Lakukan cara yang sama untuk daerah yang lain

P_1 DIDALAM JENDELA CLIPPING

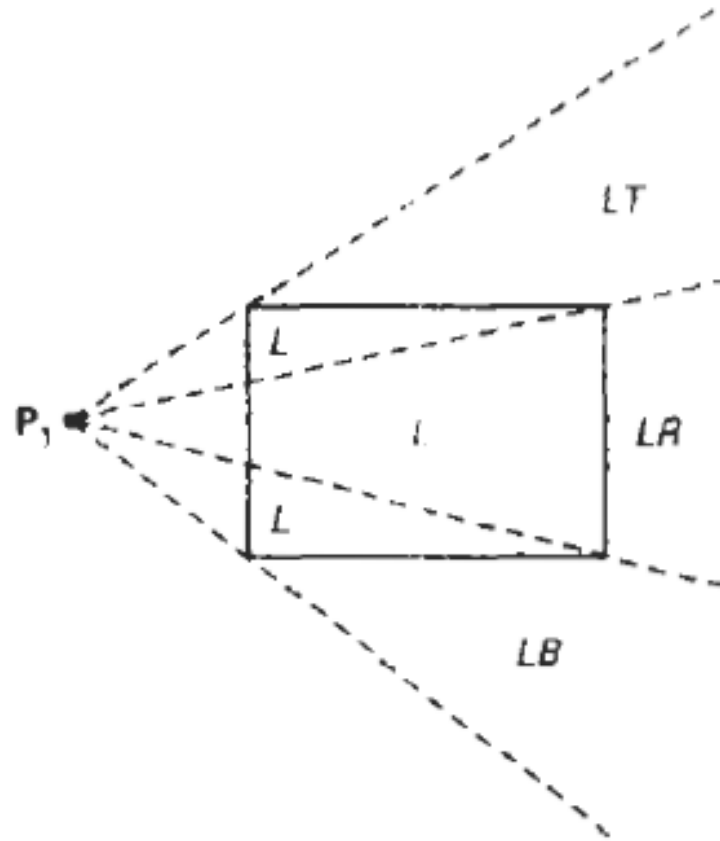


NICHOLL-LEE-NICHOLL (NLN)

Aturan bila P_1 ada dikiri jendela clipping

- Buat 4 daerah yang merupakan perpotongan P_1 dengan titik sudut jendela clipping beri nama L, LT, LR, LB
- Tentukan daerah P_2
 - Bila di daerah L maka garis diclipping dari perpotongan P_1 dengan batas kiri sampai P_2
 - Bila di daerah LT maka garis diclipping dari perpotongan P_1 dengan batas kiri jendela clipping sampai batas atas jendela clipping
 - Lakukan cara yang sama untuk daerah yang lain
 - Bila diluar daerah tersebut garis diabaikan

P_1 ADA DIKIRI JENDELA CLIPPING

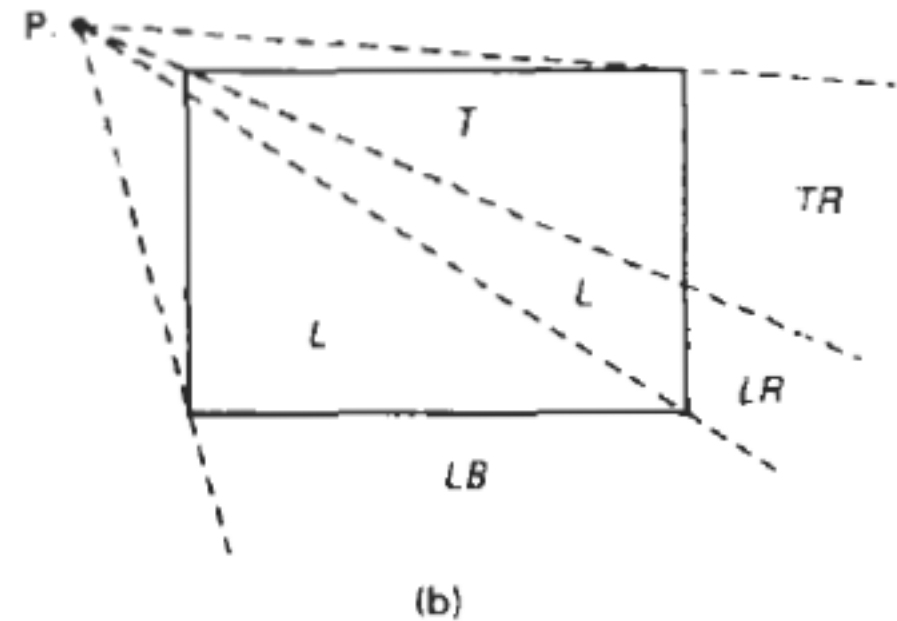
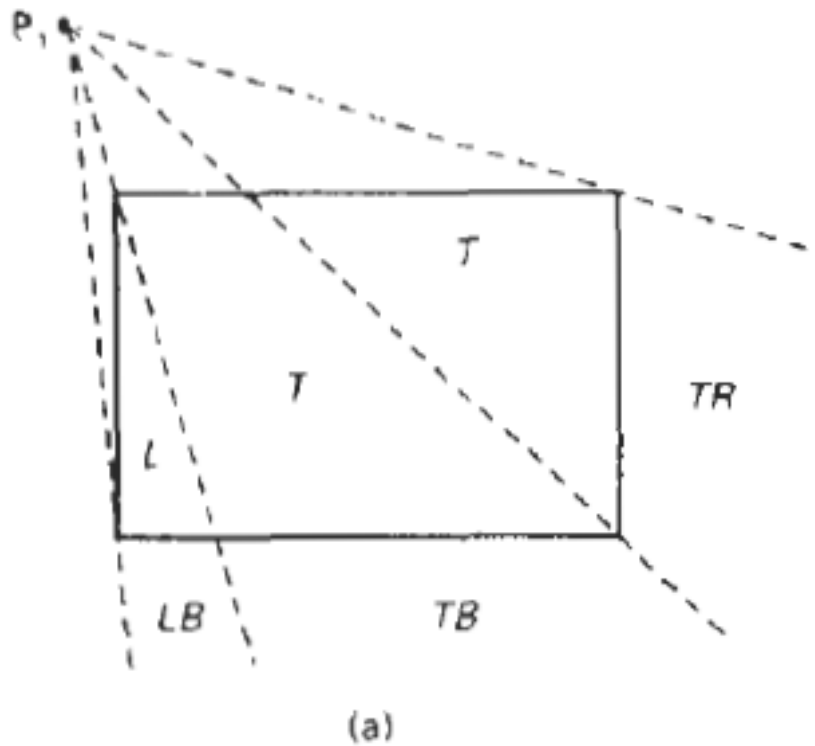


NICHOLL-LEE-NICHOLL (NLN)

Aturan bila P_1 ada di kiri atas jendela clipping

- Buat 6 daerah yang merupakan perpotongan P_1 dengan titik sudut jendela clipping beri nama T, L, TR, TB, LR, LB
- Tentukan daerah P_2 dengan membandingkan slope garis P_1-P_2 dengan slope P_1 dan batas jendela
- Lakukan cara yang sama untuk daerah yang lain
- Bila diluar daerah tersebut garis diabaikan

P_1 ADA DI KIRI ATAS JENDELA CLIPPING



POLYGON CLIPPING

- Poligon diekstrak menjadi titik dan segmen garis
- Membuat area tertutup dari hasil clipping

SUTHERLAND-HODGEMAN

- Setiap segment garis pada poligon akan dicek bila melewati batas jendela clipping :
- Bila titik pertama ada diluar jendela clipping dan titik kedua ada didalam maka titik perpotongan titik pertama dengan jendela dan titik kedua disimpan sebagai titik baru
- Bila titik pertama dan kedua ada didalam jendela clipping maka titik kedua disimpan sebagai titik baru
- Bila titik pertama ada didalam jendela clipping dan titik kedua ada diluar maka perpotongan titik pertama dengan jendela batas akan disimpan sebagai titik baru
- Bila semua titik ada diluar jendela maka tidak disimpan sebagai titik baru
- Keluaran titik baru ini akan dicek terhadap batas jendela yang lain

SUTHERLAND-HODGEMAN



Original
Polygon



Clip
Left



Clip
Right



Clip
Bottom



Clip
Top

SUTHERLAND-HODGEMAN

Kelemahan :

- Bila melakukan clipping pada polygon konkaf hasil clipping akan terdapat garis tambahan pada polygon

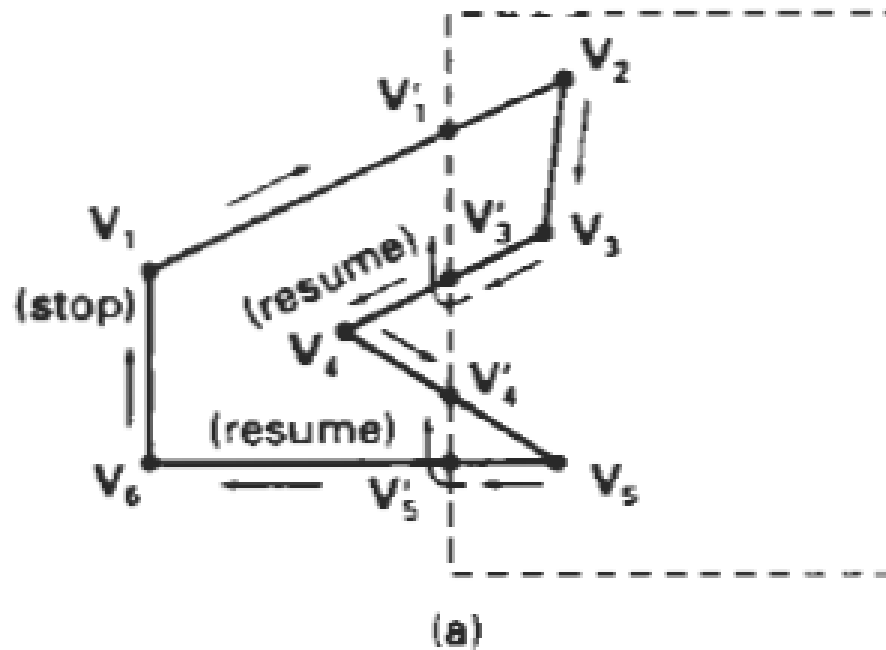
Solusi :

- Membagi polygon konkaf menjadi beberapa polygon konvek kemudian diclipping
- Mendeteksi hasil akhir dari clipping dan menghubungkan titik-titiknya dengan benar

WEILER-ATHERTON

- Menyempurnakan algoritma dari Sutherland-Hodgeman
- Algoritmanya tergantung dari arah polygon dan pasangan titik pada polygon pada arah luar ke dalam atau dalam ke keluar
- Bila urutan titik pada polygon searah jarum jam maka :
 - Untuk pasangan titik arah luar ke dalam ikuti tepian polygon
 - Untuk pasangan titik arah dalam ke luar ikuti tepian jendela searah jarum jam

WEILER-ATHERTON

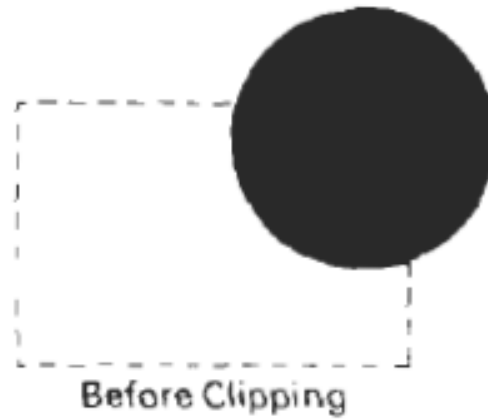


(b)

CURVE CLIPPING

- Prosedur untuk clipping kurva sama dengan clipping polygon
- Hanya menggunakan persamaan garis non linear untuk menentukan perpotongan dengan batas jendela clipping
- Untuk obyek lingkaran atau ellipse clipping dapat diperingkas dengan membagi lingkaran atau ellipse menjadi 4 kuadran

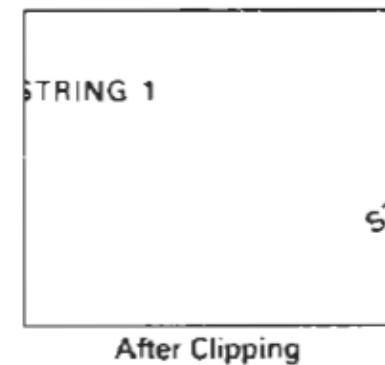
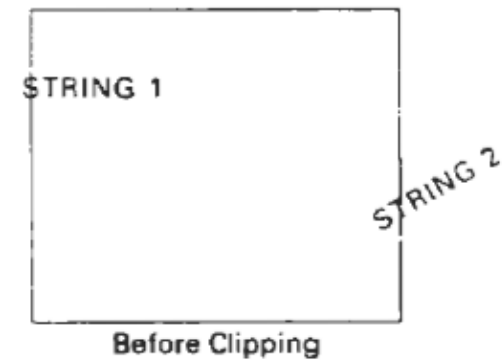
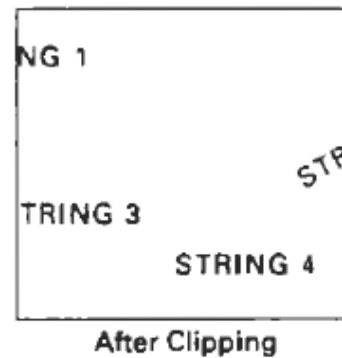
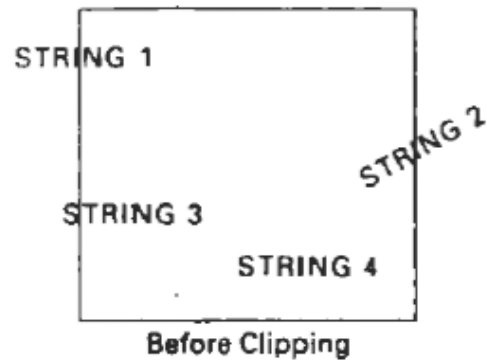
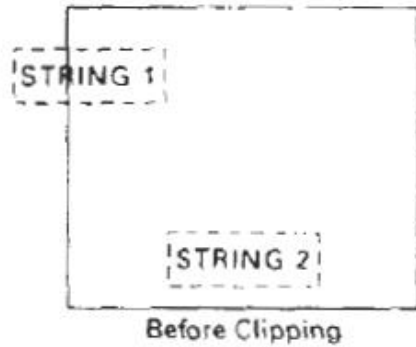
CURVE CLIPPING



TEXT CLIPPING

- Clipping teks dilakukan dengan beberapa cara :
- All or none string clipping :
 - String teks diwakili dengan area kotak yang melingkupi string teks
 - Bila semua string teks ada didalam jendela clipping maka teks akan tampil dilayar
 - Bila hanya sebagian maka semua teks dibuang
- All or none character clipping
 - Setiap karakter diwakili dengan area kotak yang melingkupi karakter
 - Karakter yang sebagian ada di dalam jendela akan diclipping menurut segment garis yang membentuk karakter tersebut

TEXT CLIPPING



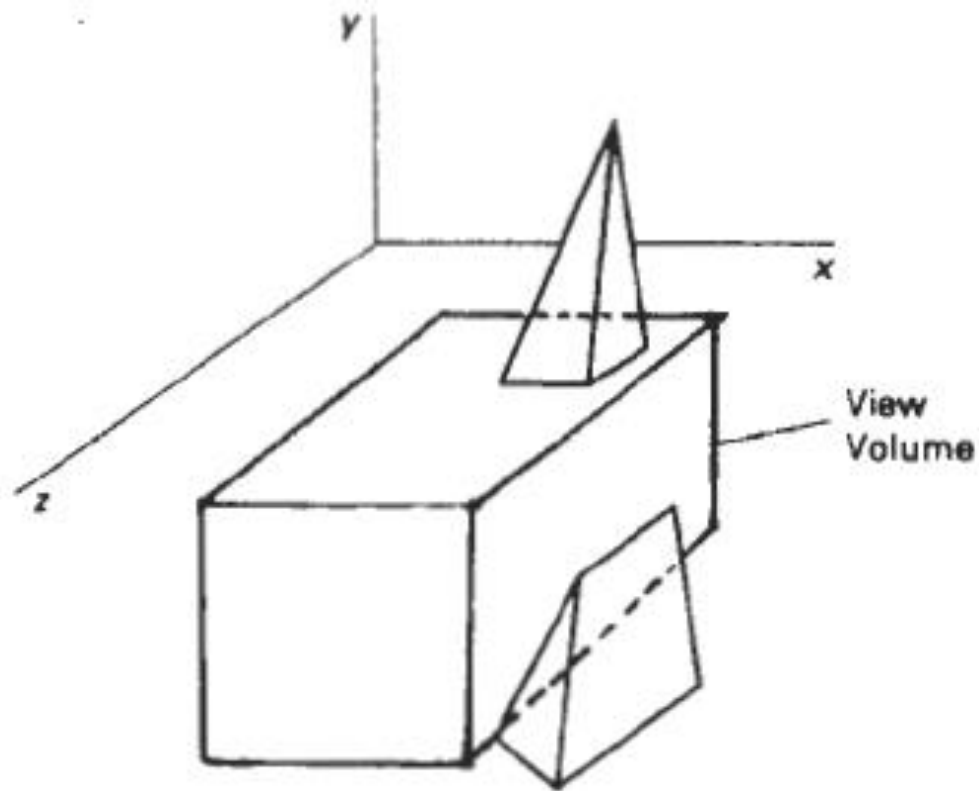
EXTERIOR CLIPPING

- Clipping eksterior akan membuang obyek didalam jendela dan menampilkan obyek diluar jendela
- Digunakan pada tampilan dengan banyak layar

3D CLIPPING

- 3D clipping lebih mudah dilakukan setelah koordinat proyeksi diterapkan
- Permukaan batas pandang akan tegak lurus dengan setiap sumbu koordinat
- Proyeksi orthogonal view volumenya sudah berupa pipa kotak yang parallel
- Proyeksi miring view volumenya akan dikonversi dengan transformasi shear untuk mendapatkan view volume kotak yang parallel
- Proyeksi perspektif akan dikonversi dengan kombinasi transformasi shear-scale untuk mendapatkan view volume yang parallel
- Normalisasi perlu dilakukan setelah transformasi proyeksi
- Digunakan agar membuat view volume independent terhadap layar
- Prosedur sama seperti normalisasi viewport hanya ditambah sumbu Z

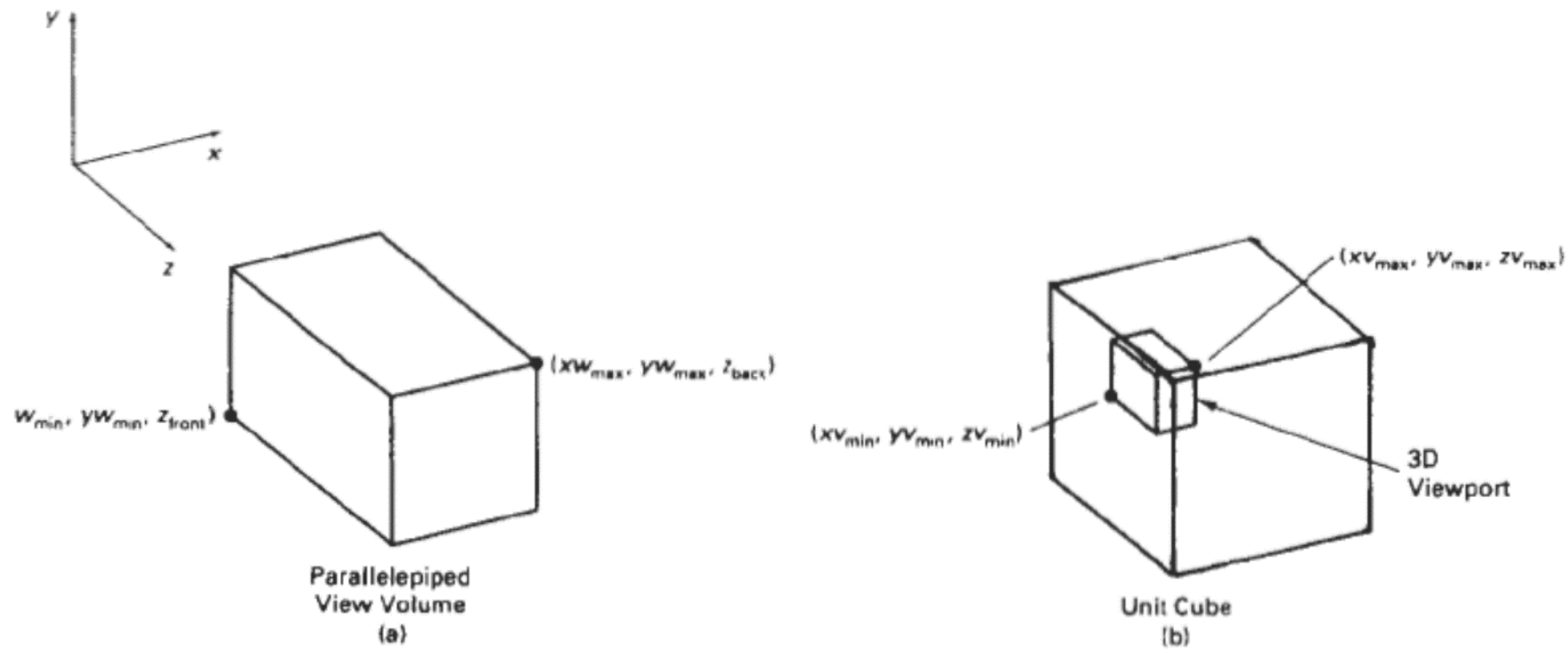
3D CLIPPING



VIEWPORT CLIPPING

- Bila menggunakan metode bit :
- Menggunakan 6 digit kode biner
- Bit 1 : kiri : bernilai 1 bila $x < xv_{min}$
- Bit 2 : kanan : bernilai 1 bila $x > xv_{max}$
- Bit 3 : bawah : bernilai 1 bila $y < yv_{min}$
- Bit 4 : atas : bernilai 1 bila $y > yv_{max}$
- Bit 5 : depan : bernilai 1 bila $z < zv_{max}$
- Bit 6 : belakang : bernilai 1 bila $z > zv_{max}$

VIEWPORT CLIPPING



VIEWPORT CLIPPING

- Misalnya kode 101000 menunjukkan bahwa titik berada di atas dan belakang viewport
- Kode 000000 menunjukkan bahwa titik ada di dalam view volume

VIEWPORT CLIPPING

- Untuk persamaan garisnya :

$$x = x_1 + u(x_2 - x_1) = x_1 + u\Delta x$$

$$y = y_1 + u(y_2 - y_1) = y_1 + u\Delta y$$

$$z = z_1 + u(z_2 - z_1) = z_1 + u\Delta z$$

- Dengan $0 \leq u \leq 1$
- Perpotongan garisnya dihitung dengan u

$$u = \frac{z_{v_{min}} - z_1}{z_2 - z_1}$$

- Bila u tidak berada di rentang $0 - 1$ maka garis tidak berpotongan dengan bidang

VIEWPORT CLIPPING

