

## **MODEL PEMBELAJARAN ADAPTIF PADA MATA KULIAH GRAFIKA KOMPUTER**

**Veronica lusiana<sup>1</sup>, Budi hartono<sup>2</sup>**

Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Stikubank  
Semarang

Jl Tri Lomba Juang No 1 Semarang

Telp. (0224) 8311668

E-mail: [vero@edu.unisbank.ac.id](mailto:vero@edu.unisbank.ac.id)<sup>1</sup>, [budihartono@edu.unisbank.ac.id](mailto:budihartono@edu.unisbank.ac.id)<sup>2</sup>

### **ABSTRAK**

Konsep pembelajaran adaptif menawarkan proses belajar yang melihat kemampuan pemahaman mahasiswa. Pembelajaran adaptif pada umumnya lebih dititikberatkan untuk proses ujian yang dikenal dengan istilah ujian adaptif menggunakan komputer (computer adaptive test, CAT; computer-based test, CBT). Disini, urutan soal yang dikerjakan mahasiswa berbeda antara satu mahasiswa dengan yang lain sesuai dengan kemampuan mengerjakan soal oleh mahasiswa. Tingkat kesulitan soal yang diberikan atau dimunculkan oleh komputer bervariasi untuk setiap mahasiswa sesuai dengan skor yang dihasilkan dari jawaban sebelumnya. Pada penelitian ini akan menerapkan konsep pembelajaran adaptif yang diterapkan pada urutan pemberian suatu bahan ajar yaitu sub-pokok bahasan kepada mahasiswa. Sub-pokok bahasan disini adalah yang bersifat pengayaan atau sub-pokok bahasan lanjutan, karena pokok bahasan pengantar atau pemahaman dasar masih diberikan secara seragam untuk seluruh mahasiswa. Urutan bahan ajar yang bersifat pengayaan diberikan secara adaptif mengikuti tingkat penerimaan mahasiswa terhadap pokok bahasan sebelumnya. Melalui model pembelajaran ini diharapkan mahasiswa dapat menguasai pengetahuan dasar mata kuliah Grafika Komputer dan menguasai pokok bahasan lanjutan atau pengembangan secara lebih baik sesuai dengan kemampuan penalarannya. Pokok bahasan yang dipilih disini adalah clipping menggunakan algoritma Cohen-Sutherland.

Kata Kunci: computer adaptive test, adaptive learning, algoritma cohen-sutherland

### **1. PENDAHULUAN**

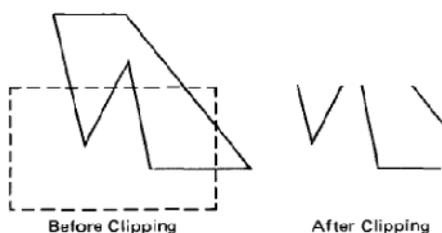
Konsep pembelajaran adaptif menawarkan proses belajar yang melihat kemampuan pemahaman siswa. Pembelajaran adaptif pada umumnya lebih dititikberatkan untuk proses ujian yang dikenal dengan istilah ujian adaptif menggunakan komputer (computer adaptive test, CAT; computer-based test, CBT). Tingkat kesulitan soal yang diberikan atau dimunculkan oleh komputer bervariasi untuk setiap siswa sesuai dengan skor yang dihasilkan dari jawaban sebelumnya. Penelitian ini akan menerapkan konsep pembelajaran adaptif yang diterapkan pada urutan pemberian suatu bahan ajar yaitu sub-pokok bahasan kepada mahasiswa. Sub-pokok bahasan disini adalah yang bersifat pengayaan atau sub-pokok bahasan lanjutan, karena pokok bahasan pengantar atau pemahaman dasar masih diberikan secara seragam untuk seluruh mahasiswa. Urutan bahan ajar yang bersifat pengayaan diberikan secara adaptif mengikuti tingkat penerimaan mahasiswa terhadap pokok bahasan sebelumnya.

Mata kuliah Grafika Komputer dipilih sebagai studi kasus untuk penelitian ini. Bahan ajar ini telah beberapa semester diajarkan oleh tim peneliti sehingga lebih memudahkan untuk penyusunan modul bahan ajar yang bersifat adaptif. Grafika Komputer adalah mata kuliah yang cukup menarik bagi siswa, secara umum menjelaskan konsep penciptaan dan manipulasi obyek dua dimensi dan tiga dimensi (Lusiana (2015), Pandey dan Jain (2013), Kodituwakku, dkk. (2012)). Didalamnya banyak mengimplementasikan teori matematika diskrit, trigonometri, aljabar linier, dan matriks. Pokok bahasan yang dipilih disini adalah pemotongan garis (*clipping*) menggunakan Algoritma Cohen-Sutherland.

## 2. TINJAUAN PUSTAKA

### 2.1 Pemotongan garis (*clipping*)

Pemotongan perlu dilakukan terhadap sebagian dari obyek yang akan digambarkan pada jendela, yaitu apabila obyek yang ditampilkan memiliki luas yang lebih besar dibandingkan dengan luas jendela. Pada prinsipnya, pemotongan ini akan memisahkan bagian obyek yang tampak dan bagian yang tidak tampak pada jendela. Seperti telah kita ketahui bahwa penyusun terkecil dari gambar adalah titik dan garis yang ditampilkan dengan intensitas warna dan pada lokasi tertentu (Santosa, 1996; Nugroho, 2005). Pada Gambar 1 dapat dilihat contoh pemotongan terhadap obyek rangka dimensi dua, setelah pemotongan maka obyek digambar mengikuti luas jendela.

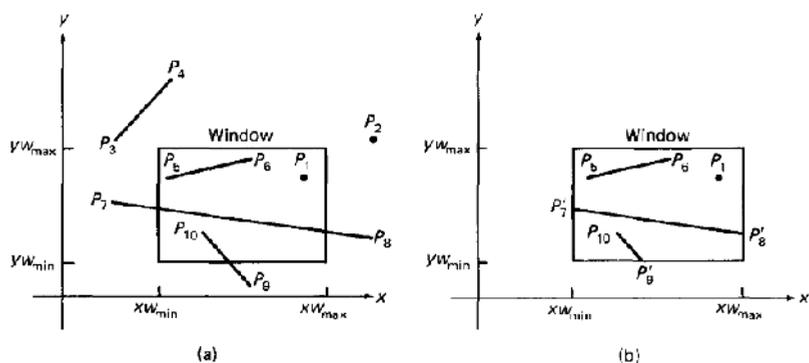


Gambar 1. Pemotongan pada obyek rangka dimensi dua (Hearn dan Baker, 1986)

Pemotongan garis dilakukan jika terdapat bagian garis yang terletak di luar jendela. Pada Gambar 2(a) dapat dilihat beberapa buah garis dalam tiga kondisi yang berbeda yaitu:

- Garis ditampilkan seluruhnya di jendela (garis P5-P6)
- Garis tidak ditampilkan di jendela (garis P3-P4)
- Garis ditampilkan sebagian di jendela (garis P2-P8, P9-P10)

Setelah dilakukan proses pemotongan maka hasilnya dapat dilihat pada Gambar 2(b).



Gambar 2. Pemotongan obyek garis pada ruang dimensi dua (Hearn dan Baker, 1986)

## 2.2 Algoritma Cohen-Sutherland pada Ruang Dimensi Dua

Algoritma Cohen-Sutherland mengusulkan cara pengujian pada titik awal dan titik akhir garis untuk menentukan apakah sebuah garis berada di dalam atau di luar jendela, tanpa harus menguji seluruh titik penyusun garis tersebut. Algoritma ini membagi sebanyak 9 wilayah yang digunakan untuk memetakan titik awal dan titik akhir garis yang akan digambar menggunakan pendekatan ruang dimensi dua (Pandey dan Jain, 2013). Masing-masing wilayah ini diberi kode yang unik. Pada Gambar 3 adalah contoh penamaan kode 4 bit untuk setiap wilayah pada dimensi dua. Setiap bit dapat bernilai 0 atau 1 digunakan untuk menyimpan informasi letak titik. Empat bit ini jika dibaca berurutan menyimpan informasi [atas/top (bit ke 3), bawah/bottom (bit ke 2), kanan/right (bit ke 1), kiri/left (bit ke 0)].

1001	1000	1010	$Y = Y_{max}$
0001	0000	0010	
0101	0100	0110	$Y = Y_{min}$
$x = x_{min} \quad x = x_{max}$			

Gambar 3. Kode 4 bit algoritma Cohen-Sutherland (Angel dan Shreiner, 2012)

Deskripsi wilayah tengah diberi kode 0000 berarti: tidak di atas  $y_{max}$ , tidak di bawah  $y_{min}$ , tidak di sebelah kanan  $x_{max}$ , dan tidak di sebelah kiri  $x_{min}$ . Wilayah ini merupakan jendela tempat garis digambarkan, apabila terdapat garis atau potongan garis di luar wilayah ini maka tidak akan digambar. Sedangkan untuk wilayah paling kanan atas diberi kode 1010 berarti: ya di atas  $y_{max}$ , tidak di bawah  $y_{min}$ , ya di sebelah kanan  $x_{max}$ , dan tidak di sebelah kiri  $x_{min}$ . Demikian seterusnya untuk wilayah yang lain.

Berdasarkan pembagian wilayah pada algoritma Cohen-Sutherland maka hasil uji sebuah garis dapat dibagi menjadi tiga kategori, yaitu:

a. Garis terlihat seluruhnya (*fully visible*)

Garis terlihat seluruhnya apabila titik awal dan akhir garis berada di dalam jendela. Garis yang memiliki titik ujung T1 ( $X_{t1}, Y_{t1}$ ) dan T2 ( $X_{t2}, Y_{t2}$ ) akan terletak di dalam *viewport* jika memenuhi kedua persamaan berikut ini, lihat Persamaan (1) dan Persamaan (2).

$$X_{min} \leq X_{t1}, X_{t2} \leq X_{max} \tag{1}$$

$$Y_{min} \leq Y_{t1}, Y_{t2} \leq Y_{max}$$

(2)

Melihat pada kode pembagian wilayah pada Gambar 3 maka garis terlihat jika kedua titik ujungnya mempunyai kode 0000.

b. Garis tidak terlihat sama sekali (*fully invisible*)

Garis tidak terlihat sama sekali atau seluruh garis terletak di luar jendela. Garis yang memiliki titik ujung T1 ( $X_{t1}, Y_{t1}$ ) dan T2 ( $X_{t2}, Y_{t2}$ ) akan terletak di luar jendela jika salah satu dari empat persamaan berikut ini dipenuhi, lihat Persamaan (3), Persamaan (4), Persamaan (5), dan Persamaan (6).

$$X_{t1}, X_{t2} > X_{max}$$

(3)

$$X_{t1}, X_{t2} < X_{min}$$

(4)

$$Y_{t1}, Y_{t2} > Y_{max}$$

(5)

$$Y_{t1}, Y_{t2} < Y_{min}$$

(6)

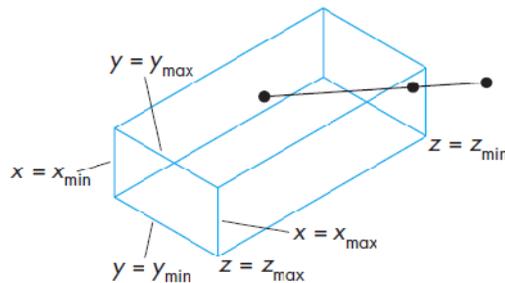
Melihat pada kode pembagian wilayah pada Gambar 3 maka garis tidak terlihat jika terdapat nilai 1 pada urutan bit yang sama dari kedua titik ujung garis, sehingga hasil operasi AND dari kedua ujung garis tidak menghasilkan 0000.

c. Garis terlihat sebagian (*partially visible* atau *clipping candidate*)

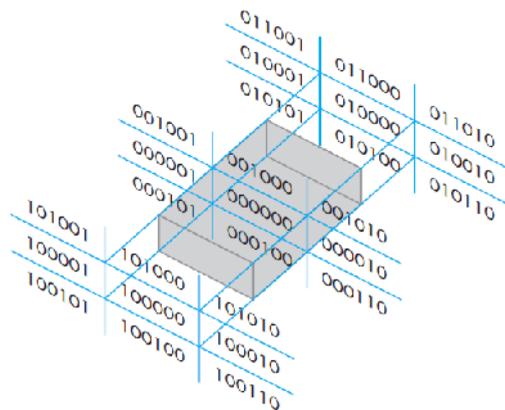
Garis terlihat sebagian atau ada sebagian garis yang tidak tampak di dalam jendela, disini garis menjadi kandidat untuk dipotong. Kondisi ini terjadi jika syarat a dan b tidak terpenuhi.

### 2.3 Algoritma Cohen-Sutherland pada Ruang Dimensi Tiga

Pengujian pada titik awal dan titik akhir garis pada ruang dimensi tiga, maka algoritma Cohen-Sutherland membagi sebanyak 27 wilayah. Masing-masing wilayah ini diberi kode yang unik (Kodituwakku, dkk., 2012). Ini adalah perluasan dari pendekatan dimensi dua. Pada Gambar 4 digunakan untuk menyusun deskripsi 27 kode wilayah.



Gambar 4. Deskripsi wilayah algoritma Cohen-Sutherland pada ruang dimensi tiga (Angel dan Shreiner, 2012)



Gambar 5. Kode 6 bit algoritma Cohen-Sutherland (Angel dan Shreiner, 2012)

Pada Gambar 5 merupakan contoh penamaan kode 6 bit untuk setiap wilayah pada ruang dimensi tiga. Setiap bit dapat bernilai 0 atau 1 digunakan untuk menyimpan informasi letak titik awal dan akhir sebuah garis. Enam bit ini jika dibaca berurutan menyimpan informasi [depan/front (bit ke 5), belakang/back (bit ke 4), atas/top (bit ke 3), bawah/bottom (bit ke 2), kanan/right (bit ke 1), dan kiri/left (bit ke 0)]. Apabila melihat kembali pada Gambar 4, maka diperoleh:

- bit ke 5 [depan/front; 0 = tidak di depan  $z_{max}$ ; 1 = di depan  $z_{max}$ ]
- bit ke 4 [belakang/back; 0 = tidak di belakang  $z_{min}$ ; 1 = di belakang  $z_{min}$ ]
- bit ke 3 [atas/top; 0 = tidak di atas  $y_{max}$ ; 1 = di atas  $y_{max}$ ]
- bit ke 2 [bawah/bottom; 0 = tidak di bawah  $y_{min}$ ; 1 = di bawah  $y_{min}$ ]

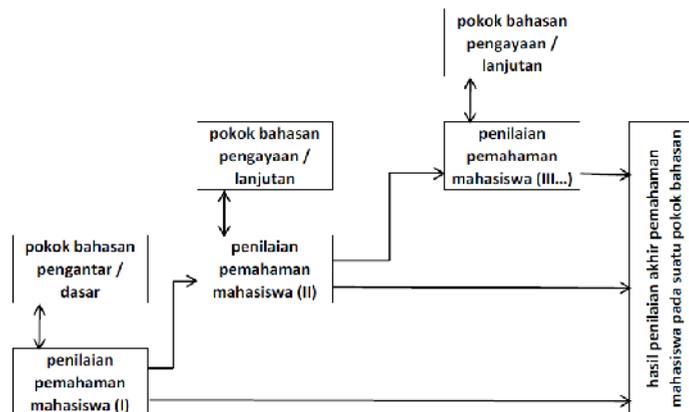
- e. bit ke 1 [kanan/right; 0 = tidak di sebelah kanan  $x_{max}$ ; 1 = di sebelah kanan  $x_{max}$ ]
- f. bit ke 0 [kiri/left; 0 = tidak di sebelah kiri  $x_{min}$ ; 1 = tidak di sebelah kiri  $x_{min}$ ]

Contoh, deskripsi wilayah tengah diberi kode 000000 berarti: tidak di depan  $z_{max}$ , tidak di belakang  $z_{min}$ , tidak di atas  $y_{max}$ , tidak di bawah  $y_{min}$ , tidak di sebelah kanan  $x_{max}$ , dan tidak di sebelah kiri  $x_{min}$ . Wilayah ini merupakan jendela tempat garis digambarkan, apabila garis berada di luar wilayah ini maka tidak akan digambarkan. Deskripsi untuk wilayah paling belakang kiri bawah diberi kode 010101 berarti: tidak di depan  $z_{max}$ , ya dibelakang  $z_{min}$ , tidak di atas  $y_{max}$ , ya di bawah  $y_{min}$ , tidak di sebelah kanan  $x_{max}$ , dan ya di sebelah kiri  $x_{min}$ . Demikian seterusnya untuk wilayah yang lain.

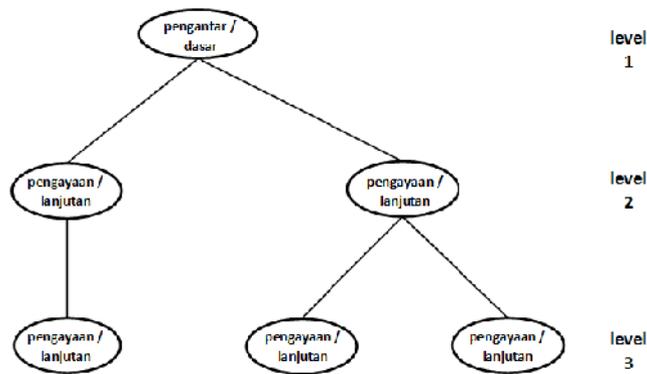
### 3. HASIL DAN PEMBAHASAN

#### 3.1 Alur Penilaian Pemahaman Pokok Bahasan

Penilaian pemahaman oleh mahasiswa terhadap suatu pokok bahasan diperoleh dari hasil jawaban latihan soal pilihan ganda sesuai dengan ruang lingkup pembahasan. Alur penilaian selengkapny dapat dilihat pada Gambar 6. Penilaian pemahaman oleh mahasiswa (I) adalah untuk pokok bahasan pengantar atau dasar, proses penilaian dapat dikerjakan lebih dari satu kali. Apabila nilai hasil penilaian adalah baik, contoh diatas 70, maka dapat dilanjutkan menuju penilaian pemahaman mahasiswa (II) yaitu untuk pokok bahasan pengayaan atau lanjutan. Sebaliknya jika hasil penilaian adalah sama dengan atau kurang dari 70 maka ini merupakan hasil penilaian akhir pemahaman oleh mahasiswa, demikian seterusnya. Pada Gambar 6 dicontohkan sampai dengan penilaian pemahaman mahasiswa (III) yaitu untuk pokok bahasan pengayaan tingkat (*level*) 2.



Gambar 6. Alur penilaian pemahaman mahasiswa pada suatu pokok bahasan

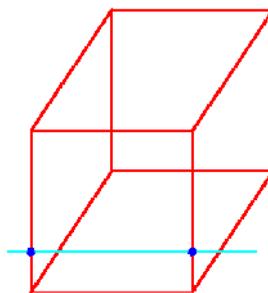


Gambar 7. Pohon keputusan tingkat kesulitan soal latihan

Tingkat kesulitan soal latihan mengikuti pohon keputusan (*decision tree*) seperti dapat dilihat pada Gambar 7. Contoh pokok bahasan *clipping* menggunakan algoritma Cohen-Sutherland, pada *level 1* yaitu pengantar atau dasar pemahaman *clipping* secara umum dan penyusunan kode Cohen-Sutherland pada ruang dimensi dua. Pada *level 2* yaitu pengayaan membahas konsep ruang dimensi tiga dan penyusunan kode Cohen-Sutherland. Pada *level 3*, pembahasan sama dengan *level 2* yang ditambah dengan membuat implementasi ke dalam kode program, disini menggunakan contoh kode program Processing.

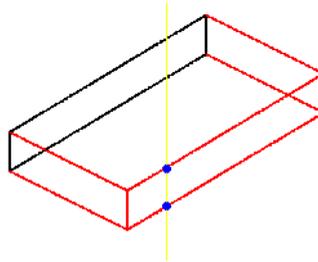
### 3.2 Pokok Bahasan Pengayaan/Lanjutan

Berikut ini adalah contoh pokok bahasan pengayaan atau lanjutan untuk pembahasan *clipping* menggunakan algoritma Cohen-Sutherland. Contoh materi pengayaan untuk level 2, pada Gambar 8 garis disusun dari ujung garis yang berada di kiri ( $u_i$ ) sampai dengan ujung garis yang berada di kanan ( $u_{ii}$ ). Kode untuk ( $u_i$ ) adalah 000001 sedangkan kode untuk ( $u_{ii}$ ) adalah 000010. Ruas garis yang berada di dalam jendela dimensi tiga memiliki batas  $\{x=x_{min}; z=z_{max}\}$  di titik awal dan  $\{x=x_{max}; z=z_{max}\}$  di titik akhir pada batas jendela tersebut. Garis yang ada diluar batas jendela tersebut akan dipotong atau tidak ditampilkan.



Gambar 8. Clipping pada ruang dimensi tiga (i)

Contoh materi pengayaan untuk level 3, pada Gambar 9, garis disusun dari ujung garis yang berada di atas ( $u_i$ ) sampai dengan ujung garis yang berada di bawah ( $u_{ii}$ ). Kode untuk ( $u_i$ ) adalah 001000 sedangkan kode untuk ( $u_{ii}$ ) adalah 000100. Ruas garis yang berada di dalam jendela dimensi tiga memiliki batas  $\{x=x_{max}; y=y_{max}\}$  di titik awal dan  $\{x=x_{max}; y=y_{min}\}$  di titik akhir pada batas jendela tersebut. Garis yang ada diluar batas jendela tersebut akan dipotong atau tidak ditampilkan. Kode program Processing dapat dilihat pada Gambar 10.



Gambar 9. Clipping pada ruang dimensi tiga (ii)

```

1 void setup()
2 { size(600,600); background(255); }
3
4 void draw()
5 {
6   fill(5); textSize(16);
7   stroke(0,0,0); strokeWeight(2);
8   line(50,300,50,350); line(50,300,300,150);
9   line(300,150,300,200); line(50,350,300,200);
10  stroke(255,0,0);
11  line(50,350,200,425); line(200,425,200,375);
12  line(200,375,50,300); line(300,150,450,225);
13  line(450,225,200,375); line(450,225,450,275);
14  line(450,275,300,200); line(450,275,200,425);
15  stroke(255,255,0);
16  line(250,100,250,500);
17  stroke(0,0,255); strokeWeight(10);
18  point(250,395); point(250,347);
19 }

```

Gambar 10. Kode program

#### 4. KESIMPULAN SARAN

Kesimpulan dan saran dari penelitian ini adalah:

- a. Konsep pembelajaran adaptif dapat lebih memudahkan dan meningkatkan motivasi belajar mahasiswa, karena materi belajar disesuaikan dengan kemampuan pemahaman mahasiswa.
- b. Konsep pembelajaran adaptif diharapkan dapat diterapkan pada mata kuliah yang lain, dengan melakukan penyesuaian terhadap karakteristik mata kuliah tersebut.
- c. Menyusun modul pembelajaran yang lebih menarik dan mudah digunakan oleh pengguna, modul yang dapat mengintegrasikan bahan ajar yang bersifat adaptif, soal latihan, dan sistem penilaian.

#### UCAPAN TERIMA KASIH

Penelitian ini difasilitasi oleh Lembaga Penelitian dan Pengabdian Masyarakat Universitas Stikubank Semarang dan Hibah Bersaing Kementerian Riset Teknologi Dan Pendidikan Tinggi Republik Indonesia.

#### DAFTAR PUSTAKA

Angel, E., Shreiner, D. (2012). *Interactive Computer Graphics a Top-Down Approach with Shader-Based OpenGL 6th ed*, Addison-Wesley.

- Cisar, S.M., Radosav, D., Markoski, B., Pinter, R., Cisar, P. (2010). Computer Adaptive Testing of Student Knowledge. *Acta Polytechnica Hungarica*. Vol. 7, No. 4.
- Haryanto. (2011). Pengembangan Computerized Adaptive Testing (CAT) dengan Algoritma Logika Fuzzy. *Jurnal Penelitian dan Evaluasi Pendidikan*. Tahun 15, Nomor 1, Hal. 47-68.
- Hearn, D., Baker, M.P. (1986). *Computer Graphics*. Prentice-Hall International, USA.
- Kodituwaku, R., Wijeweera, K.R., Chamikara, M.A.P. (2012). An Efficient Line Clipping Algorithm for 3D Space. *International Journal of Advanced Research in Computer Science and Software Engineering*. Volume 2, Issue 5, May 2012.
- Lusiana, V. (2015). Proses Clipping Menggunakan Algoritma Cohen-Sutherland pada Ruang Dimensi Tiga. *Jurnal Teknologi Informasi Dinamik*, Volume 20, No.1, Januari 2015.
- Nugroho, E. (2005). *Teori dan Praktek Grafika Komputer Menggunakan Delphi dan OpenGL*. Penerbit Graha Ilmu. Yogyakarta.
- Pandey, A., Jain, S. (2013). Comparison of Various Line Clipping Algorithm for Improvement. *International Journal of Modern Engineering Research (IJMER)*. pp-69-74, Vol.3, Issue.1, Jan-Feb 2013.
- Santosa, P.I. (1996). *Grafika komputer dan antarmuka grafis: teknik penyusunan program aplikasi berbasis grafis yang profesional*. Andi Offset, Yogyakarta.
- Santoso, A. (2010). Pengembangan Computerized Adaptive Testing untuk Mengukur Hasil Belajar Mahasiswa Universitas Terbuka. *Jurnal Penelitian dan Evaluasi Pendidikan*. Tahun 14, Nomor 1, Hal. 62-83.
- Thompson, N.A., Weiss, D.J. (2011). A Framework for the Development of Computerized Adaptive Tests. *Practical Assessment, Research & Evaluation Journal*. Volume 16, Number 1, January 2011.
- Winarno. (2012). Pengembangan Computerized Adaptive Testing (CAT) Menggunakan Metode Pohon Segitiga Keputusan. *Jurnal Penelitian dan Evaluasi Pendidikan*. Tahun 16, Nomor 2, Hal. 361-379.