

BAB 6

KESIMPULAN DAN SARAN

Bagian ini berisi kesimpulan dan saran. Pada bagian ini disimpulkan hal-hal yang dilakukan pada skripsi ini. Beberapa saran dipaparkan pada bagian ini agar penelitian yang dilakukan pada skripsi ini dapat terus berkembang.

6.1 Kesimpulan

Scalable Vector Graphics (SVG) merupakan sebuah gambar dua dimensi berbasis vektor yang didefinisikan dalam format *Extensible Markup Language* (XML). Sebuah gambar SVG disimpan dalam bentuk dokumen. Dokumen SVG diawali dengan tag pembuka `<svg>` dan diakhiri dengan tag penutup `</svg>`. Pada dokumen SVG terdapat elemen-elemen yang merepresentasikan bagian gambar SVG. Setiap elemen memiliki atribut-atribut yang menentukan ukuran dan bentuk elemen tersebut.

Terdapat enam buah bentuk dasar yang dapat digunakan pada dokumen SVG untuk membentuk sebuah bagian gambar, yaitu, persegi panjang, lingkaran, elips, garis, *polyline*, dan poligon. Selain menggunakan bentuk dasar, *path* dapat digunakan untuk mendefinisikan bentuk pada dokumen SVG. Terdapat sepuluh perintah yang dapat digunakan untuk membentuk sebuah *path*, yaitu, *moveto*, *lineto*, *horizontal lineto*, *vertical lineto*, *quadratic Bezier curveto*, *shorthand quadratic Bezier curveto*, *cubic Bezier curveto*, *shorthand cubic Bezier curveto*, *elliptical arc*, dan *close path*.

Setiap elemen yang didefinisikan pada dokumen SVG diubah menjadi bagian *graph*. Algoritma untuk mengubah elemen menjadi bagian *graph* berbeda-beda sesuai dengan jenis elemen. Kurva dan busur elips perlu diubah menjadi ruas-ruas garis terlebih dahulu sebelum diubah menjadi bagian *graph*.

Luas *bounding rectangle* digunakan untuk memperkirakan ukuran suatu elemen. Bagian gambar yang terlalu kecil tidak dijadikan bagian soal permainan menghubungkan titik. Setelah gambar SVG diubah menjadi *graph*, setiap *edge* merepresentasikan ruas garis yang membentuk gambar tersebut. Setiap titik potong pada *edge* yang berpotongan dijadikan *vertex* untuk disertakan pada algoritma pencarian *euler path* Hierholzer.

Algoritma Hierholzer berfungsi untuk mencari *euler path* pada sebuah *euler graph* terhubung. *Graph* yang dihasilkan dari proses pengubahan gambar SVG belum tentu merupakan *euler graph* sehingga sebuah heuristik diimplementasikan untuk mengubah *graph* menjadi *euler graph*. Heuristik yang diimplementasikan tidak selalu menghasilkan solusi yang optimal pada setiap kasus sehingga tidak setiap soal yang dihasilkan perangkat lunak memiliki garis bantuan yang minimal.

6.2 Saran

Perangkat lunak yang dikembangkan pada skripsi ini hanya berfungsi untuk menghasilkan soal permainan menghubungkan titik berdasarkan gambar SVG masukan. Beberapa fitur seperti *print* soal dan pembuatan beberapa soal sekaligus dapat ditambahkan pada perangkat lunak. Penambahan fitur-fitur tersebut bertujuan untuk mempermudah pengguna dalam menghasilkan soal permainan menghubungkan titik.

Masukan perangkat lunak pada skripsi ini dibatasi, yaitu gambar SVG yang dapat diubah menjadi sebuah *graph* terhubung. Algoritma Hierholzer tidak berjalan dengan baik pada *graph* yang tidak terhubung. Penelitian lebih lanjut dapat dilakukan agar perangkat lunak dapat mengubah *graph* tidak terhubung menjadi soal permainan menghubungkan titik. Sebuah *graph* yang tidak terhubung terdiri atas beberapa *graph* terhubung sehingga algoritma Hierholzer dapat diterapkan pada setiap *graph* terhubung tersebut.

Terdapat dua jenis garis bantuan pada soal permainan menghubungkan titik yang dihasilkan oleh perangkat lunak yang dikembangkan pada skripsi ini. Sebuah bagian gambar yang terlalu kecil tidak diproses lebih lanjut oleh perangkat lunak sehingga bagian gambar tersebut dijadikan garis bantuan pada soal dengan cara memindahkan elemen yang merepresentasikan bagian gambar tersebut ke dokumen SVG keluaran. Berbeda dengan hal sebelumnya, garis bantuan yang dihasilkan pada proses pengubahan *graph* masukan menjadi *euler graph* direpresentasikan menggunakan sebuah elemen garis. Sebuah garis bantuan berbentuk kurva yang dihasilkan pada proses pengubahan *graph* menjadi *euler graph* akan direpresentasikan menjadi ruas-ruas garis, sedangkan sebuah kurva yang terlalu kecil akan dijadikan garis bantuan yang tetap berbentuk kurva.

Proses pembuatan garis bantuan yang dilakukan oleh perangkat lunak dapat dikembangkan sehingga setiap garis bantuan tetap menggunakan bagian gambar aslinya. Setiap garis bantuan yang berbentuk kurva tidak direpresentasikan sebagai ruas-ruas garis karena setiap garis bantuan menggunakan bagian gambar aslinya. Tidak semua garis bantuan yang dihasilkan pada proses pengubahan *graph* menjadi *euler graph* merupakan sebuah elemen utuh. Penelitian lebih lanjut dapat dilakukan untuk menangani hal tersebut.

DAFTAR REFERENSI

- [1] Eisenberg, J. D. dan Bellamy-Royds, A. (2014) *SVG Essentials: Producing Scalable Vector Graphics with XML*. " O'Reilly Media, Inc."
- [2] REC-SVG11-20110816 (211) *Scalable Vector Graphics (SVG) 1.1 (Second Edition)*. World Wide Web Consortium. Cambridge, Massachusetts, U.S.
- [3] Farin, G. (2014) *Curves and surfaces for computer-aided geometric design: a practical guide*. Elsevier.
- [4] Gross, J. L. dan Yellen, J. (2004) *Handbook of graph theory*. CRC press.
- [5] Even, S. (2011) *Graph algorithms*. Cambridge University Press.
- [6] Halim, S. dan Halim, F. (2013) *Competitive Programming 3*. Lulu Independent Publish.