The Journal of the Society for Art and Science Vol. 7 No. 4 pp. 155 - 169

Accurate and Efficient Drawing Method for Laser Projection

Purkhet Abderyim, Osama Halabi, Tadahiro Fujimoto, Norishige Chiba
Department of Computer Science, Graduate School of Engineering, Iwate University
E-mail: {purkhet, ohalabi, fujimoto, nchiba}@cis.iwate-u.ac.jp

Abstract

Laser projectors are commonly used in the entertainment industry to create vector-based graphics. In order to present a stable image to observers, the laser projector moves the beam quickly using computer-controlled mirrors. The speed is high enough so that the human eye perceives it as a persistent line of light. In contrast with other vector graphic devices, laser projectors have their own inherent problems when drawing a laser image. These problems include an unexpected laser trail when the laser is switched on and off, the inertia effect for sharp corners, flickering, and the inconsistent brightness of a visible laser line. This paper presents an accurate and efficient approach for drawing graphics with a laser projector. The first section of the paper attempts to solve the unexpected laser trail problems by testing a simple pattern where this effect occurs. In the second section, we discuss a technique to compensate the error caused by the inertia effect for different rotation angles of the mirrors. We have also proposed the use of an Eulerian graph (unicursal) to reduce flickering and an efficient drawing sequence for the created Eulerian graph.

1. Introduction

1.1. Background

Laser projectors are commonly used in the entertainment industry, especially in laser shows. The advantage of laser projectors is that they can move the laser beam quickly to create images. There are two types of laser effects. The first is the beam effect, in which the audience can view patterns created by a laser beam in air. The other is the screen effect, in which graphics are created on a screen by moving a laser spot on the screen. Applications using laser graphics are popular not only in night clubs but also in concerts and live shows. If the laser is powerful enough, graphics can be projected onto large surfaces such as dams, water surfaces, or even mountains. Lasers are also widely used during major events and championships. Laser projection technology is also being utilized in the motor vehicles industry. One of the applications used in vehicles involves a scanned laser beam that is used to display information on the wind shield as a laser beam provides good contrast, brightness, and sensitivity [1]. There is an association known as ILDA [2] (International Laser Display Association), which comprises various hobbyists, laser display manufacturers, and general users; however, their main goal is to establish a trade group. Although this group had made some efforts for the establishment of standard file formats and hardware specifications that can be used with lasers, these standards are strictly commercial in nature as they only benefit the exchange of products and services.

1.2. Laser projection technology

A laser projector is a device used to create vector graphics. Laser graphics are created onscreen by moving a laser spot on the screen. The laser projector is connected to a computer, which controls the position of the laser spot and the switching on/off of the laser, as shown in Figure 1.

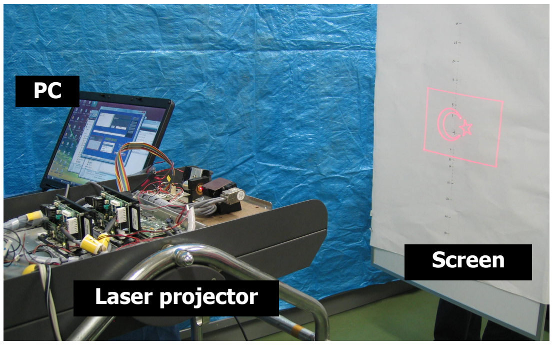


Figure 1: Laser projection system.

The working of a laser projector is given as follows: line segments can be displayed by the projector by controlling two mirrors that are connected to two galvanometers (see Figure 6). The mirrors are positioned so that the laser beam strikes the first mirror, which scans it horizontally, and then strikes the second mirror, which scans the beam vertically. Finally, the laser beam travels to the display screen.

As the laser beam moves with considerable speed, the human eye perceives it as a persistent line of light. The laser beam speed can be controlled by changing the rotating speeds of the two mirrors. The faster the beam moves, the less intense it appears to the eye. On the other hand, reducing the speed of the beam increases its brightness. If the mirrors remain stationary for a fixed time period, known as a *time delay*, when the laser is on, the laser beam appears as a spot on the screen.

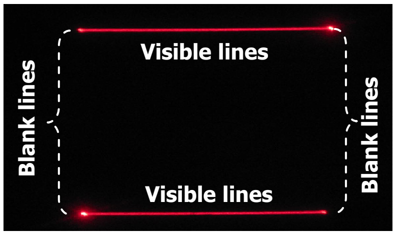


Figure 2: Two parallel visible lines need to be connected by two blank lines.

The display of a line segment is one of the most primitive operations that can be performed by a laser projector. The laser projector draws a line by moving the laser beam in the direction of the line to be created. For example, to draw a single line from point A to point B, the projector focuses on point A, turns the laser on, and then moves the laser beam directly to point B. This line is also known as a *visible line* (Figure 2). In order to create complete images with a single beam, even when images include separate disconnected visible lines, *blank lines* are required (see Figure 2). When the projector draws a blank line, the mirrors rotate in the same manner as when the visible lines are drawn but the laser beam is turned off.

In order to present an image to a human observer, laser projectors create visible lines and blank lines by rotating the mirrors continuously and turning the beam on and off. If the speed with which the image is created is not sufficiently high, the image will not be stable. This is because the laser beam cannot complete all the lines before the image starts fading. This effect, which is known as *flickering,* is usually undesired and is due to the perceptual effect in laser graphics.

Few studies have been performed on laser projection in spite of its importance in a wide variety of practical applications. Patrick et al. [3] conducted a study on laser graphics and discussed the two reasons why laser graphics lagged behind raster graphics. The reasons are as follows. First, there are no advanced techniques such as texture mapping or ray tracing in laser graphics, and second, laser companies are not well equipped to develop advanced techniques as government grants and university sponsored research are practically nonexistent. However, the main subject of their study was the introduction of the laser projection system.

Laser hobbyists and entertainment companies have developed their own equipment and software to create laser shows [4][5]. However, these developments are only for practical and commercial usage and as a result there have been no attempts to establish a formal research base for laser graphics. Hence, there is a need to explore the inherent problems in laser projection displays in order to improve their quality.

We believe that the image quality of laser projectors can be improved by optimization of vector data and the drawing techniques used by laser projectors.

1.3. Problems related to laser projection

In order to develop an accurate and efficient way of drawing graphics with a laser projector, we have to first take into account the inherent problems of laser projectors. The characteristic problems can be described as follows:

Endpoint blurring

In order to draw an entire image with a single beam, blank lines are required to connect the visible lines. When the projector draws a blank line, the mirrors rotate as if the projector was drawing visible lines but the laser beam is turned off. The software used for creating images in some laser shows assign the same time for drawing blank lines as that for drawing visible lines. Therefore, the drawing of an entire vector image is a time-consuming process.

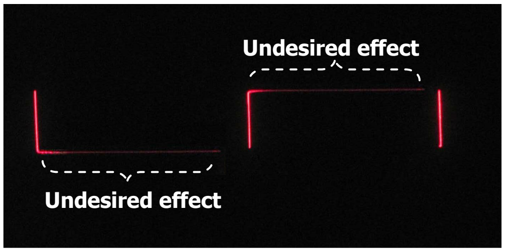


Figure 3: Undesired effects at the endpoints of the blank lines.

Technically speaking, in a single tick (time unit), the positions of the X-Y mirrors can be changed to anywhere inside the projecting range of the X-Y mirrors. In view of the fact that the blank lines are invisible and the laser projectors can change their speed when drawing a line, we can draw a blank line in a single time interval (time unit) to reduce the total drawing time. This helps reduce the flickering effect. Since a blank line can be created in a single time unit regardless of the length of the line, the total time for drawing blank lines does not have a considerable impact on the total drawing time. However, turning the laser beam on/off when the mirrors are moving can cause certain undesired effects at the terminating points of the blank lines, as shown in Figure 3. This is because of the delay between the issue of a command to stop the rotation of the mirrors and the time it takes for the mirrors to stop rotating completely. In order to solve this problem, every blank line requires a certain compensatory delay time to be added towards the end.

The main contribution of currently available software applications is that they can convert vector data for laser use. Using a software application, a suitable time delay can be determined for the two terminations or endpoints of every visible line so that the visible lines can be drawn properly. However, this makes the drawing of an entire image a time-consuming process. It should be noted that visible lines that are connected to each other do not require a time delay at the common endpoint.

Inertia effect

Another problem in laser projectors, which is not found in vector displays, is the inertia effect; this effect occurs due to the use of rotating mirrors to create line segments.

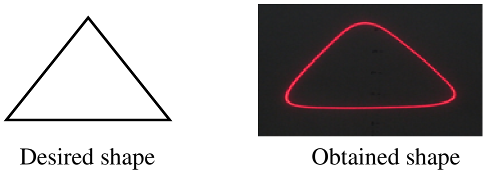


Figure 4: Change in shape of the angles due to inertia

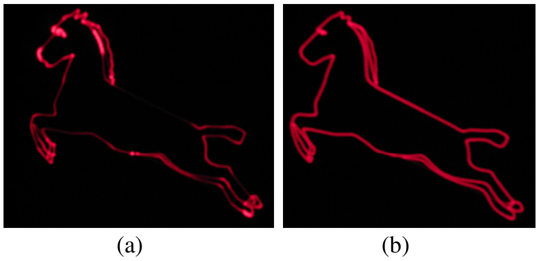
When the mirrors switch from one position to the next, sharp corners in the images are affected because the X-Y mirrors cannot follow the desired path accurately. Hence, a laser projector has a low image quality, especially if the images have sharp angles, as shown in Figure 4. It is difficult to obtain precise, sharp angles using the software applications that are currently available as these software applications do not take into account the inertia effect. A possible solution to this problem is adjusting the time delay for the angle on the basis of its value in degrees so as to obtain a high-quality laser image. Further, sharp angles can be avoided altogether by rearranging the order of drawing the lines.

Flickering

Flickering is the most evident problem when drawing a laser image. As the beam moves, to the human eye it appears that a line is being drawn. However, after a few moments, the effect starts to disappear. Quick redrawing makes the line steady. If the speed of redrawing is not high enough, it causes flickering. The image to be drawn is decomposed into a finite set of visible lines that are drawn only once per drawing loop. In order to reduce the flickering effect, we have to reduce the total drawing time to make the drawing process as quick as possible. Since the drawing time of a visible line cannot be reduced, the other option is to reduce the drawing time of blank lines. Consequently, in order to achieve flickerless images, the number of blank lines number should be kept as small as possible. However, existing laser-show software applications are not capable of minimizing the number of blank lines. Hence, the vector data is not graphically optimized before projection.

Variation in brightness

We know that the faster a laser beam moves, the less intense it looks to the eye. On the other hand, reducing the speed of the beam increases its brightness. In other words, if two lines with different lengths are drawn in the same time interval, the short one looks brighter than the long one, as can be seen from Figure 5(a).



(a) (b)

Figure 5: Image before and after application of techniques for constant brightness.

Constant brightness can be achieved either by moving the laser beam with a constant velocity for all vectors, by adjusting the beam intensity for different vectors, or by having a laser beam with constant intensity and variable moving speed. There is another method to achieve constant brightness in laser graphics. In this method, all segments are broken down into vectors of similar size and are then displayed at constant speed [6]; see Figure 5(b).

1.4. Related technologies

In order to develop an efficient drawing method for graphical devices, we considered some related studies that solved similar problems. A vector graphics device known as the *pen plotter* draws a figure by moving a pen across the surface of a piece of paper. It draws lines with the “pen down” movements on the paper and connected those lines with “pan up “movements. The main problem in the case of this device is how to reduce the total plotting time.

M. Iri et al. [7] developed a practical model of the optimal movements of a pen plotter by minimizing the plotting time and reducing the total length of blank (pen up) lines. They took into account the fact that in pen plotter the number of blank lines does not affect the total drawing time and the quality of plotted image does not need to be considered (Table 1).

Laser plotters are used in the manufacturing industry to create trademarks and decorations on metallic surfaces using a laser light source. G. Ghiani et al. [8] modelled and solved the routing problem of the laser plotter beam by treating the problem as a constrained *arc routing problem*. In order to guarantee a high-quality pattern and to reduce maintenance costs, they took into account the number of spots that are created on the surface each time a new line is drawn. This number of spots is equal to the number of blank lines or ‘non-drawing’ moves. However, in the case of the laser plotter, it is necessary to consider the drawing order and the turning angles as well.

Unlike pen plotters and laser plotters, in laser projectors, the laser beam source is fixed. Laser images are displayed by two mirrors that reflect the laser beam (see Figure 6). The laser beam is first reflected by the first mirror, which controls the horizontal direction, and is then reflected by the other mirror, which controls the vertical direction; after this, it finally travels to the display screen. The laser beam speed is decided by the rotation speed of the two mirrors. When the projector draws a blank line, the mirrors rotate with the laser beam turned off; this is the main differentiating point between the projectors and the pen plotter and laser plotter. Laser projectors can draw blank lines in a single time unit regardless of the length of the blank line. This feature is their biggest advantage as compared to pen plotters and laser plotters when it comes to reducing the overall drawing time. Table 1 presents some conclusions based on the above discussions.

Table 1 : The laser projection problem

	Pen plotter	Laser plotter	Laser projector
Total drawing time	●	●	●
Number of blank lines		●	●
Blank lines total distance	●	●	
Turning angles			●
Lines sequence			●

When the mirrors try to switch quickly from one position to the next, an effect known as the inertia effect affects sharp corners; this is because the mirrors cannot accurately follow the desired design path. Although this problem is not applicable for the pen plotter and the laser plotter, it is necessary to consider this problem when studying laser projectors in order to guarantee high-quality laser image projection (Table 1).

In contrast to pen plotters and laser plotters, there are several additional aspects that need to be taken into account for laser projectors. In Table 1, the black dots indicate the problems that need to be considered. The ‘Laser projector’ column in Table 1 shows that the total length of all the blank lines does not affect the total drawing time because the blank lines can be drawn in a single time unit, irrespective of their lengths. However, it is necessary to consider the turning angles and the order in which the lines are drawn in the laser projector to obtain an accurate laser image.

Hence, in order to develop an accurate and efficient method to draw graphics with a laser projector, we need to take into account its characteristic problems, as given in Table 1.

1.5. Approach adopted by our method

In this paper, we present an efficient method to draw graphics with a laser projector. In order to achieve this, we perform several experiments, described as follows. In the first experiment, we attempt to solve the problem related to the creation of blank lines by testing a simple pattern where endpoint blurring occurs. Following this, we attempt to compensate the error caused by the inertia effect for different rotation angles. Finally, in order to reduce the number of blank lines and avoid sharp angles, we create an Eulerian graph and propose an efficient drawing sequence using graph theory.

2. Experimental laser projector

The most common technique used by laser projectors to control the laser beam is the use of rotating mirrors with corresponding galvanometers. We assembled a laser projector in our laboratory using separate parts, as shown in Figure 6. The laser projector is connected to a computer through a serial port. The basic components of the laser projection system are as follows:

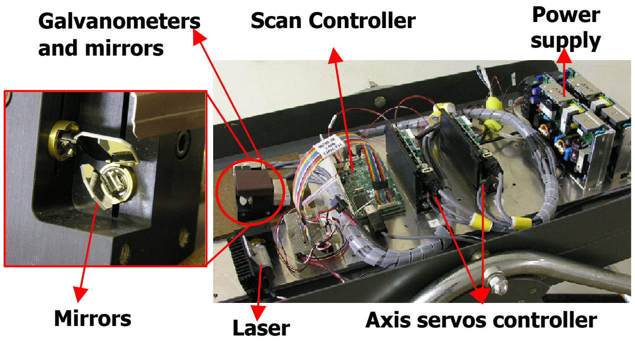


Figure 6: Experimental laser projector

Scan controller: It is responsible for communicating with the PC and the axis servos, and it also has some memory that can store programs before sending them to the axis servos.
Axis servos controller: It converts digital signals from the main controller into analogue currents for the galvanometers and also communicates feedback on the position to the scan controller.
Power: It is used to supply power to the laser projector.
Galvanometers and mirrors: The galvanometers turn the mirrors to control the x and y directions of the laser beam.
Laser: It produces laser beams of a given colour and intensity.

The projector is operated by sending instructions to the controllers. Instructions are sent to the servo controllers that rotate the X and Y mirrors by changing the current on the galvanometers. It should be noted that the laser beam remains static, and it is the reflection of the beam on these two fast-moving mirrors that creates the illusion of lines and shapes on the projection screen.

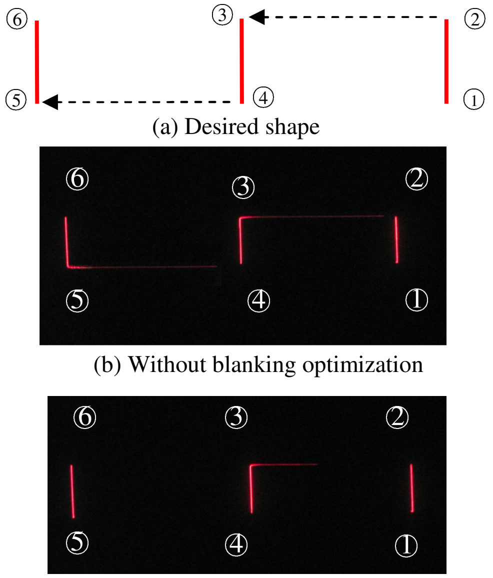
The projector used in our experiments was assembled using the following components:
Scan Controller: *GSI Lumonics SC2000*
Servo Controllers: *GSI Lumonics MiniSAX (2 units)*
Galvanometers: *GSI Lumonics VM-500*
Laser: *Coherent MVP Control Diode Laser* Module. *Wavelength: 670 nm ± 10 nm (**Red**).* Power: *0.95 mW*

3. Laser Graphics Optimization

In order to obtain more precise laser images and to reduce the flickering effect, the inertia of the mirrors with respect to sharp angle corners needs to be evaluated and the number of blank lines should be made as small as possible. In this section, we perform some simple experiments to emphasize the problem, and depending on the observed characteristics, discuss suitable solutions to optimize the final image.

3.1. Blanking optimization

In this section, we try to resolve the endpoint blurring problem that we discussed in section 1.3. When the projector draws a blank line, the mirrors rotate with the laser beam turned off. The speed of rotation of the mirrors for blank lines can be increased to decrease the flickering effect. Theoretically, any blank line can be covered in a single time unit, regardless of its length.



(a) Desired shape

(b) Without blanking optimization

(c) After application of two blanking methods

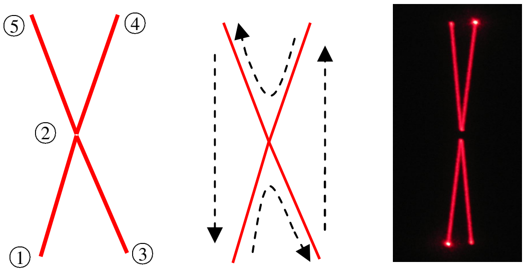
Figure 7: (b) Blank lines drawn in a single time unit (c) Blank lines drawn between②③ and ④⑤ using two different approaches.

This possible acceleration of mirror speed is the main advantage of laser projectors over pen plotters and laser plotters when it comes to reducing the overall drawing time; however, when only a single time unit is used to draw a blank line, there is an undesired effect on the image, as shown in Figure 7(b). It can be noticed that when the beam jumps from position ② to position ③ in a single time unit, the laser projector leaves a dim laser trail. This happens because the command ‘turn on beam’ takes less time to execute than the actual ‘jump’ command. There are two methods to solve this problem. The first method is to increase the time for drawing a blank line, i.e. to slow down the drawing speed for blank lines. As a result, the laser trail becomes shorter.

Therefore, it follows that in order to completely delete the undesired laser trail on the blank line; we need to spend more time to draw it. However, by adopting the above-mentioned approach, the main advantage of the laser projector is lost. The second method to solve this problem is to spend as little as time as possible on blank lines and provide a certain time delay to take into account the jump to the destination point before turning on the laser. This method is more time efficient and shows better results. In Figure 7(c), two blank lines of the same length ②③ and④⑤ are drawn using the same time delay but with two different methods. The fist method is applied to the blank line between ② and③; however, it can be observed that the laser trail has not been removed completely. The second method is applied to the blank line between ④ and ⑤. In this case, the undesired laser trail has been completely removed.

3.2. Angle optimization

Another problem that we need to consider is the inertia effect when a laser projector draws sharp corners as the mirrors cannot always follow the desired direction.



(a) Desired shape (b) Drawing path (c) Obtained shape

Figure 8: Undesired effects caused by inertia

The desired shape in Figure 8(a) is drawn by laser projector as the drawing order showing in Figure 8(b). The results show that pairs of vertical angles have not been created accurately in the obtained image. The vertex ② appears at two different positions when two angles ∠①②③ and ∠④②⑤ are drawn by following the path shown in Figure 8 (b).

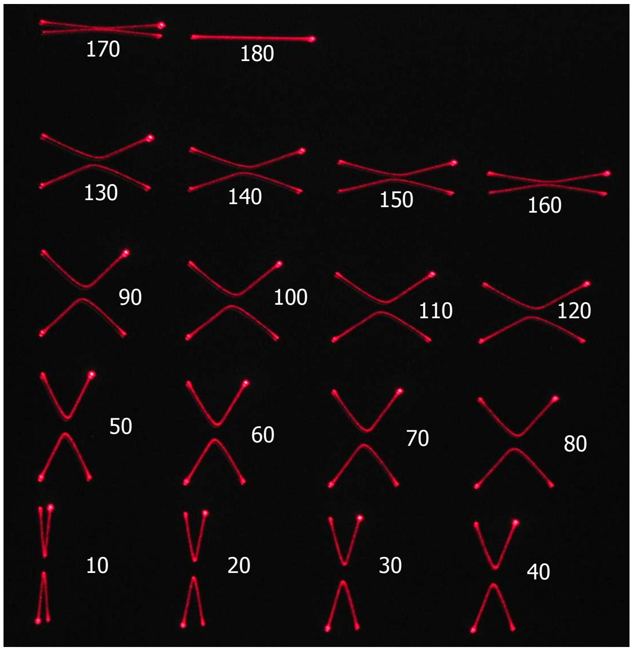


Figure 9: Undesired inertia effect in different angles

Figure 9 has the same shape as that in Figure 8(a) but with different angles ranging from 10° to 180°. In Figure 9, we can observe that the inertia effect depends on the angle between two lines. Moreover, the problem is more evident in acute rather than obtuse angles. In acute angles, the angles appear disconnected due to the inertia effect. Moreover, if the angles are larger than 40°, they appear curved. The reason for both the separation of the angles and the curved effect is that the mirror changes its direction too fast at corners. This problem can be solved by reducing the scanning speed of the mirrors. However, if the scanning speed is too slow, flickering occurs. A more efficient way to avoid the inertia effect is to add an extra time delay at corners in order to help the mirrors follow the desired path more accurately. However, redundant time delay will cause unwanted bright dots on corner. This raises the question of how much time delay is required.

Figure 10 shows the results after the addition of delays during the drawing of angle vertices. It can be observed that the pairs of angles within the range of 90° and 180° are connected together but the pairs of angles within the range of 10°and 80° still have some gaps. On the basis of the results obtained on observation of Figure 10, we can conclude that more delays are required when drawing corners in order to reduce the gaps between sharp angles. The addition of more time delays for every corner results in all the angles being connected. In fact, as the results in Figure 9 show, angles of 170° and 180° do not actually need extra delays. Moreover, the addition of extra delays for large angles makes the connection points brighter.

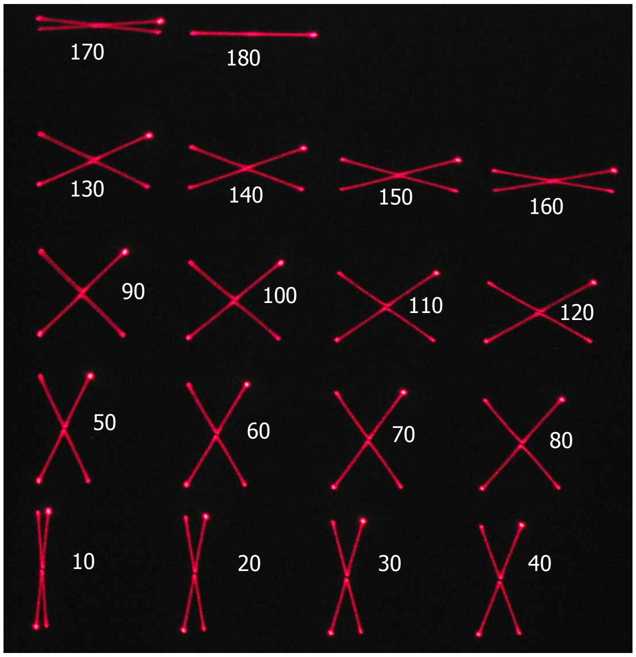


Figure 10: After the addition of extra delay for drawing the corners

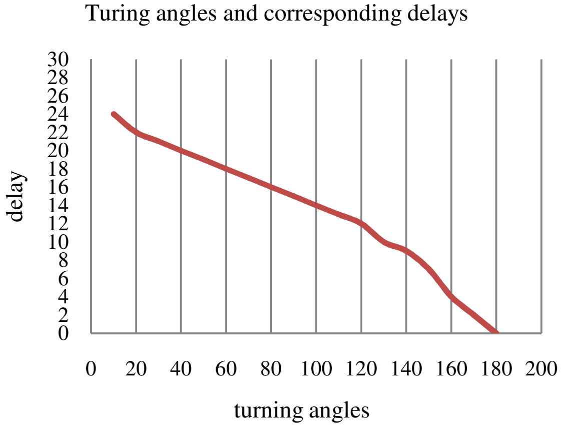


Figure 11: Relation between demanded delays and angles

In order to obtain a suitable time delay for every angle depending on its degree, the shape in Figure 8 was redrawn with different angle sizes. We increased the time delay for each angle gradually until the two vertical angles were connected. The time delay when they were finally connected together was set as the suitable delay for this angle size. The connection between two angles was judged by five different observers. On the basis of the experimental results, (Figure 11) we know that the relation between the angles and the suitable delay is approximately a linear relation.

The scan rate also influences the amount of delay. Since the faster the mirror moves, the greater the delay required for the corner. Because of the different brightness aspects, it is not convenient to change the scanning speed while drawing connected visible lines; therefore, we propose suitable time delay for every different turning corner. The appropriate delay times and the turning angles are linearly related as follows:

$$\mathrm{delay} = \mathrm{max\_dealy}\left(1 - \frac{\mathrm{between\_angles}}{180.0}\right) \qquad (1)$$

On the basis of equation (1), we can determine the appropriate delay for every angle.

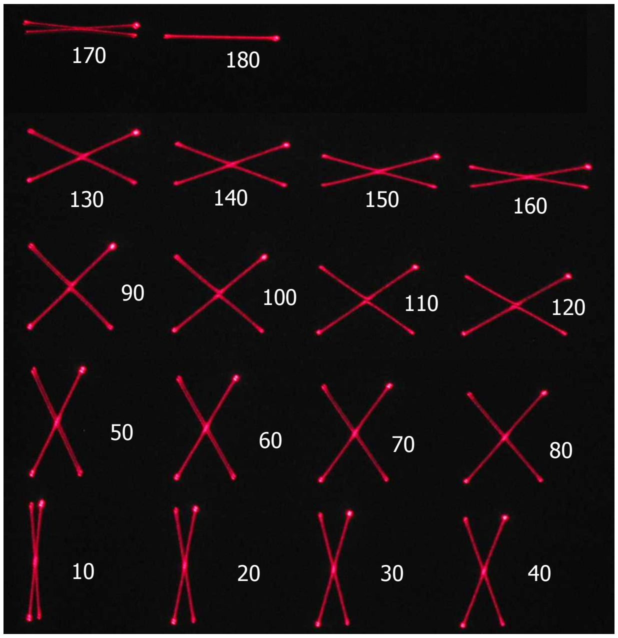


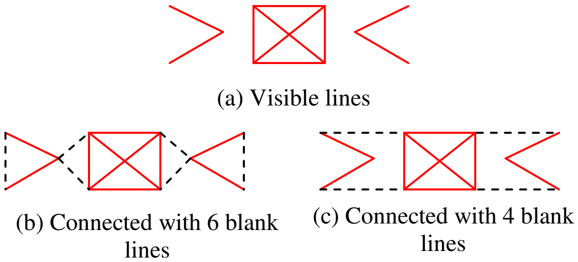
Figure 12: After addition of corresponding delays for the corners

Figure 12 shows the corresponding delay for every angle after optimization. Thus, all the angles can be connected and the drawing time is also reduced.

3.3. Graph optimization

In this section, we describe a method to determine the minimum number of blank lines so that a given image can be redrawn as fast as possible.

In order to reduce flickering and to draw an image efficiently, it is necessary to determine the minimum possible time to draw all the visible and blank lines in the image. The drawing time required to draw visible lines in the image is constant, and from 3.1, we know that the total length of blank lines does not affect the overall drawing time, whereas the number of blank lines does. Therefore, we need to find the minimum necessary number of blank lines such that the set of visible lines and blank lines can be drawn unicursally. Let us consider Figure 13 as an example. The set of visible lines in (a) can be unicursally drawn with 6 blank lines in (b) and with 4 blank lines in (c). Obviously, (c) is more efficient than (b) because (c) has less blank lines than (b), and the length of the blank lines does not affect the drawing time.


(a) Visible lines

(b) Connected with 6 blank lines

(c) Connected with 4 blank lines

Figure13: Addition of blank lines

The problem of determining the number of blank lines required for a given set of visible lines can be solved using graph theory. The visible and blank line segments can be easily related to graphical elements by equating a segment to an *edge* and the terminating points of the line to *vertices*. In graph theory, we know that if every vertex in a connected graph is of an even-degree, then the graph has an *Eulerian circuit*. An Eulerian circuit is a sequence comprising the edges of a connected graph G such that every edge in G is traversed only once; however, each vertex in G can be visited more than once. In other words, the circuit provides a drawing order to traverse all the edges in G unicursally by starting and ending at the same vertex. If a graph has an Eulerian circuit, the graph is also known as an Eulerian graph. On the basis of the above information, we consider a method to determine the minimum number of blank lines for a given set of visible lines in a given image such that the set of the visible and blank lines becomes a connected Eulerian graph. This allows us to achieve an efficient drawing method where every visible line and blank line is drawn only once per drawing loop.

In order to convert an original non-Eulerian graph, which is defined by only visible lines, into an Eulerian graph, we use the following algorithm. First, we find the *connected components* in the original graph using a depth-first search algorithm. If the original graph consists of disconnected parts, it is separated into connected components. In Figure 14, the given graph has 3 connected components (a), (b), and (c).

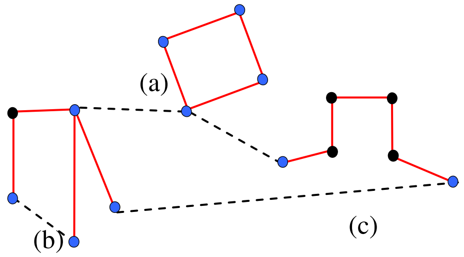


Figure 14: Example of connected components. (a) Eulerian component; (b) and (c) non-Eulerian component; the blue vertices indicate free vertices.

Then, we divide the connected components on the basis of the values of the degree of the vertices into two types: *Eulerian components* and *non-Eulerian components*. If every vertex of a connected component has even-degree, then it is an Eulerian component. An Eulerian component has an Eulerian circuit and can be drawn unicursally by starting and ending at the same vertex. In Figure 14, the connected component (a) is an Eulerian component. If a connected component has at least one vertex of odd-degree, it is a non-Eulerian component; in this case, actually, the number of odd-degree vertices is even. The non-Eulerian components in Figure 14 are (b) and (c).

Next, we define *free vertices*, which connect to each other to create blank lines. The definition of a free vertex varies depending on the component type. In an Eulerian component, every vertex is a free vertex. In a non-Eulerian component, every odd-degree vertex is a free vertex. In Figure 14, the free vertices are coloured in blue.

The final step is to create blank lines between the free vertices. In order to make the resulting graph an Eulerian graph with a minimum number of blank lines, we use the following rule: when a connected component is an Eulerian component, as in Figure 14 (a), any of its vertices can be used to connect it to other components. However, once the vertex is selected, the remaining vertices cannot be used to connect to other components. Moreover, the chosen vertex has to be connected to one or two other connected components by two blank lines. In the case of a non-Eulerian component, as in Figures

14 (b) and (c), the number of odd-degree vertices is even, i.e. $2m$. From among the $2m$ odd-degree vertices, two vertices are chosen to connect to other components. The remaining $2m - 2$ odd-degree vertices are paired to make $m - 1$ pairs. Then, the two odd-degree vertices in each pair are connected to each other, as shown in Figure 14 (b).

When the above algorithm is applied, the number of blank lines reduces to the minimum necessary number, which can be calculated by using the following equation:

$$r = p + n/2 \qquad (2)$$

Where p is the number of Eulerian components, and n is the number of odd-degree vertices of all non-Eulerian components. In Figure 14, there is only one Eulerian component, and p is equal to one. Moreover, the number of odd-degree vertices n is 6. Therefore, the minimum number of blank lines is r is 4.

3.4. Drawing Order Optimization

A laser image is generated by redrawing the set of visible lines and blank lines only once in every drawing loop. In section 3.3, we converted the given visible lines into an Eulerian graph G with blank lines. This implies that there is at least one unicursal drawing order, i.e. a drawing order that traverses each line (edge) in the graph G exactly once per drawing loop. In general, G can have more than one unicursal drawing order, which means that G can have more than one Eulerian circuit.

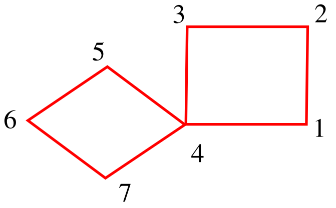
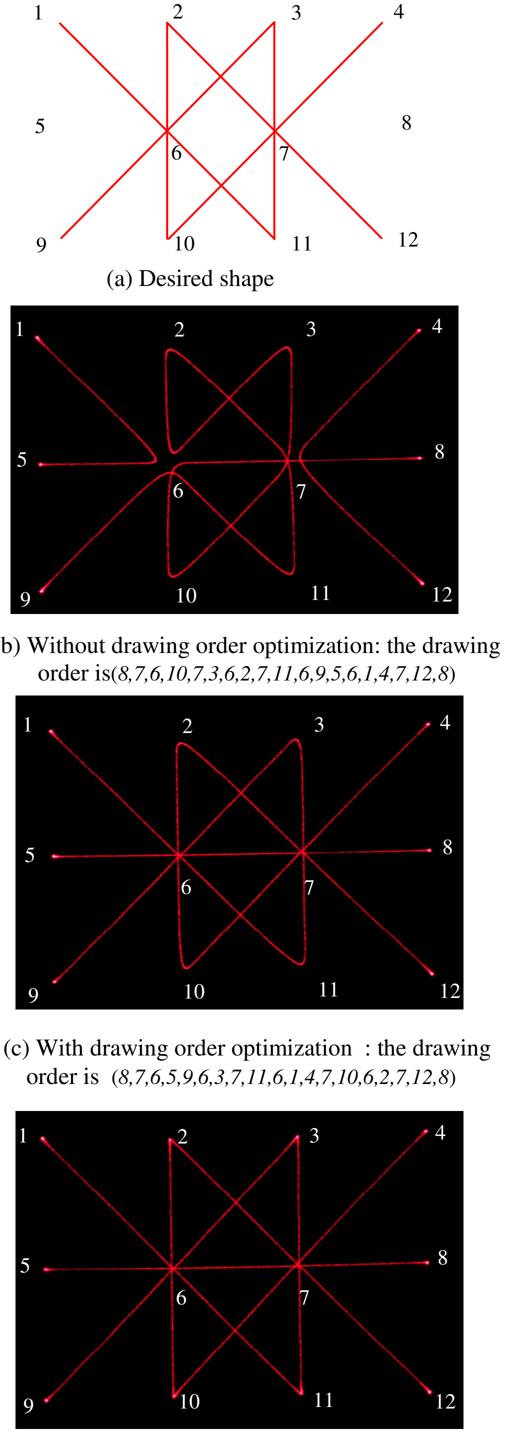


Figure15: Eulerian graph

For example, the Eulerian graph in Figure 15 an Eulerian circuit as follows. First, the circuit starts from vertex 1 and moves to vertex 4. Once it reaches vertex 4, the next vertex is chosen from among the three vertices 3, 5, and 7. However, vertex 3 is not appropriate for creating a unicursal drawing of the graph. Hence, we have to choose vertex 5 or 7 after vertex 4. Choosing vertex 5 or 7 allows us to create a unicursal drawing that first returns to vertex 4 and finally returns to vertex 1. Generally, there can be various types of Eulerian circuits in an Eulerian graph. In laser graphics, it is important to select the most appropriate Eulerian circuit because the sequences to draw the visible lines also affect the image quality, especially if the lines form sharp angles.



(a) Desired shape

(b) Without drawing order optimization: the drawing order is(*8,7,6,10,7,3,6,2,7,11,6,9,5,6,1,4,7,12,8*)

(c) With drawing order optimization : the drawing order is (*8,7,6,5,9,6,3,7,11,6,1,4,7,10,6,2,7,12,8*)

(d) With both drawing and angle optimization

Figure 16: Effect of drawing order.

In Figure 16, we observe that there are various ways to draw the predesigned shape (a), depending on the order in which the edges are drawn. Image (b) is drawn without any consideration of the angles between two connected edges. The drawing order is *8 ,7 ,6 ,10 ,7 ,3 ,6 ,2 ,7 ,11 ,6 ,9 , blank 5 ,6 ,1 , blank 4 ,7 ,12 , blank 8*; there are three blank lines. The results show that there are some unconnected and curved effects appeared on the vertices 2, 3, 6, 7, 10, and 11. We can remove this undesired effect by giving a higher priority to straight angles, if there are any. In image (c), the same shape is drawn following a different order of drawing the vertices. The drawing order is *8, 7, 6, 5, blank 9, 6, 3, 7, 11, 6, 1, blank 4, 7, 10, 6, 2, 7, 12 blank 8*; here too, there are three blank lines. In the images (b) and (c), the number of blank lines and the drawing speeds of the visible lines are the same; the only difference is between the drawing orders. Consequently, the resulting image is different. Image (c) illustrates that the image quality of vertices 6 and 7 improves significantly. However, vertices 2, 3, 10, and 11 still appear curved. Since these vertices do not comprise straight angles, the curved effect can be improved by angle optimization, as described in section 3.2. Image (d) is the resultant image after both drawing and angle optimization.

Our algorithm for constructing an Eulerian circuit is based on Hierholzer's algorithm [9]. The algorithm that we have implemented in our experiments is as follows: In the algorithm, the *drawing order T* indicates an Eulerian circuit defined on a subgraph G′ of an objective Eulerian graph G. T is the sequence of traversed vertices on G′ in the order of their traversal, in which the first vertex is the same as the last one. Further, *E*(*T*) indicates the set of all the edges traversed in T. We have outlined the steps below:

Step 1. Choose a vertex ν_0 on an objective Eulerian graph G. Then, construct a drawing order T_0 that starts and ends at the vertex ν_0 by traversing edges not yet traversed, one by one. This makes $T_0 = (\nu_0, \ldots, \nu_0)$. During the traversal, if there are more than two edges that have not been traversed and that are connected to a common vertex, choose the edge close to the straight angle. Set i = 0.

Step 2. If $E(T_i) = E(G)$, then go to step 7.

Step 3. In drawing order T_i, choose an arbitrary vertex ν_{i+1} that has a connecting adjacent edge that has not been traversed.

Step 4. Construct a drawing order T'_i starting and ending at ν_{i+1} such that any edge in $E(T'_i)$ is not in $E(T_i)$.

Step 5. Construct a drawing order T_{i+1} by merging $T_i = (\nu_0, \ldots, \nu_{i+1}, \ldots, \nu_0)$ and $T'_i = (\nu_{i+1}, \ldots, \nu_{i+1})$ such that T'_i is inserted into ν_{i+1} in T_i; in other words, ν_{i+1} in T_i is replaced with T'_i. This yields $T_{i+1} = (\nu_0, \ldots, \nu_{i+1}, \ldots, \nu_{i+1}, \ldots, \nu_0)$. Set i = i +1.

Step 6. Go to step 2.

Step 7. T_i is the final drawing order of the graph G.

This algorithm can also be applied to Figure 15 as follows:

Step 1: If $\nu_0 = 1$, then T_0 can be (1, 4, 3, 2, 1).

Step 2: $E(T_0)$ is not equal to E(G).

Step 3: Set $\nu_1 = 4$ because vertex 4 has connecting edges not traversed.

Step 4: Select T'_0 from (4, 5, 6, 7, 4).

Step 5: Merge T_0 and T'_0 to yield T_1 (1, 4, 5, 6, 7, 4, 3, 2, 1).

Step 6: Go to step 2.

Step 2: $E(T_1) = E(G)$. Go to step 7.

Step 7: T_1 is a final drawing order of Figure 15

3.5. The drawing method for the laser projector

Laser projectors are devices used to create vector graphics, and they use visible lines and blank lines to create laser images. The collection of lines can be seen as a graph by equating a segment to an edge and the beginning and ending points of the segment to vertices. Hence, we can use graph theory techniques to optimize vector data. For this, we first convert a given graph (visible lines) to an Eulerian graph (unicursal) with a minimum number of blank lines. Second, we determine an efficient drawing order for the Eulerian graph. Third, we compensate the error caused by the laser projector with some time delay to obtain an accurate laser image. Our algorithm for laser graphics is as follows:

(1) Input the vector data.
(2) Create a graph from the vector data.
(3) Find the connected components using the depth-first search algorithm.
(4) Determine the free vertices for every connected component.
(5) Connect the connected component with free vertices using blank lines (section 3.3).
(6) Construct the Eulerian circuit (section 3.4).
(7) Add an extra time delay for blank lines (section 3.1).
(8) Add a corresponding time delay for corners (section 3.2).
(9) Adjust the brightness to make it consistent.

4. Experimental Results

In order to test our algorithm, we applied our drawing method to a set of vectors. We evaluated the results by comparing the total drawing time, the undesired inertia effect at the corners, and the turning on/off of the laser beam.

Vector data were obtained from the raster image of the Turkish flag, see Figure 18 (a), and the data was displayed using our laser projector without any optimization process. The results are shown in Figure 18 (b). Because of the blanking lines and the mirrors' inertia, the projected image appeared highly distorted; Moreover, the process to draw the flag was time consuming (Figure 17). In order to obtain a high-quality accurate laser image and decrease the flickering, we reduced the number of times the laser was turned on/off as well as the total drawing time using the graphical optimization process described in section 3.3. The visible lines were connected to each other using a minimum number of blank lines. The image, Figure 18(c), quality was improved on the contrary it can be noticed that undesired laser trails still exist near the blanking lines. Next, we applied blanking optimization (described in section 3.1) to the blank lines. This increases the time of drawing slightly (Figure 17). However, it ensures that the laser trail is avoided (Figure 18(d)). Moreover, due to the inertia effect, the angles of the star were still curved. Adding the same delay on every angle, which means that a constant value is given as delay in equation (1) for every angle, improves the star image as shown in Figure 18(e), even though unwanted bright dots appeared on the curve of the crescent moon and the total drawing time increased (Figure 17).Subsequently, by applying the angle optimization process, the delays for every angle were set appropriately and we could see significant improvement in the angles of the star and the curve of the crescent moon. In the second example, we applied our algorithm to large vector data. Figure 19 shows that our algorithm improves the image quality of large vector data and reduces the total drawing time.

From Figure 17, the following observations can be concluded. In the case of the Turkish flag image without optimization, there was significant flickering (11.9 FPS). However, after the number of blank lines was reduced the drawing time and the flickering decreased (23.4 FPS). The application of blanking optimization results in a slight increase in the time of drawing; however, it can improve the quality of the graphics (21 FPS). Uniform delays on every angle can improve the drawing of angles but it will increase the drawing time and reduce the overall image quality (12.2 FPS). A suitable delay improves the image quality and reduces flickering (18.5 FPS). In the case of the "Virtual Man" image, we can improve the drawing time and the quality of the image; Nevertheless, we are still constrained by the laser projector's speed limit as going far beyond the limit would result in flickering, (Figure 17).

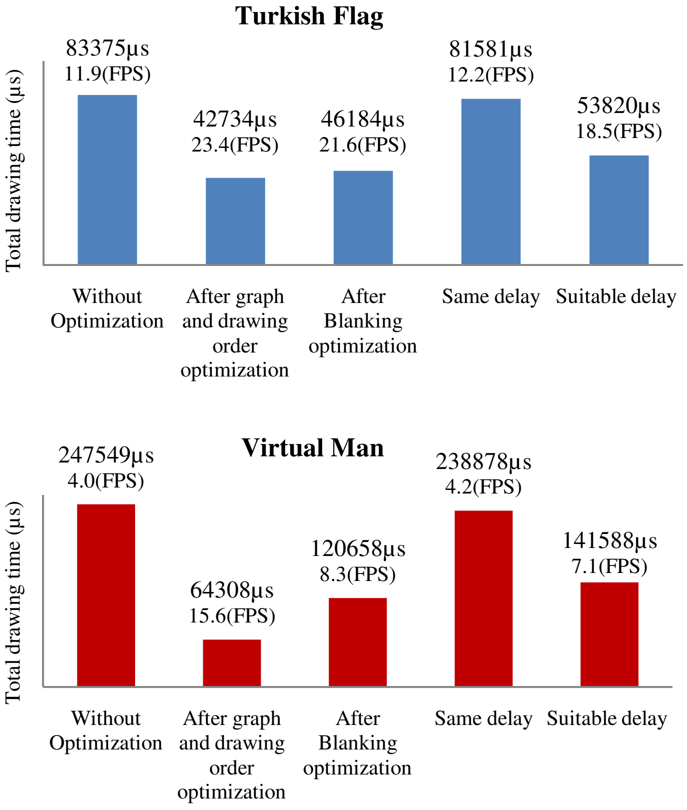


Figure 17: Drawing times of the Turkish flag and Virtual Man

5. Conclusion and Future Works

A laser projector is a device used to create vector graphics. It differs from other vector-based devices in the way it displays graphics. In this paper, we present an accurate and efficient way for drawing graphics with a laser projector. In the initial sections of the paper, we have demonstrated using experiments, the problems related to laser graphics such as flickering, endpoint blurring, and the inertia effect at sharp angles. Following this, we provided solutions to the adverse effect caused by the turning on/off of the laser beam and the inertia effect associated with different rotation angles. Finally, in order to reduce the number of blank lines and avoid sharp angles we proposed the development of a suitable Eulerian graph and an efficient drawing sequence based on graph theory.

However, it must be pointed out that in laser projection, as the data becomes more complex, the scanning speed of a projector might not be fast enough to display a stable image, even if the data is optimized. In a future study, we intend to investigate this problem and find solutions on how to draw more complex data with laser projectors. Hence, the next step in our research includes the investigation of multiple laser projectors.

References

[1] Microvision, Inc., Vehicle Displays, http://www.microvision.com/vehicle_displays/index.html, Accessed 25 June 2008.

[2] The International Laser Display Association, Interactive Laser Show, http://www.laserist.org/Laserist/index.htm, Accessed 25 Jun 2008.

[3] P. Murphy and W. R. Benner, Jr., 'Computer Graphics for Scanned Laser Displays', SMPTE Journal, Dec. 1993, pp 1125–1133.

[4] Pangolin Laser Systems, Inc., http://www.pangolin.com, Accessed 25 June 2008.

[5] PHOENIX Showcontroller GmbH & Co. KG, Inc., http://www.phoenix-lasersoftware.de, Accessed 25 June 2008.

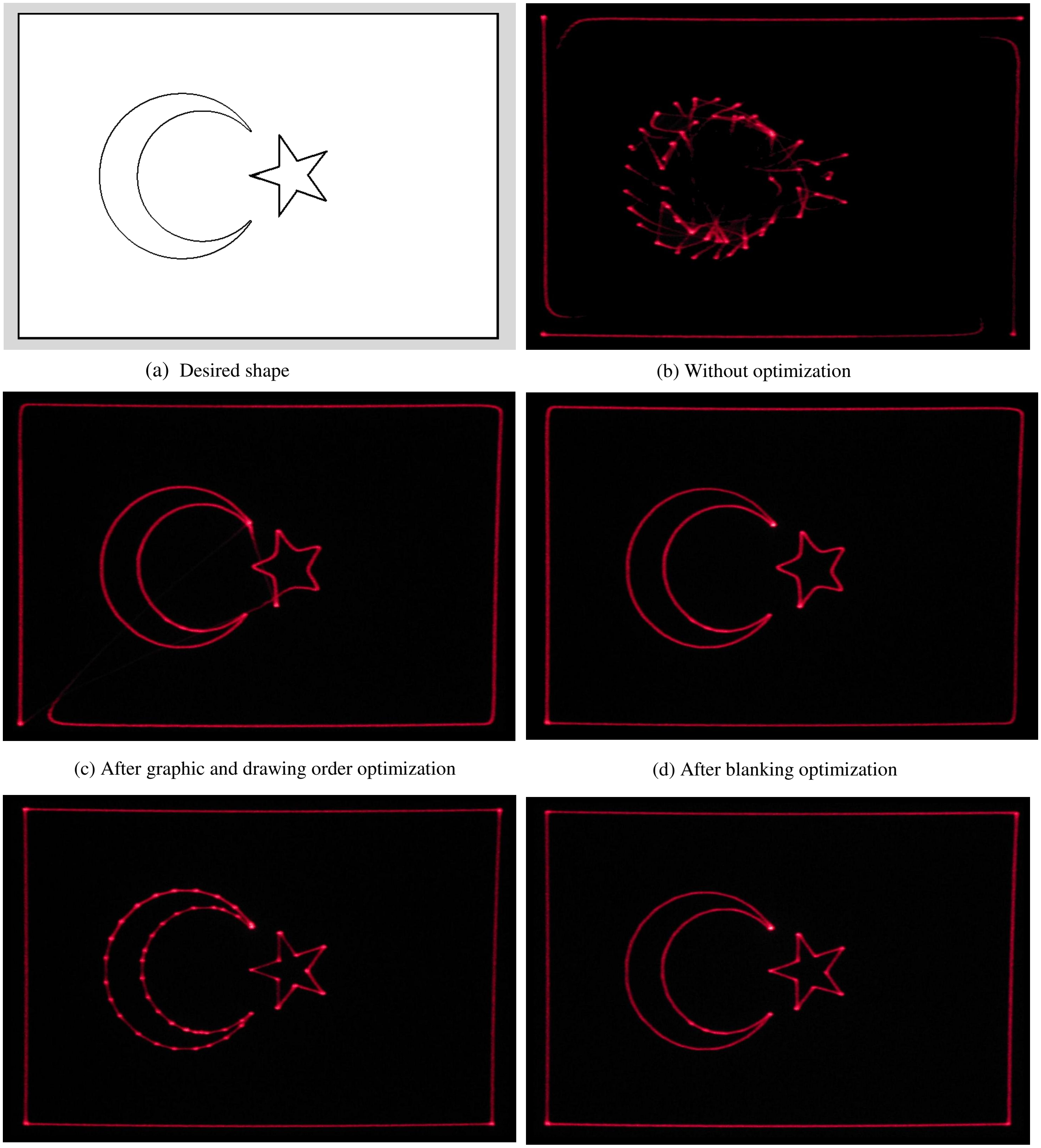
[6] P. Abderyim, M. Francisco, O. Halabi, and N. Chiba, 'Morphing-based Vectorizaed Candle Animation for Laser Graphics' IWAIT 2007, CD-ROM.

[7] M. Iri, K. Murota and S. Matsui, 'An approximate solution for the problem of optimizing the plotter pen movement' System Modeling and Optimization, Lecture Notes in Control and Information Sciences, Springer-Verlag: Berlin, 1982,pp 572–580.

[8] G. Ghiani and G. Improta, 'The laser-plotter beam routing problem' Journal of the Operational Research Society, 2001, 52, pp 945–951.

[9] M. Dror, 'Arc Routing: Theory, Solutions, and Applications', Kluwer Academic Publishers, USA, 2000.

[10] P. Abderyim, O. Halabi, T. Fujimoto, and N.Chiba, 'An Efficient Drawing Method for Laser Projectors', Nicograph International 2008, CD-ROM.



(a) Desired shape

(b) Without optimization

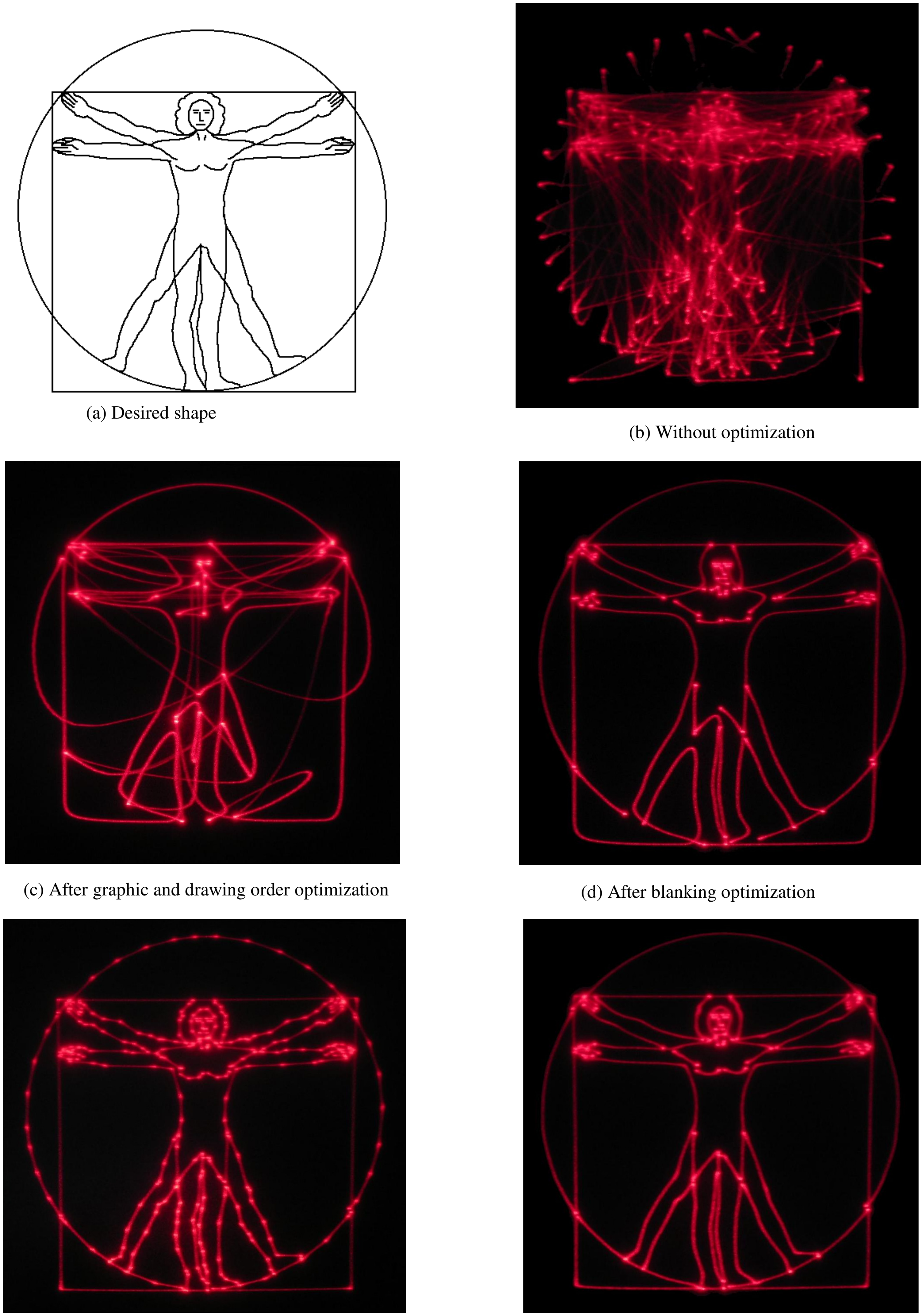
(c) After graphic and drawing order optimization

(d) After blanking optimization

(e) After addition of the same delay for all angles

(f) After angle optimization

Figure 18: Projected image of the Turkish flag



(a) Desired shape

(b) Without optimization

(c) After graphic and drawing order optimization

(d) After blanking optimization

(e) After addition of the same delay for all angles

(f) After angle optimization

Figure19: Projected image of Virtual Man

Authors' biographies:

PURKHET ABDERYIM is currently a Ph.D. candidate in Computer Science at Iwate University. His research interests include computer graphics, computer animation and laser graphics. He received the B.E in computer Science from Xinjiang University, china and an ME in computer science from Iwate University in 1999 and 2006, respectively. He worked at Xinjiang University from 1999 to 2003.

OSAMA HALABI is currently an assistant professor in the Department of Computer and Information Sciences at Iwate University. His research interests include human computer interaction, haptic interface, virtual reality, laser graphics, and computer graphics. He received a B.Sc., in electronic engineering from Damascus University, M.Sc. in computer science from Shanghai University, and Ph.D. in Information Systems from Japan Advanced Institute of Science and Technology (JAIST) in 1992, 1998, and 2001, respectively. He was an assistant professor at Japan Advanced Institute of Science and Technology as a Fujitsu Endowed Chair from 2001 to 2003, a researcher in Virtual Systems Laboratory at Gifu University from 2003 to 2006. He is a member of VRSJ Japan, and ACM Computer Graphics.

TADAHIRO FUJIMOTO is currently an associate professor in the Department of Computer and Information Sciences at Iwate University. His research interests include computer graphics, geometric model, fractal theory, laser graphics and mathematics for shape description in general. He received a BE in electrical engineering, and an ME and Ph.D. in computer science from Keio University in 1990, 1992, and 2000, respectively. He worked at Mitsubishi Research Institute from 1992 to 1995. He was a research associate in the Department of Computer and Information Sciences at Iwate University from 1999 to 2002, and a lecturer from 2002 to 2005. He is a member of SAS Japan, IEICE Japan, IPS of Japan, IEEE and ACM.

NORISHIGE CHIBA is currently a professor in the Department of Computer and Information Sciences at Iwate University. His research interests include computer graphics, algorithm theory and science on form. He received a BE in electrical engineering from Iwate University and an ME and DE in information engineering from Tohoku University in 1975, 1981 and 1984, respectively. He worked at Nippon Business Consultant Co., Ltd. from 1975 to 1978. He was a research associate in the Department of Communication Engineering at Tohoku University from 1984 to 1986, an associate professor of computer science at Sendai National College of Technology from 1986 to 1987 and an associate professor of the Department of Computer and Information Sciences at Iwate University from 1987 to 1991. He is a member of SAS Japan, IEICE Japan, IPS of Japan, IEEE and ACM.