

A CORRECTION ALGORITHM FOR DISTORTED LASER IMAGES

MENENDEZ Francisco J., HALABI Osama, FUJIMOTO Tadahiro, CHIBA Norishige

Department of Computer Science

Graduate School of Engineering

Iwate University

E-mail: fcomenendez@cg.cis.iwate-u.ac.jp, [ohalabi | fujimoto | nchiba] @cis.iwate-u.ac.jp

ABSTRACT

In this paper, we demonstrate a simple, effective and robust method for accurately calibrating laser projectors.

The method relies on a digital camera placed at the location of the audience, and images are obtained in order to observe the deformations caused by the projection angle and surface deformations.

Once the deformations are detected, the necessary compensations are performed to obtain a correct projection of the desired imagery.

Furthermore, we extend this method to combine two or more projectors in order to create a single, overlapped projection area. We demonstrate that this method creates a collaborative effect between the projectors and allows the display of more complex images without increasing the scan rate.

Keywords: Laser Graphics, Laser Projection, Camera-Based, Calibration, Projector Array

1. INTRODUCTION

1.1. Background

Laser projectors are devices that rely on laser light to display images. The displays are considered to be vector-based displays because instead of turning pixels on and off, they draw straight-segment lines between pairs of x-y coordinates.

Laser projectors were first used in the 1960s [ROB96] to display images for the first ever laser show (abstract imagery accompanied with music). However, the elevated cost of the projectors, particularly the laser beam generators, has made it difficult to acquire them;

consequently, conducting research on the topic has been quite difficult.

However, with recent improvements in manufacturing methods, laser projectors have become relatively cheaper and widely available. This has opened a new window for researchers and developers to explore this new technology; furthermore, it has provided visual artists with a new tool to express themselves [KEN08].

A laser projection system is basically composed of three parts:

- 1) A computer or some other electronic source, which generates signals that control the projector.
- 2) A hardware interface that serves as a communication bridge between the computer and the projector. Depending on the technology, this interface may exist in the computer as a connected board, inside the projector, as part of its hardware, or as a third, separated component (for example, a network-based interface).
- 3) The projector usually consists of one laser beam (three beams in the case of RGB projectors); a couple of high-speed scanners (mirrors attached to their corresponding galvanometers) rotate according to the signals transmitted by the computer in order to move the laser beam and display the desired images.

1.2. Laser projector related problems

As display devices, laser projectors have certain advantages that separate them from regular raster projectors, for example,

- Laser light is a coherent, focused beam of light. Therefore, lenses are unnecessary. The image will

remain focused regardless of the distance between the projector and the screen.

- Laser light, if bright enough, can be projected in lit areas; therefore, any non-reflecting object (walls, ceilings, water fountains, clouds, mountains) can be used as a projection surface.
- Images drawn with laser light display a high level of contrast. Although raster projectors produce black colours by trying to occlude light with black pixels, some light passes through the blocking pixels. In contrast, laser displays define 'black' as the colour corresponding to the beam turned off; providing a higher rate of contrast.

Laser projectors also have drawbacks, mainly due to the physical nature of the method employed to move the beam. Limitations in rotation speed lead to problems related with image complexity and flickering. Flickering in such devices is produced due to insufficient refresh speed.

When the dot of laser light created by the projector moves over a given surface at a high speed, an effect called 'persistence of vision' allows the eye to view solid lines and shapes instead of a moving dot. However, if the speed is not sufficient to redraw the entire image before the effect disappears, it would appear as if the image goes on and off repeatedly. This is what we call 'flickering'.

Although increasing the speed would reduce the flickering effect, it causes another problem. The weight of the mirrors and its inherent rotational inertia causes deformations when drawing sharp angles as the rotation speed increases.

Some other problems are directly associated with direct eye exposure to the beam [LIA07]. Safety measures should be adopted with regard to the position of the projector and the selection of the appropriate laser power for each occasion.

1.3. Proposed method

In this paper, we aim to solve two of the abovementioned problems. To maintain the beams away from the

audience, we must sometimes place the projector in a position that does not guarantee a correct projection. To solve this problem, we will use a two-dimensional adaptive grid that will perform iterations until it produces a solution. In each iteration, a digital camera will register the changes and determine whether further correction is necessary.

We then extend this method to combine multiple projectors to increase the refresh rate and reduce flickering.

1.4 Related study

Although techniques for the calibration of laser projectors have not yet been developed, many studies have been conducted regarding raster projectors. Calibration methods for raster projectors can be categorized into two types, according to the way they use to perform geometric registration: 3D geometry and 2D image based methods.

Raskar et al. [RAS98a] [RAS98b] [RAS99] [RAS00] [RAS03], have made significant advances in the 3D geometry-based method. In their study, a single (or stereo) camera is used to capture a predefined pattern that is projected on the screen by raster projectors. This pattern is then detected and used to create a 3D mesh in projector space. Thus way, if the 3D configuration of the projection surface is known, the visual errors can be compensated by determining the position of the viewer. Calibration techniques developed by Johnson et al. [JOH07a] [JOH07b] and Brown et al. [BRO05] are based on the method of Raskar et al.; that is, they essentially create a 3D configuration of the surface before applying the corrections.

In contrast, 2D image based methods [ZOL06] [OKA06] [SUR95] do not use 3D geometry, but rather use visual cues and/or homographic transformations to perform geometric registration of the screen.

In this study, we also use a two dimensional, image based method to detect deformations on the surface. However, we further increase the robustness of the method by introducing an iterative approximation

algorithm in which a grid is reshaped until it matches the required configuration.

Moreover, most raster projector calibration algorithms mentioned aim to enlarge the display area by combining two or more projectors. In this work, however, our aim is to combine those areas into a single projection surface, in order to increase image complexity while reducing visual flickering.

An attempt to reduce flickering was undertaken by Abderyim et al. [ABD07][ABD08], by removing unnecessary jumps (blanking lines) as much as possible. In this paper, although we do not consider the blanking lines problem, implementing his method along ours could theoretically allow the creation of more complex imagery without the undesired flickering effect that usually follows.

1.5 Method outline

To elucidate our method, we break the problem into two parts. First, we explain the method employed to overcome the problems caused by incorrect projection for a single projector. Subsequently, we extend this method to produce a collaborative result between two or more projectors.

For each projector (separately)

1. Take two pictures—one with and one without a projected grid on it—of the projection surface.
2. From the two images, obtain the coordinates for every point on the grid in pixels.
3. Fit the points into a grid and transform from pixel into laser units.
4. Estimate the deformation error. If the error is below a given threshold, exit the algorithm.
5. Find the compensation values for the observed deformations. Go to step 2

For every projector (combined)

1. Find the common centre of all the projection areas and translate each grid to the new centre.
2. Calculate the average scaling value of all the grids, and resize all the grids to match a common size.

To calibrate the images:

1. For every vector in the image, find the cells within the grid the vector passes through.
2. Find the corresponding cells on the calibration grid and obtain a new set of vectors by using bilinear interpolation.

2. SINGLE PROJECTOR CALIBRATION

Raster projectors are often placed almost immediately in front of the projection area. This is possible because the light coming from the lamp does not pose immediate risks to anyone who looks into it. Hence, the projector is always placed at a convenient position. In contrast, laser light can cause retinal deterioration if it hit a person's eye. For this reason, it may not always be appropriate to place the projector straight in front of the screen; instead, it should be placed at an angle to avoid direct eye exposure (figure 1).

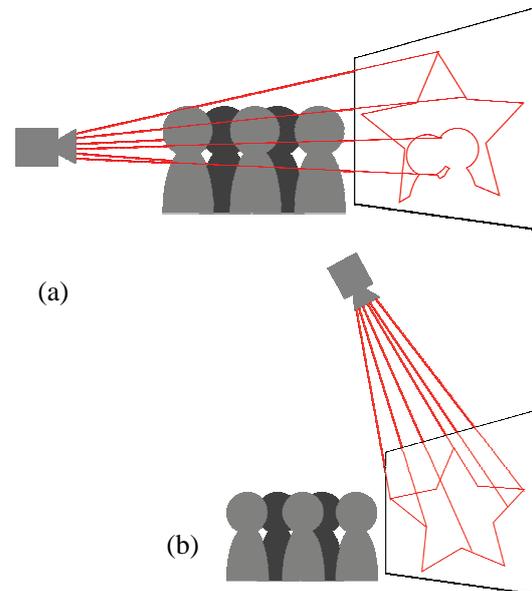


Figure 1: (a) Placing the projector in front of the screen could produce a correct projection; however, occlusion may occur and there exists the risk of laser-eye contact. (b) Our objective is to place the projector in a safe position while maintaining a correct projection.

This situation applies to not only the entertainment field but also to other situations in which lasers are involved. This method can easily be applied to calibrate lasers involved in augmented reality scenarios [GLO03] and

industrial applications [OND05].

The angle at which the projector is placed against the screen causes perspective deformations that distort the image. Furthermore, because we intend to take full advantage of the capabilities of the laser projector, we set the premise that the projection area might not always be plane, i.e. the area might have additional deformations caused by the projection surface itself.

To overcome the abovementioned problems, we propose a grid-based calibration method that relies on a camera, which is placed at the location of the audience, to automatically capture the deformations caused by the projection angle and the irregularities in the projection surface; furthermore, the calibration method can process the data and obtain a correction or ‘calibration’ grid that overcomes the deformations observed.

2.1. Image capture and processing

To fully automate our process, we take advantage of one feature that comes with most recent digital cameras. The PTP protocol allows the computer to directly command the camera [ISO05], take snapshots and transfer pictures back to the computer. This not only helps to automate the process but also to eliminate possible blurring caused by manually pressing the shutter button.

The capture process starts after the digital camera is placed at the location of the audience. In this paper, we assume that the camera is already calibrated. Distortions caused by the lens undoubtedly affect the calibration process.

Once the camera is set up, the first picture of our projection area is taken, with the projector turned off, and the image is saved in memory. Subsequently, a square grid made of interchanging dots and crosses is projected, and a second picture is taken. For calibration purposes, this grid must have an odd number of positions $N \times N$ (e.g.: 3×3 , 5×5 , 7×7 , etcetera). The size of the grid is determined by the maximum and minimum laser coordinates and the distance between the projection surface and the projector. An example of this grid is shown in figure 2.

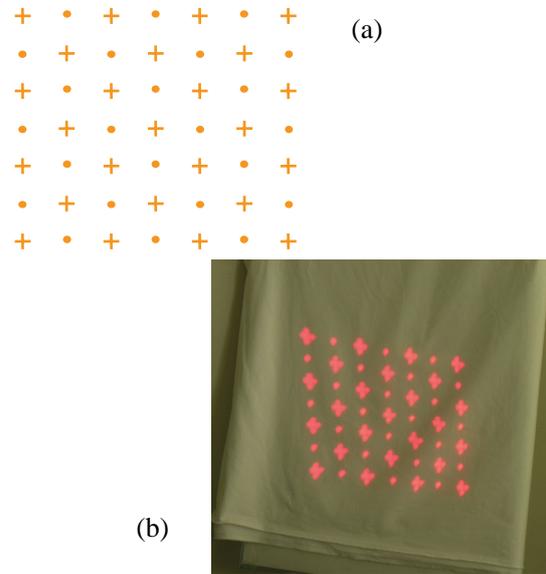


Figure 2: (a) To detect the deformations, we project a square, regular grid against our target surface with the laser beam. (b) As we capture the result with the digital camera, we can observe changes in the size and shape of the grid caused by the projection angle and surface irregularities

The next step involves the detection of the grid from the images. The difference between the first and second images is calculated by using equation (1), where $diffimg(x,y)$ stores the colour difference between the pixels from $imgbase(x,y)$ (base image) and $imggrid(x,y)$ (projected grid) at the x - y coordinate, thus removing the background and leaving only the dots from the projected grid.

$$diffimg(x, y) = \max(imgbase(x, y) - imggrid(x, y), 0) \quad (1)$$

Due to variations in luminosity or due to undesired reflections, the difference image might contain some noise that may complicate the detection process. To eliminate these noises, the difference image is binarized and the threshold is varied according to the number of positions found with respect to the number of positions expected. In other words, if the number of coordinates obtained exceeds that expected, it is reasonable to suppose that the extra positions are caused by noise; therefore, the threshold is increased to filter the noise out. This procedure is explained in equation (2), where $diffbin(x,y)$ holds the binary pixels; the threshold θ is used to filter noise.

$$diffbin(x, y) \begin{cases} 1 & \text{if } diffimg(x, y) > \theta \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

The coordinates (in pixels) of the detected grid locations are estimated from the binarized image by selecting the centre of each block of pixels that represent one grid dot or cross and storing it in memory for further processing.

2.2 Grid detection

Although the coordinates (in pixels) of the dots and crosses that form the grid are detected from the image, these coordinates lose the grid arrangement they had prior to the detection. In other words, they are simply a group of dots (pt_i) separated into two classes.

To rearrange the coordinates back into a grid arrangement, the two different patterns are used to rapidly detect the grid ($g_{x,y}$).

First, the cross coordinate that is closer to the top-left corner of the image is detected. Since, by design, the corners of the grid are always crosses, it is safe to assume that this coordinate represents the top-left corner of the grid arrangement ($g_{1,1}$).

The remaining coordinates are rearranged using the nearest-neighbor search algorithm, which is separated into two parts.

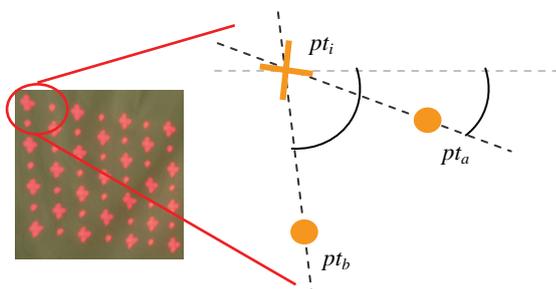


Figure 3: A detected coordinate (pt_i) and its two closest neighbors (pt_a, pt_b) being evaluated. By comparing the size of the angles, we can find which neighbor is to the right of the reference point and which one is to the bottom.

In the first part, we search for the two coordinates closest to the opposite pattern of our reference point, that is, if

our current coordinate is a cross, we search for the closest two dots and vice versa. Applying this step to the first, top-left coordinate, we find the immediate right and bottom neighbors by comparing the angles they form against the horizontal line that passes through the reference point (figure 3).

Once finding the two points, we remove them from the search list. We then advance rightwards, repeating the process for the next point until we reach the second-to-last point of the first row ($g_{1,N-1}$).

The second step involves finding only one closest neighbour of a different pattern. Since we already know the left-to-right order of the grid, from this point forward, the algorithm only searches downwards, repeating the second step until all the coordinates have been searched (figure 4).

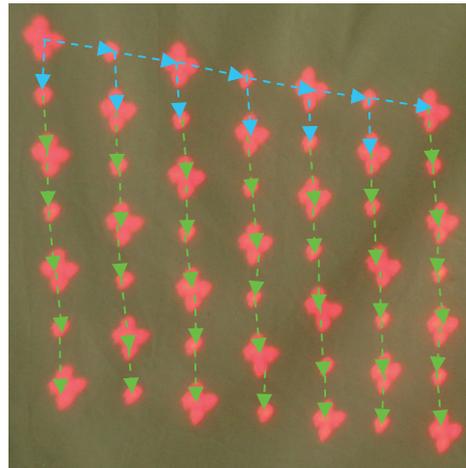


Figure 4: The algorithm starts with the top-left corner, moving to the right, finding the two closest neighbors, until it reaches the end of the first row (light blue arrows). Subsequently, it searches for only the closest neighbor, moving downwards (green arrows).

This detected grid (figure 5) describes the distortions caused by the projection angle and surface irregularities. However, since the data is obtained from an image, these points are expressed in pixel units instead of the original laser projector units. In order to obtain a grid we can actually use, we need to transform these pixel coordinates into laser coordinates.

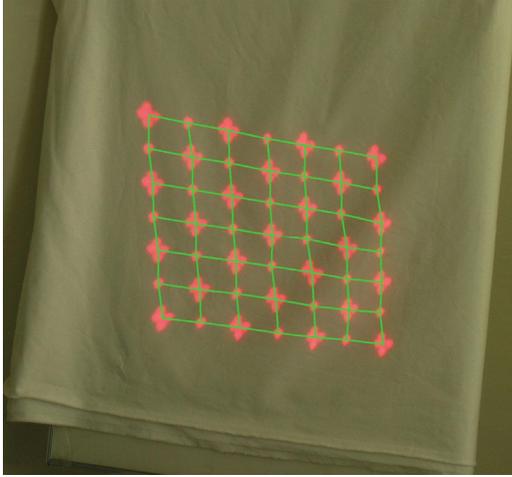


Figure 5: The points observed with the camera are now arranged in a grid fashion (drawn in green on top of the captured image)

Since the centre of the coordinate systems for the image and the laser projector are different (shown in figure 6), it is necessary to correlate the middle point of the grid with the centre of the laser projection zone. In the detected grid, the centre point will be a certain x-y coordinate in pixel units. However, since the center of projection for a laser projector is always (0, 0), we must translate the grid so that the centre points are aligned, as shown in equation (4), where $o_{cx,cy}$ denotes the centre of the laser coordinates of the ideal grid we first projected (figure 2a), $g_{cx,cy}$ denotes the centre of the observed grid, in pixel units and t denotes the resultant translation value.

$$t = o_{cx,cy} - g_{cx,cy} \quad (4)$$

Once the two grids are aligned at the centre, the observed grid is resized to transform it from pixels to laser units. The correlation between laser units and pixels (pixel-to-laser scale) is obtained by minimizing the distance between the ideal grid ($o_{x,y}$) and the grid detected ($g_{x,y}$) multiplied by a scaling value. The distance between the grids is defined as the average value of the distances between the coordinates at the corresponding grid positions (equation 5).

$$\|o - g \cdot s\| = \frac{\sum_{x,y=1}^N (o_{x,y} - g_{x,y} \cdot s)^2}{N^2} \quad (5),$$

where o and g denote the ideal and detected grids, respectively, and N^2 denotes the number of locations in the grid.

This distance is then minimized by using Newton's method for finding a local minimum, as shown in equation (6)

$$s_{n+1} = s_n - \frac{\frac{d}{ds} (o - g \cdot s_n)^2}{\frac{d^2}{ds^2} (o - g \cdot s_n)^2} \quad (6),$$

$$= s_n + \frac{\sum_{x,y=1}^N (o_{x,y} - g_{x,y} \cdot s_n) \cdot g_{x,y}}{\sum_{x,y=1}^N (g_{x,y})^2}$$

where s denotes the scale that minimizes the distance shown in (5) and n denotes the iteration number.

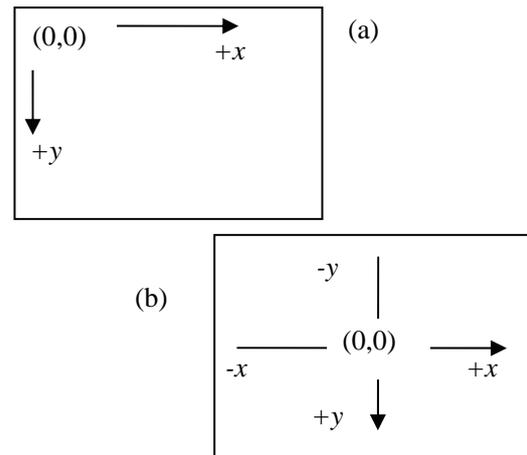


Figure 6: (a) The coordinate system for bitmaps starts at (0, 0) on the top-left corner and increases along the positive axes only. (b) The coordinate system for laser projectors is centered at (0, 0) and increases along both the positive and negative coordinates.

Once the scale value that minimizes the distance between the grids is obtained, we can consider this grid to be the visual deformation caused by the projection angle and surface irregularities; the grid is expressed in laser units. However, this is not the correction grid we require. This ‘calibration’ grid can be considered to be an inverted grid that compensates for the aforementioned deformations, and it can be defined as the difference between the visually obtained deformation grid and the original projected (regular) grid, as shown in equation (7), where $c_{x,y}$ denotes the final calibration grid.

$$c_{x,y} = 2 \cdot o_{x,y} - (g_{x,y} + t) \cdot s \quad (7)$$

2.3. Calibration

Once a calibration grid is obtained, we can correct the coordinates of our original image and overcome the observed deformations.

The calibration process is started by considering each vector in the original image and finding which cells it intersects from the initial, unchanged grid. The vector is then broken down into smaller parts, each part fully contained in its cell. Each part is then transformed by using bilinear interpolation so that it matches the deformations in the calibration grid. These new set of vectors is finally displayed with the laser projector.

Figure 7 illustrates the calibration method using a simple image as an example on a highly irregular surface with different grid resolutions.

2.4. Verification and re-calibration

Since the process employs linear interpolation to compensate for the observed errors, this method fails to calibrate accurately for very sharp angles. Therefore, to confirm that the process is successful, we repeat the attempt to capture the grid as explained in 2.1.

We measure the success of the calibration by calculating the distance between the ideal grid and the observed grid, as shown in equation (5). If the average distance between the grid positions is less than a given threshold,

we consider the projector to be calibrated.

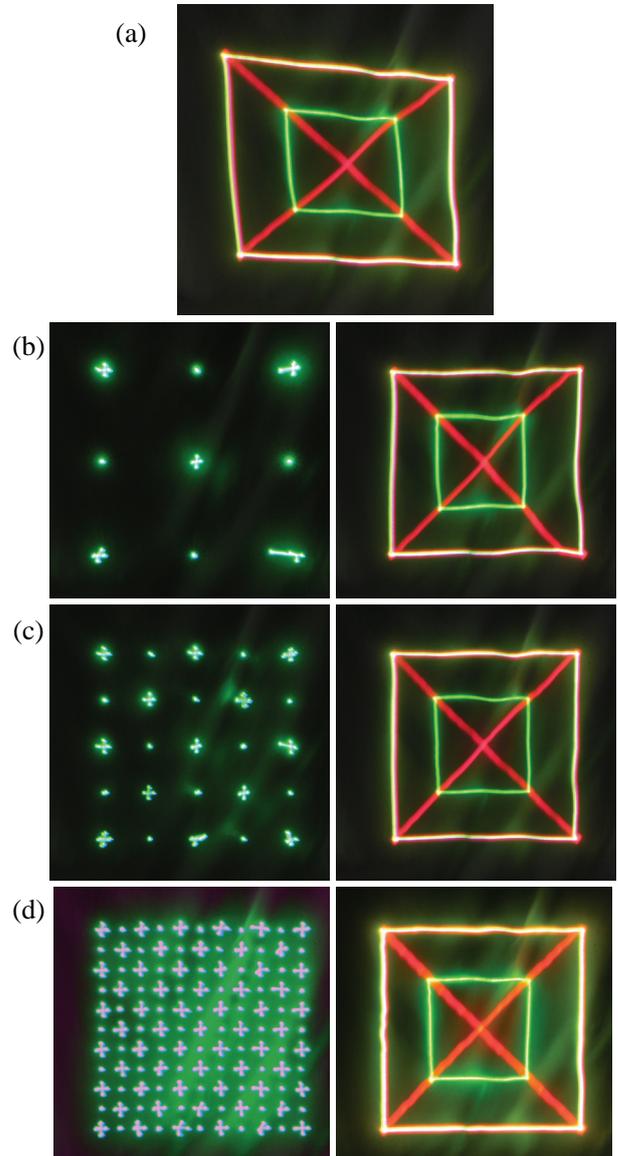


Figure 7: (a) Test pattern projected on a very irregular surface. The pattern is deformed because of the wrinkles on the fabric. (b), (c), and (d) show the calibration results for grid sizes of 3×3 , 5×5 , and 11×11 . The higher the grid resolution, the better are the chances of correcting the irregularities on the surface.

3. MULTIPLE PROJECTOR CALIBRATION

Thus far, we have explained the procedure to calibrate a single projector by using only a digital camera placed at the location of the audience. This method can further be extended in order to combine two or more projectors,

thereby allowing for the projection of more complex images without flickering or increasing the scan rate.

The scan rate in a laser projector is defined in points per second (PPS), that is, the number of coordinates the scanners can switch to in one second. At lower scan rate speeds, the laser moves slowly between coordinates, thereby failing to create an image in the retina. As the speed increases, the image becomes solid; however, sharp corners become curvy (due to inertia of the mirrors), and the overall intensity of the image decreases.

To increase the number of points drawn per second without increasing the scan-rate speed, we combine two or more projectors, split the drawing in parts and allow each projector draw a smaller section of a bigger drawing.

The process is separated into two parts. First, each scanner is calibrated individually and then the calibration grids are translated and resized to fully overlap the projection areas.

Calibration is only possible if the calibration area is smaller than the overall projection area. This allows us to expand the grid and, if necessary, move it across the projection area. There is a tradeoff between the size of the grid and the flexibility of the method. That is, the size of the grid is inversely proportional to the amount of space available for moving it (figure 8).

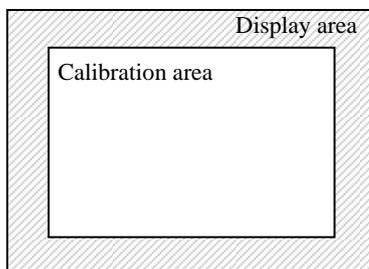


Figure 8: The calibration grid must be smaller than the total display area so that it can be moved and stretched. In our experiments, the calibration area is 75% of the display area, allowing $\pm 12.5\%$ freedom of movement in each direction. Thus, in the case of multiple projectors, the displays areas must be overlapped manually so that the calibration areas have enough freedom to be able to match each other automatically.

This relation between the display and calibration area sizes plays an important role in the calibration of two or more projectors. Since the grid cannot move beyond the display area of the projector, we must manually overlap the projection areas of the displays involved in the calibration process. Thus, the small alignment differences between projectors that may remain afterwards can be solved automatically by the method.

However, when manually aiming the projectors to produce this overlap, perspective deformations will arise from the arrangement since they cannot be in the same place. We solve these deformations by first calibrating each projector individually as explained in the previous section. Once each projector is individually calibrated, we take the middle point of each of the grids in pixels, as we considered them in (4), and find the average middle point. Since this middle point is in pixels, we multiply it by the pixel-to-laser scale value for each projector and move the centre of the projection area for each projector to the common averaged centre by applying a pre-translation to the entire grid, as shown in (8), where t_i denotes the translated value obtained in (4); for the projector i , s_i denotes the scale value obtained by minimizing (6), and t'_i denotes the pre-translation value to be applied to projector i .

$$center = \frac{\sum t_i}{n} \quad (8)$$

$$t'_i = center \cdot s_i$$

The scaling problem is solved in a similar manner. We find the average pixel-to-laser scale value of each projector and then apply a pre-scale value to the calibration grid prior to making the necessary corrections.

To perform the final calibration, we simply use the new centered, resized grid, instead of the grid obtained by using the single-calibration method.

4. RESULTS

For our experiments, we used two ILDA compliant laser projectors [ILD08]. Two interface boards—a Pangolin

QM2000 internal PCI interface and a QM2000.NET, network-based interface—were used. In order to perform the communication, we used the SDK provided by the manufacturer [PAN08].

For the single projector calibration, we conduct two studies. In the first case, we project against a flat surface and attempt to project at different angles against the screen. We continue to decrease the angle between the projector and the screen until further calibration is not possible.

In the second case, we attempt to calibrate the projection against an irregular surface.

4.1 Perspective deformation

Since the method relies on a refining loop to correct the deformations, this algorithm proves to be quite resilient to perspective deformation. As long as all the points of the grid can be projected on the surface and can be captured by the camera, the algorithm will find a calibration grid to solve the problem.

In a practical case, we tested the method against different projection angles. As the angle between the screen and the projector decreased and the deformation increased, the method relied on a larger number of iterations to converge to a solution. However, in each case, a solution was obtained.

4.2. Irregular surface

We constructed an irregular surface by using two flat boards forming an angle, and we covered the boards with a piece of fabric. The original grid (without calibration) projected into this surface is shown in figure 9a

The deformations are captured and corrected through few iterations until the error threshold is satisfied. In this case, the algorithm converges to a solution within only two iterations (figures 9b and 9c)

4.3 Verification and recalibration

The number of iterations required to converge to a solution depends of the level deformations observed by the camera. The threshold can be determined as a percentage of the maximum display area. For our experiment, we set our threshold to be close to 0.002% (25 laser units over a maximum width of 12,000 laser units) and projected a 9×9 grid, as shown in figure 9

Although the algorithm converges to the solution in 7 iterations, almost no difference can be observed after the third iteration. The result, shown in figure 10, clearly shows little change after three iterations.

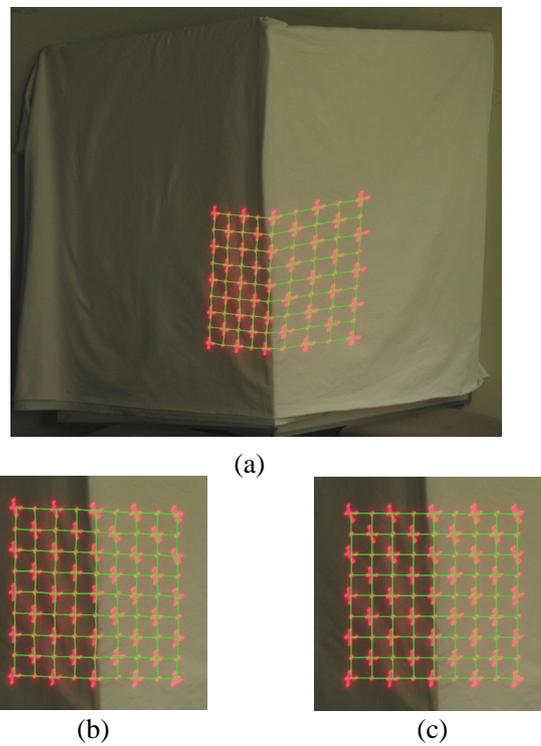


Figure 9: (a) First, we project the grid without calibration onto a highly irregular surface. (b) (Detail) after the first iteration, a close solution is obtained, but the grid continues to have some visual irregularities. (c) In the third iteration, an almost perfect square grid is obtained.

4.4 Robustness

4.4.1. Angle between screen and projector

As long as the grid can be fully captured by the camera, the method can converge to a solution given any angle between the screen and the projector. This capability of the method was tested by placing a rotating screen right

in front of the projector (ideal situation), and gradually changing the angle between the screen and the projector. The results obtained at angles of $\pm 5^\circ$, $\pm 15^\circ$, $\pm 30^\circ$, $\pm 45^\circ$, and $\pm 60^\circ$ are shown in table 1.

4.4.2. Projector roll angle

This method is also robust against minor variations in the roll angle of the projector (rotation around the z axis). Because of the design of our algorithm, the projector roll angle cannot be equal or greater than $\pm 45^\circ$, since this would disrupt the grid orientation. In other words, if the angle exceeds $\pm 45^\circ$, the left side (or right side, depending on the direction of rotation) of the grid is erroneously considered to be the top side, its top side is considered to be the right side, and so on. We conducted experiments with rotation angles of $\pm 5^\circ$, $\pm 15^\circ$, and $\pm 30^\circ$. The results are shown in table 2.

Table 1: The projector is placed against the screen at different angles. As the angle increases, the number of iterations necessary to obtain a solution also increases. The final error between the observed and the expected results increases as well, as a consequence of the large deformation caused by the projection angle.

Angle					
	$\pm 5^\circ$	$\pm 15^\circ$	$\pm 30^\circ$	$\pm 45^\circ$	$\pm 60^\circ$
Iterations	2	2	3	4	6
Error	0.7%	0.7%	0.8%	0.9%	1.0%

Table 2: As in the case of perspective deformation, if rotation is performed around the axis of the projector, the number of iterations increases with the angle. The final error also increases at higher angles.

Angle			
	$\pm 5^\circ$	$\pm 15^\circ$	$\pm 30^\circ$
Iterations	2	3	5
Error	0.6%	0.8%	0.9%

4.4.3. Grid occlusion

Unfortunately, to determine a threshold for image binarization, the method relies on the number of points detected. If one or more points are occluded, the method fails to detect the grid properly. In our future study, we

will examine the possibility of combining two projectors to compensate for the occluded areas (one projector will display a pattern in an area where the other cannot display it).

4.5. Dual calibration

In the case of multiple-projector calibration, we must overlap the projection areas as much as possible before the calibration process begins.

The deformations are corrected for each projector; subsequently, the method compensates for the differences in size and positioning of the projectors involved. The method appears to work perfectly as long as the first premise is true. If the initial placement of the projectors is such that the display areas are too far from each other, the method fails to find an overlapping zone. This happens because of limitations in the amount of displacement of the projection area that can be performed by using software.

In our experiment, we used two projectors placed at different distances from the screen (changes in size) and at different angles (changes in perspective). The algorithm successfully manages to fix the differences between the two devices to create a single, overlapped projection zone.

An example of this experiment is shown in figure 12. An image is vectorized so that it can be displayed using our laser projectors; the image is then broken in two parts and displayed using each projector, prior and post calibration. The projection surface is at an angle (as shown in figure 9) and, to avoid flickering, the scan rate was set to the maximum possible rate for each projector. As long as the number of points in the frame is less than the maximum number of points per frame possible, flickering will not occur, regardless of the shape and number of vectors displayed by each projector.

To test the effectiveness of our method, we created frames with increasing complexity and displayed them using both the methods (single and dual projection), and we varied the scan rate until a noticeable flickering effect appeared. To measure the flickering effect, we set

up a camera that can capture images with a shutter speed of 1/30 s. This value is selected on the basis of known animation frame rates. To avoid flickering, frame rates for video playback are set to around 30 frames per second.

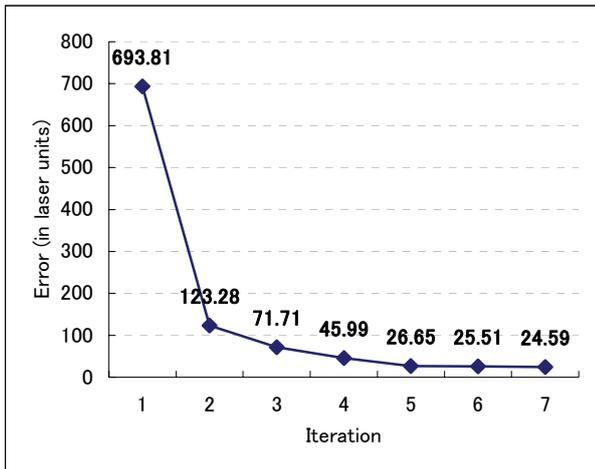


Figure 10: This graph shows the convergence curve for the error values of a 9×9 grid experiment. As the iterations increases, the error decreases exponentially.

In this way, if the picture taken showed the entire drawing as it was intended, the scan rate was adequate and we could assume that no flicker appeared (the refresh ratio was 30 or more times per second). However, if the picture was incomplete, this meant that the projector failed to refresh at an appropriate speed (less than 30 times per second). The results are shown in figure 11. It should be noted that this value was selected to test the performance of the method. Flickering as observed by people depends on various factors such as their sensitivity to light and the level of illumination of the room where the images are displayed. Analysis of these factors is beyond the scope of this study.

From the results, it is inferred that using two projectors instead of one allows us to draw more points per frame by using almost exactly half the scan rate. In other words, by using the same regular scan rate as that of a single projector, we can achieve images that are twice as complex.

5. CONCLUSIONS

We have demonstrated a method that automatically corrects the deformations caused by projection angles and surface irregularities for single projectors. By using the method, we can safely project images onto a surface without risking eye injuries on the viewers.

We also extended this method by combining two projectors to fully overlap, creating a single projection area. This new area can help reduce the flickering effect for complex images with large numbers of vectors, while maintaining the same scan rate.

6. FUTURE STUDY

Although we have implemented a consistent method for combining multiple laser projectors for collaborative projection, the problem of obtaining an adequate break-down of the image and assigning the responsibilities to each projector according to their capabilities remains unsolved.

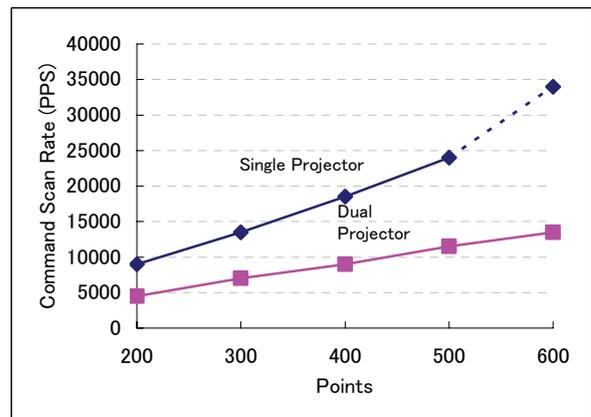


Figure 11: As the number of points increases, it becomes necessary to increase the scan rate to avoid flickering. By combining two projectors, the scan rate could be halved, allowing more complex images to be drawn at the same scan-rate. The sharp increase in the necessary scan rate in the case of a single projector is due to a hardware limitation. Although the speed set by software was close to 35,000 points per second, the actual speed at the scan heads might have been lower, and therefore, there was an apparent sudden increase in the necessary scan rate.

This would be helpful, for example, in a heterogeneous environment, where faster projectors can compensate for slower ones.

Furthermore, as explained earlier, if any part of the grid

cannot be captured by the camera due to occlusion, the method fails to detect the grid. To solve this problem, we will extend our method of combining two projectors to fill the gaps produced by occlusion.

Another interesting point of study is to extend this method to handle not only still frames but also animations. In this case, the speed of the projectors must be taken into account to avoid the slower projector lagging behind. Issues such as synchronization and frame-rate will become areas of research.

ACKNOWLEDGEMENTS

This work was partially supported by the Japanese Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Exploratory Research 19650014.

REFERENCES

[ABD07] *An Efficient Drawing Algorithm for Laser Graphics*, Purkhat ABDERYIM, Osama HALABI and Norishige CHIBA, p9, Tohoku-Section Joint Convention of Institutes of Electrical and Information Engineers 2007

[ABD08] *An Efficient Drawing Algorithm for Laser Projector*, Purkhat ABDERYIM, Osama HALABI and Norishige CHIBA, NICOGRAPH International. May 2008

[BRO05] *Camera-Based Calibration Techniques for Seamless Multi-Projector Displays*. BROWN Michael, MAJUMDER Aditi, YANG Ruigang, IEEE Transactions on Visualization and Computer Graphics, March/April 2005 (Vol. 11, No. 2) pp. 193–206

[GLO03] *Projection Augmented Reality System for Computer Assisted Surgery*. GLOSSOP N., WEDLAKE C., MOORE J., PETERS T., WANG Z. In Proc: Lecture notes in computer science, Vol. 2879, (2003).

[ILD08] *International Laser Display Association*. <http://www.laserist.org/>. Accessed July 2008.

[ISO05] *Picture transfer protocol (PTP) for digital still*

photography devices. ISO Standard 15740:2005

[JOH07a] *Real-Time Projector Tracking on Complex Geometry Using Ordinary Imagery*, JOHNSON T., and FUCHS H. IEEE Conference on Computer Vision and Pattern Recognition, 2007, p.1-8, June 2007

[JOH07b] *A Personal Surround Environment: Projective Display with Correction for Display Surface Geometry and Extreme Lens Distortion*, JOHNSON T., GYARFAS F., SKARBEZ R., TOWLES H., and FUCHS H. IEEE Virtual Reality 2007, March 2007

[KEN08] *Planetary VJ performance at Science Dome*. MOTOMURA Kenta, HALABI Osama, NASUKAWA Norihiro. <http://kenta.edu.iwate-u.ac.jp/dome/>

[LIA07] *LIA Laser Safety Guide*, Laser Institute of America Safety Comitee, 2007, ISBN# 0-912035-06-4

[MEN07] *Camera-Based Calibration for Laser Projectors*, Francisco MENENDEZ, Osama HALABI and Norishige CHIBA, p.3, Tohoku-Section Joint Convention of Institutes of Electrical and Information Engineers 2007

[OND05] *Improving the appearance of all textile products from clothing to home textile using laser technology*. ONDOGANA Z., PAMUKA O., ONDOGAN E., OZGUNEY A.

[OKA06] *Autocalibration of an Adhoc Construction of Multiprojector Display Systems*, OKATANI T., and DEGUCHI K. Proceedings of the International Workshop of Projector Camera Systems 2006.

[PAN08] *Pangolin Laser Systems*. <http://www.pangolin.com/>. Accessed July 2008.

[RAS98a] *Seamless Projection Overlaps Using Image Warping and Intensity Blending*, RASKAR R., WELCH G., and FUCHS H, Fourth International Conference on Virtual Systems and Multimedia, (Gifu, Japan), Nov. 1998

[RAS98b] *The Office of the Future: A Unified Approach*

to *Image-Based Modeling and Spatially Immersive Displays*, RASKAR R., WELCH G., CUTTS M., LAKE A., and FUCHS H., Proceedings of the 25th annual conference on Computer graphics and interactive techniques, p.179-188, July. 1998

[RAS99] *Multi-Projector Displays Using Camera-Based Registration*, RASKAR R. BROWN, M., YANG R., CHEN W., WELCH G., TOWLES H., SEALES B., and FUCHS H., Visualization Conference, IEEE, p. 26, 1999.

[RAS00] *Immersive Planar Display using Roughly Aligned Projectors*, RASKAR R., Proceedings of IEEE Virtual Reality, p. 109-116, 2000.

[RAS03] *iLamps: Geometrically Aware and Self-Configuring Projectors*, RASKAR R., BAAR J.,

BEARDSLEY P., WILLWACHER T., SRINIVAS R., and FORLINES C, Proceedings of ACM SIGGRAPH, 2003, 2003.

[ROB96] *Laser F/X: The Light Show Handbook*, ROBERTS L Michael, Laser F/X International, Burlington, 1996.

[SUR95] *Scalable Self-Calibrating Display Technology for Seamless Large-Scale Displays*, SURATI R, Ph.D. Thesis, 1995.

[ZOL06] *Passive-Active Geometric Calibration for View-Dependent Projections onto Arbitrary Surfaces*, ZOLLMAN S., LANGLOTZ T., and BIMBER O. Workshop on Virtual and Augmented Reality of the GI-Fachgruppe AR/VR, pp. 181-191, 2006

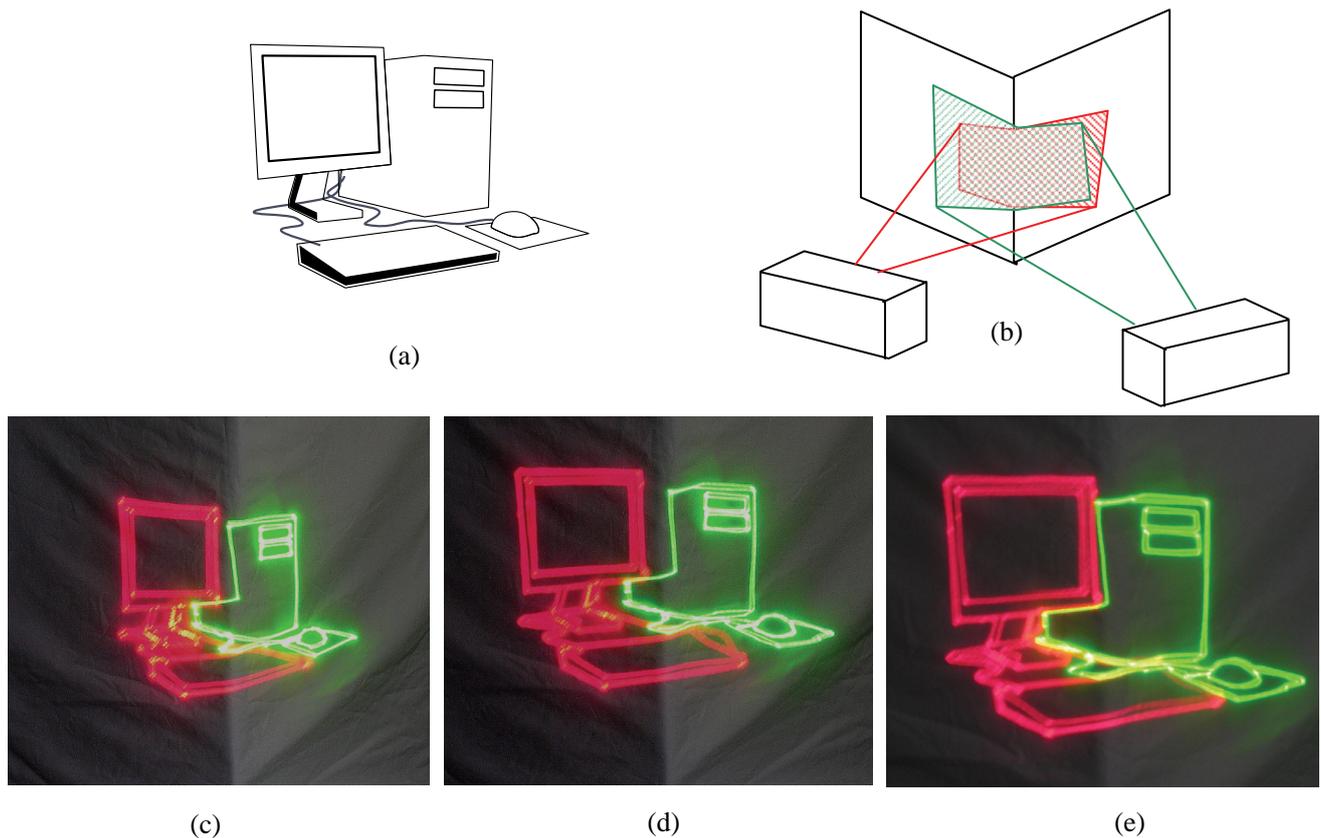


Figure 12: (a) An example image is vectorized and broken into two pieces so it can be displayed with each laser projector. (b) In this experiment, we created a setting in which each projector is against an angle towards an irregular screen. (c) Each projector displays the assigned part without calibration. (d) Each projector is calibrated individually. The shapes are correct but the sections are disconnected. (e) The final image is recomposed after the two projectors are combined.

AUTHORS' BIOGRAPHIES



FRANCISCO J. MENENDEZ is currently a Ph.D. candidate in Computer Science at Iwate University. His research interests include computer graphics, and computer simulation. He received the B.E in computer Science from Mendoza University, Argentina and an ME in Computer Science from Iwate University in 2002 and 2006, respectively. He is a member of IEEE



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.