Machine Vision and Applications
https://doi.org/10.1007/s00138-019-01044-6

ORIGINAL PAPER

Scalable and view-independent calibration of multi-projector display for arbitrary uneven surfaces

Shahaboddin Askarian Bajestani[1] · Hamidreza Pourreza[1] · Shayan Nalbandian[2]

Received: 12 September 2018 / Revised: 22 June 2019 / Accepted: 30 July 2019

Abstract
Most existing systems for calibrating multi-projector display suffered from several important limitations such as dependence on point of view, restriction on the display surface and moreover the number of projectors and using obtrusive markers. In this paper, a new method for view-independent calibration for multi-projector displays is presented. Given that the calibration problem of a multi-projector display is an optimization problem, we compute the calibration parameters by writing the appropriate energy functions for the geometric calibration phase. In this method, the camera and projector are introduced as a pair. In the first step, calibration of the pair of camera and projector was carried out. After that, the calibration problem of the system decreases to estimation of the number of camera positions relative to each other. In this method, there is no particular shape for the screen. In addition, due to the 3D shape of the screen, this method is view independent and eventually the image can be wallpapered on the screen. According to the tests carried out to evaluate the system, the accuracy of the proposed system is sub-pixel, and as a result, no misalignment is observed by the human eye in the overlapping area of projectors.

Keywords Multi-projector display · Camera calibration · Projector calibration

1 Introduction

Today, due to the ever-increasing development of technology and digital revolution, significant advances have been made in various areas, including display systems. The need to produce systems that, in addition to large dimensions, have the right quality for display, is felt more and more every day. Several innovations have been made to meet these needs, for example the introduction of major LCD TVs. Today, the latest technology introduced by technology companies is the multi-projector display system. These systems have features and specifications that are, in most cases, the best and most cost-effective solution for display in public and large environments. If a larger display is available in the future, it is impossible to have that flexibility and scalability based on the resolution and pixel count. However, the only way to build a large, high-resolution display is to use the mosaic layout of a number of projectors. Using multi-projector displays eliminates the need of producing large-scale displays. Additionally, in order to recent advances in this field, these displays are able to produce unified images on usual surfaces (flat, quadratic, etc.) by using software algorithms.

The construction of a multi-projector display requires the implementation of two phases of geometric calibration and color calibration. In the geometric calibration phase, we are looking at how to create distortions in the image so that the image is correctly visualized by the viewer, and in the color calibration phase we are looking to match the color component among different projectors. We focus in this paper on the geometric calibration section. We will continue to review the works in the geometric calibration section.

According to the papers presented in the field of multi-projector displays, geometric calibration can be done in two ways: view-dependent geometric calibration and view-independent geometric calibration.

✉ Hamidreza Pourreza
hpourreza@um.ac.ir

Shahaboddin Askarian Bajestani
sh.askarian@mail.um.ac.ir

Shayan Nalbandian
shayannalbandian14@yahoo.com

1 Machine Vision Laboratory, Department Of Computer Engineering, Faculty of Engineering, Ferdowsi University of Mashhad, Mashhad, Iran

2 Department of Computer Engineering, Faculty of Engineering, Azad University of Mashhad, Mashhad, Iran

Published online: 27 August 2019

In the first case, the final integrated image is corrected from the viewpoint of the camera geometrically, but in the second case, the image appears to be correct from each viewpoint. In fact, the image is wallpapered on the screen. Whether the image needs to be geometrically correct from a viewpoint or whether the image is geometrically correct from all points is due to the application that it is intended to use such a display. When your viewer is one person (such as a flight simulator) and the goal is to correctly visualize the image from one viewpoint, view-dependent geometric calibration methods are desirable; however, in most applications (for example, educational and promotional applications), the number of viewers is higher and it is desirable to perform view-independent geometric calibration.

Based on the notation in Reference [1], in view-independent calibration, we seek to determine the geometric transformation of the projector coordinate system to the screen coordinate system. Consider the projector coordinate system (x, y), the camera coordinate system (u, v) and the screen coordinate system (s, t), such a geometric transformation of the projector coordinate system to the screen coordinate system is defined as Eq. (1).

$$G_{(x,y)\to(s,t)} = F_{(u,v)\to(s,t)}.H_{(x,y)\to(u,v)} \tag{1}$$

In the view-dependent calibration, Eq. (1) is reduced to Eq. (2) and the display coordinate system will be eliminated from our calculations.

$$G_{(x,y)\to(s,t)} = H_{(x,y)\to(u,v)} \tag{2}$$

Therefore, in view-independent calibration, it is desirable to determine each geometric transform H and F, and in the view-dependent calibration, the geometric transformation H is just to be determined. We will first review the papers presented in the field of view-dependent calibration, and then, we will examine the papers presented in the field of view-independent calibration.

As we said earlier, in view-dependent calibration, we do not need to explore the 3D shape of the screen. In fact, we only seek to discover the geometric transformation between the camera and projector. In finding this geometric transformation, the most basic solution to the problem is the correspondence between the camera image and the projector image, which has been solved in various papers in various ways. The paper [2] has used the structured light pattern for the solution of the correspondence problem. [3,4] first solved the problem for a number of points using the Gaussian points projection. Then, they used interpolation methods to create the correspondence between the rest of the points. Ahmed et al. [5] estimated the distortion caused by the image projection on a uneven surface with the help of the Bezier functions. After calculating the parameters of the two-dimensional Bezier function, which represents the surface of the screen, reverse Bézier is applied to the image points so that the image does not have a distortion after projection.

In a view-independent calibration and if the screen is a flat surface, we can make the above geometric transformations based on the homographies [1]. Usually, a camera is capable of observing a 2*4 array of projectors in its field of view. In the face of larger screens, reference [6] presented a method that uses one camera. The camera is completely controlled by the computer, so that it can pan–tilt–zoom. The motion of the camera is such that it can cover the whole screen. The camera moves to observe the points and lines projected by each projector. With the help of the camera, the relationship between each projector and the screen is discovered. Then, the homographies are calculated using the simulated annealing algorithm to minimize the error between the corresponding points of the projectors and the angle between the corresponding projectors line. While this method works well, it is unfortunately very slow. Reference [7] provided a method in which several cameras are used. Several cameras observe the screen, and due to the overlapping field of view, cameras communicate with each other through homography. One camera is considered as a reference, and each camera is related to the reference camera through concatenating of homographies. The reference camera is then related to the screen, and so the relationship between each camera and the screen is obtained through a series of homography multiplications. This method, like [6], does not spend much time and does a geometric calibration operation for 32 projectors in just a few minutes. Note that this method is only suitable for smooth displays. Factors such as camera lens distortion and projector lens distortion reduce the accuracy, so that in some cases the use of a compact model such as homographies has many errors. Reference [8] used a nonlinear relationship (Eq. (3)) to find geometric transformation between the projector and the camera. This method cannot be extended to larger screens.

$$(u, v) = \sum_{i=0}^{3}\sum_{j=0}^{i}(A^{ij}, B^{ij}) \cdot X^{i-j} \cdot Y^{j} \tag{3}$$

Reference [9] used one camera and used a new model to explore the geometric transformation between the projector and the screen. In this model, the distortion of the projector was estimated using the Bezier functions. In fact, in this model nonlinear factors such as key-stone, radial and tangential distortion are included in a function. In this method, the sampling was carried out as sparse and based on feature points. These points are supposed to perform an estimate of a function parameters, which assigns each point of the projector coordinate to the camera coordinate system. Then, using these parameters, the projector image changes

so that it is finally geometrically aligned. Sparse sampling for geometric calibration seems appropriate because the Bezier functions suggest a piece-wise curved representation instead of a linear one that is more suitable for the distortion of the lens that turns lines into curves. In addition, in the case of sparse sampling, we do not have to use a high-resolution camera. Paper [10] used two cameras to estimate the 3D shape of the screen. The screen used in this article is a quadratic. Both cameras are placed in a position that can observe the entire screen. One of the cameras is the reference and using these two cameras, the corresponding points were discovered in two cameras, and then, the second-order approximation was estimated for the screen. However, the proposed method is only able to calibrate uneven screens that have a quadratic shape. Paper [11] used a piece-wise method to estimate the geometry of the screen. In this method, a checkerboard pattern is installed on the top and bottom of the screen. With the help of this checkerboard pattern, the parameterization of a 2D piecewise linear is done for the screen. This 2D parameterization is also carried out for projectors, which is done with the help of a camera that monitors both the screen and the image of the projector. Because the sampling has not been done in dense (only on the top and bottom of the screen), the distortion phenomena are visible in the middle sections of the image. Articles [12,13] estimate 3D geometry of the screen in another way. In these articles, an estimate of the camera matrix is calculated using assumptions on the screen and two-step nonlinear optimization problem solving. Then, the screen shape was estimated. Note that in these articles, the specificity of the screen shape is used (the screen is vertically extruded) and the camera matrix is calculated. Also, the length-to-width ratio of the screen should be specified. Article [14] considered the screen as cylindrical surface with known parameters and thus, in accordance with the cylindrical relations, created a geometric transformation between the image points and the points of the screen.

Since the processing of a video for playback on a multi-projector display system needs to be done in real time, there have been many articles focusing on improving the performance of a multi-projector system such as [15,16] and [17]. Because of sharing the optical path for the view pairs of a stereoscopic projector, Reference [15] improved performance by eliminating additional computation and processing both views at the same time. Reference [17] initially used a PC (personal computer) to preprocess the display system and then with the help of the FPGA performed video input preprocessing without using the PC. Reference [17] presented an FPGA-based hardware architecture for geometric improvement of the projector image. This article removes the limitation of implementing multi-projector display systems on desktop systems and execution of the program on the GPU and provides a scalable system by providing a new architecture. The method [18] also relates to one projector and is not universally applicable to multi-projectors.

In Sect. 2, we will have a comparison based on different criteria between previous works in the field of geometric calibration of multi-projector displays. Our proposed method for geometric calibration of a multi-projector display will be presented in Sect. 3. Also, experimental results will be presented in Sect. 4, and finally, a summary about what we have done will be presented in Sect. 5.

2 Comparison of previous works

The presented articles can be compared from the following perspectives:

2.1 Automatic or requiring human monitoring

Some of the above articles are automated [2–4,6,7] and [8] and others [5,9–13] and [14] require launching by a human operator. A system is called automatic if a usual user can set it up by placing projectors beside each other and running software. Consider that a usual user does not have any information about working with the system except placing projectors overlapping each other. In our viewpoint, if the setting up the system needs other considerations, the system is not automatic.

2.2 A camera or several cameras

The number of cameras in presented works is different. Increasing the number of cameras in a multi-projector system may be due to the lack of coverage of the entire screen by a camera or the estimation of a 3D shape screen with the help of several cameras. Typical single-camera methods are not universally applicable to multi-camera methods or hardly generalizable. For example, in [7], the size of the screen is such that the entire screen is not visible in the camera's field of view. Such a display is observed by several cameras, and for this purpose, the connection between different cameras has been discovered through homographies. However, in the other methods presented, there is no way for the author to claim that the proposed method can be extended to a multi-camera system. This system is appropriate only for flat surfaces. In other words, because the screen is flat and the depth of all spots on the screen is equal, all of coordinate systems are considered two dimensional and this issue is not able to be extended to uneven surfaces.

2.3 Having primitive information of the screen or lack of primitive information of the screen

On the methods presented in the field of multi-projector displays, the use of a marker around the screen is known as an obtrusive element [12]. Having primitive information on screen geometry makes it easy to discover geometric transformation of the camera to the screen. For example, the smoothness of the screen (means flat screen) ensures that the image on the screen is also rectangular and is the only difference with the original image on the longitudinal and transverse scale (assuming that the projector is perpendicular to the screen). In this way, using the markers, the display area is specified and each point on the screen is defined in the coordinate system of the screen. Such a method can not be generalized to uneven screens since in uneven surfaces the third dimension of the screen does not have a constant value for all points and there will be an image distortion and this amount should be discovered in some way. Having previous information about the screen removes the problem from an automatic mode, and it is desirable to create a multi-projector display system without access to display information. For example, articles [12] and [13] put previous assumptions on the screen (without the use of a marker), but assumptions such as knowing the length-to-width ratio of the screen and having a specific type of screen, leave the provided method from auto mode. In these methods, having the previous information on the display plays a key role in performing the calibration, so that by violating such conditions, the presented algorithm is not capable of geometric calibration.

2.4 Relative alignment of projectors

The relative alignment of projectors is considered to be a disturbance before calibration operations because again the issue leaves auto mode. However, with relative alignment, we only need to find the corresponding points in the overlapping regions, because adjacent projectors should display the same pixels at each point of the overlapping region.

2.5 Ability to generalize to larger screens

All the methods presented in the geometric calibration section cannot be extended to larger screens. When the screen magnifies, a camera alone cannot cover the entire screen in its field of view. On the other hand, placing the camera at a large distance from the screen reduces the accuracy of the calibration process and usually, the camera is placed at a distance from the screen which the viewer is expected to observe the screen. Due to the fact that the geometric calibration section does not require a camera with special features, the number of cameras is not an important parameter in geometric calibration. When the screen is smooth, it is easy to generalize the system to several cameras based on homographies, so that communication between different cameras can be done through homographies. However, a method that is not based on homographies and can easily be extended to large screens is still not provided.

In constructing a multi-projector display, we want to eliminate all the additional factors known in this system. In fact, the only factor that is important in building a multi-projector display is to put projectors in a way that adjacent projectors overlap. If we can solve the rest of the geometric calibration operations using software with this simple assumption, then we call such a system an automated system. Regarding the papers on flat surfaces, it can be said that the set of tasks performed on these surfaces has reached maturity. On flat screens, the only problem that may cause the calibration process to be distorted is the camera and projector lens distortion and the nonlinear estimates described in the previous sections can be an appropriate response to these nonlinear factors. Unfortunately, in a view-independent calibration for uneven surfaces, there is still no general and automatic way to solve this problem. The methods presented have primitive information about the screen, or the method presented is only capable of answering some of the changes. The main challenge in these methods is to estimate the 3D shape of the screen. If we can estimate the relative depth of the screen points, then points can be defined on the screen in its own coordinate system. There are different methods for estimating the depth of the screen points, and in this paper, we will consider one of these methods based on the camera and projector pair. After extracting the relative depth of some points on the screen (and not all points), we can estimate the 3D surface of the screen. Due to the work performed in the geometric calibration, we do not need a particular camera to implement the algorithms of this section. On the other hand, the design of view-independent displays is more public. An approach to view-independent calibration can be to consider a camera for each projector. Such a system also has the ability to generalize to large screens, as cameras and projectors can be combined to cover the entire screen. As we mentioned earlier, most of the methods presented here cannot be extended to larger screens, and this is a negative point for a system that in most applications, the size of the screen is such that the field of view of a camera does not cover the entire screen. In the following, we suggest a method based on camera–projector pairs.

A new method for calibrating a multi-projector display is presented in this paper. Viewpoint independence and having arbitrary uneven display are the challenges in the field of multi-projector displays (as it is mentioned before). We have designed the multi-projector displays problem from the beginning in this paper because the other methods are not capable of presenting an approach to solve the challenges above. In this method, we designed the multi-projector

Fig. 1 A pair of proposed systems

Fig. 2 An example of an arbitrary rugged screen

displays as an optimization problem so all calibration parameters will be calculated after solving this optimization problem. Also, proposed method is designed in a way that can be used in projection on any uneven surfaces in addition to projection on a flat screen. In presenting this system, we did as much as effort we could to make no need of user monitoring in setting up a multi-projector display and make the calibration operation done in a totally automatic way.

3 Proposed method

The proposed system consists of camera–projector pairs so that in each pair, the camera's field of view covers a wider range of field of view for the projector in the same pair. In fact, in each pair, the camera observes the whole image of the projector in the same pair. Also, in each pair, the camera and projector are stay fixed with respect to each other. As a result, we assume that after the external parameters are calculated between the camera and the projector for the first time, these values are constant during system operation. This also applies to the internal parameters of camera and internal parameters of projector. In fact, after the first calculation of internal parameters of the camera and internal parameters of the projector, these values will remain constant during system operation. Figure 1 shows a pair of proposed systems. As you can see, the camera is installed at a distance from the projector's lens, because, as we will explain later, the pair of camera and projector will play the role of stereo vision in the same pair.

In this system, similar to the past works, neighboring projectors have overlaps. Considering this overlapping area prevents seam in the overlapping area, because in case of displacement in each projector, resetting the calibration process will calibrate the system again. We did not consider a specific shape for the display surface in this system. The display surface can be flat, a quadratic or a vertically extruded and so on. In addition, the display surface may be a 3D object covered with white cloth and aiming to project that object. Such applications have an advertising aspect and create appealing displays, an example of which is shown in Fig. 2. In this article we will use the term "arbitrary uneven" for these types of screens. In the proposed system, there is no preconceived overlap between the placement of different pairs next to each other, except the side-by-side placement of neighboring projectors in an overlapping formation. Pairs are arranged together and each will project part of the display surface.

To create a multi-projector display, the problem we are facing is the system calibration problem. In the initial step, calibration of the internal parameters for the camera, calibration of the internal parameters for the projector and calibration of the external parameters of the camera and projector in each pair should be performed. After this step, the system's calibration problem will reduce to camera position estimation problem. To solve this, we will first calculate the initial estimate of the position of each camera with respect to adjacent cameras. Note that the reason for calculating the position of each camera related to its adjacent cameras and not the reference camera is that there is no guarantee that each camera will overlap with the reference camera in its field of view and thus have a common space. Clearly, in our proposed system, cameras do not observe the same view, but in fact we are faced with a wide-area problem. In this regard, we will first calculate the initial estimate of each cameras position relative to the adjacent cameras. Then, by creating the adjacency graphs of the cameras and finding the shortest path from each camera to the reference camera, we will calculate the relative position of each camera with respect to the reference camera and finally by writing an energy function, we will optimize the position of each camera with respect to the reference camera.

Because cameras do not observe the same view, each camera is related to the reference camera through concatenating of rotation and transition matrices. Regarding this issue, the fact that which camera is considered as a reference and also the path that relates each camera to the reference camera is a problem to be solved in the next step. In fact, this will be achieved taking into account a graph and then selecting the

reference camera and finally selecting the shortest path from each camera to the reference camera. After calculating all the calibration parameters, we will consider a scenario for projecting a three-dimensional object which will be explained more in the relevant section. In the end, we summarize the system calibration algorithm in a multi-projector display as follows:

1. Camera–projector pair calibration
   - I Calibration of the internal parameters of the camera in each pair
   - II Calibration of the internal parameters of the projector in each pair
   - III Calibration of external parameters between camera and projector in each pair
   - IV Extraction of the three-dimensional shape of each object the pair observes in its field of view
2. Solving the problem of estimating the position of several cameras relative to each other
   - I Imaging and detecting the center of gravity of feature points
   - II Creating matrix W or feature point weight matrix
   - III Calculating the position of each camera relative to adjacent cameras
   - IV Creating adjacency graphs of cameras
   - V Selecting the reference camera
   - VI Finding the shortest path between each camera and reference camera
3. Image production scenario
   - I Extracting the 3D shape of the screen in the unified coordinate system
   - II Wallpapering the image on the 3D shape of the display
   - III Providing lookup tables between 3D object and projectors' images

3.1 Calibration of the camera–projector pair

The camera–projector pair calibration is performed according to the method presented in [19]. To clarify this topic, we will review this article. (You can also use the method presented in [20].)

As you know, the Zhang method [21] is widely used because of its simplicity and high accuracy in camera calibration systems. In this method, a checkerboard is used to calibrate the camera. The checkerboard with specific square (or rectangles) dimensions is positioned in different places from the camera and then the camera capturing this board. Due to the corner distances in the checkerboard coordinate system and the camera image, Zhang provides a method for calculating camera parameters. The paper [19] also calibrated the camera using Zhang method in a system that includes a camera and a projector. Before describing the method presented in this paper, we will first review the camera–projector model in this paper.

3.1.1 Camera and projector model

Suppose $X \in R^3$ is a point in the reference coordinate system centered on the projection center of the camera. Also, suppose $u \in R^2$ is the image coordinate for X point in the camera image, then X and u will be related by Eqs. (4) to (9):

$$X = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \widetilde{u} = \begin{bmatrix} \widetilde{u}_x \\ \widetilde{u}_y \end{bmatrix} = \begin{bmatrix} \frac{x}{z} \\ \frac{y}{z} \end{bmatrix} \tag{4}$$

$$u = K_c.L(\widetilde{u}) \tag{5}$$

$$K_c = \begin{bmatrix} f_x & \gamma & o_x \\ 0 & f_x & o_y \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

$$L(\widetilde{u}) = \begin{bmatrix} \widetilde{u}.(1 + k_1 r^2 + k_2 r^4) + \Delta_t(\widetilde{u}) \\ 1 \end{bmatrix} \tag{7}$$

$$\Delta_t(\widetilde{u}) = \begin{bmatrix} 2k_3\widetilde{u}_x\widetilde{u}_y + k_4(r^2 + 2\widetilde{u}_x^2) \\ k_3(r^2 + 2\widetilde{u}_y^2) + 2k_4\widetilde{u}_x\widetilde{u}_y \end{bmatrix} \tag{8}$$

$$r^2 = \widetilde{u}_x^2 + \widetilde{u}_y^2 \tag{9}$$

In these equations, K_c is the internal parameters matrix of the camera, k_1 and k_2 are radial distortion coefficients and k_3 and k_4 are tangential distortion coefficients of the camera. Also, assuming R and T as the rotation matrix and transition vector, respectively, which determine the location of the projector relative to the camera and also if $v \in R^2$ is considered the coordinate of the X point in the projector image, then Eqs. (10) and (11) are presented for transformation between the camera and projector coordinate system.

$$X' = \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = R.X + T, \widetilde{v} = \begin{bmatrix} \frac{x'}{z'} \\ \frac{y'}{z'} \end{bmatrix} \tag{10}$$

$$v = K_p.L(\widetilde{v}) \tag{11}$$

In Eq. (11), K_p is the projector internal parameters matrix.

3.1.2 Imaging

In the Zhang method [21], imaging should be applied on a checkerboard in different places. In paper [19] the imaging has changed so that the camera or the projector can be calibrated with acquired images. In fact, instead of a single capturing of any place on a checkerboard, a complete set of

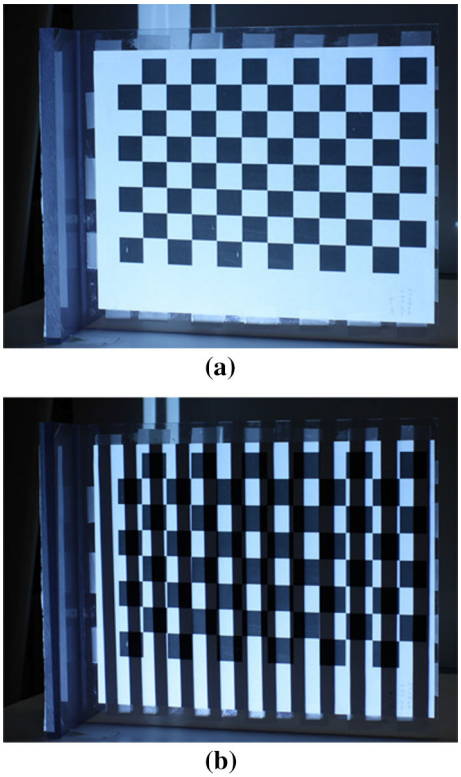


Fig. 3 An example of the images required for the calibration procedure presented in [19]. **a** Full brightness for the checkerboard and **b** projection of one of the gray code patterns on the checkerboard

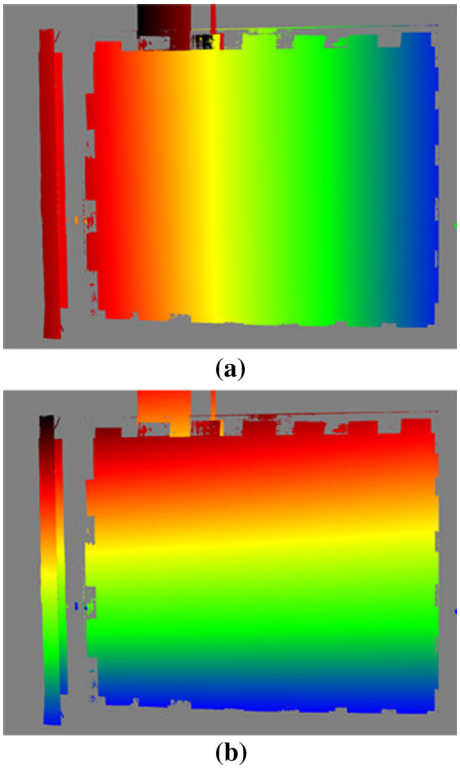


Fig. 4 An example of the decoding of the gray codes in the method presented in [19]. **a** Pixels that have the same color are related to a row from the projector, **b** pixels that have the same color are related to a column from the projector

structured light patterns should be projected on a checkerboard. This paper uses the gray code structured light pattern. Note that when capturing different patterns at a specific location on a checkerboard, the board should have no movement. Figure 3 is an example of the images needed to perform the calibration method of this paper. Figure 3a is a view of the checkerboard which is completely projected by the projector. We will need this image at the time of the thresholding of black and white stripes. Figure 3b illustrates the image of one of the structured light patterns on a checkerboard.

Figure 4 illustrates an example of the gray code decoding in the structured light method. In Fig. 4a, the pixels that have the same color are related to a row from the projector, and in Fig. 4b, pixels that have the same color are related to a column of the projector.

3.1.3 Calibration of internal camera parameters

Calibration of the internal parameters of the camera means estimating the internal parameters of the camera based on the model chosen for the camera. To perform the calibration using the Zhang method, the coordinates of all the corner points must be extracted in all the images captured from the checkerboard. Then, according to the coordinates of the corner points in different images, and also with respect to the coordinates of the corner points in the coordinate system of the checkerboard, according to the Zhang's method, we will calculate the internal parameters.

3.1.4 Calibration of the internal parameters of the projector

Projectors and cameras are described with the same mathematical model. As a result, we will use the same method to calculate the internal parameters of the projector, but the projector does not have a camera and it is not possible to capture images from the view point of the projector from the checkerboard. Nevertheless, the camera can be used to extract the coordinates of the corner points on the checkerboard from the projectors viewpoint. Note that in the imaging that was performed and after decoding the gray code in the images, we extracted the relationship between the camera image and the projector image. In the following, we will show how to use this information to determine the corner coordinates in the projector coordinate system.

Calculating the corner points in the projector coordinate system consists of three steps:

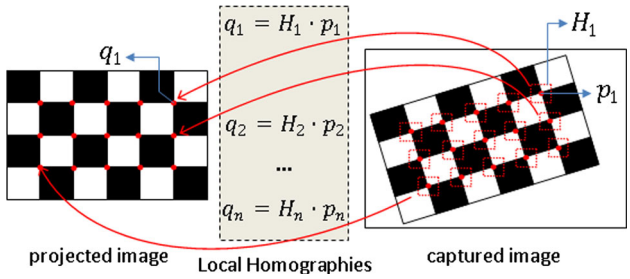


Fig. 5 Use of local homographs to convert points from the camera coordinate system to projector coordinate system—reproduced from Reference [19]

1. Decoding from gray code patterns and creating correspondence between camera image pixels and projector rows and columns
2. Estimating local homography for each corner point on a checkerboard
3. Applying the local homography to the corresponding corner point to convert the corner point of the camera coordinate system to the projector coordinate system.

Note that the obtained result from the decoding step in the gray code patterns is not directly applicable. To solve this problem, the concept of homography has been used. In addition, instead of using a global homography for the entire image, a local homography is used for every corner. Local homography is a homography that is valid only in a region of the plane. Instead of applying a single global homography to translate all the checkerboard corners into projector coordinates, we find one local homography for each of the checkerboard corners. Each local homography is estimated within a small neighborhood of the target corner and is valid only to translate that corner into projector coordinates and no other corner. Local homographies allow to model nonlinear distortions because each corner is translated independently of the others. Additionally, they are robust to small decoding errors because they are overdetermined; they are estimated from a neighborhood with more points than the minimum required. Figure 5 shows how to use local homography to convert the corners of the camera coordinate system to the projector coordinate system. In the following, the local homography estimation method will be described.

Suppose p is the coordinates of the corner points located in a neighborhood in the camera image. Also, suppose q decoded those corner points in the projector coordinate system. The local homography H will be obtained from the minimization of Eq. (12).

$$\widehat{H} = \arg\min_{H} \sum_{\forall p} \|q - Hp\|^2 \tag{12}$$

$H \in R^{3*3}$, $p = [x, y, 1]^T$, $q = [\text{col}, \text{row}, 1]^T$

Finally, the optimal corner point p in the camera coordinate system, located in the center of the neighborhood, will be converted to the projector coordinate system according to Eq. (13) using the homography $\widehat{H}$.

$$\overline{q} = \widehat{H}.\overline{p} \tag{13}$$

This method should be performed for all the corners on the checkerboard. Then, by knowing the location of all the points in the coordinate system of the projector, all of the internal parameters of the projector can be calculated using the Zhang method.

3.1.5 Camera–projector pair stereo calibration

Stereo calibration means finding the amount of relative rotation and translation between camera and projector. Till this stage, we have calculated internal parameters of the camera, internal parameters of the projector, the relation between camera's coordinate system and checkerboard's coordinate system and the relation between projector's coordinate system and checkerboard's coordinate system. The coordinates of the corner points are also known in the checkerboard coordinate system. On the other hand, the projection of corner points to the camera image and projector image is known. As a result, stereo calibration of a system including camera–projector pair is equal to calibration of a system including two cameras and according to this, external parameters between camera and projector are calculable. After calibration of camera–projector pair, each pair will be able to extract 3D shape of what it observes in its field of view. Thus, each pair will extract a 3D part of the screen.

3.2 Solving the problem of estimating the position of several cameras relative to each other

After each pair calibration, the calibration of the system is reduced to position estimation problem of multi-camera system. In fact, at this stage, one of the cameras is considered as reference and we calculate the relative position of each of the other cameras with respect to this camera.

3.2.1 Camera pair stereo calibration

Different algorithms for camera pair stereo calibration are presented. In this section, we will perform stereo calibration based on the information that the system provides. As you know, in a multi-projector display system, due to having a tool called projector in each pair, we will be able to project feature points on the screen and set them at camera's field of view in that pair. On the other hand, due to the stereo

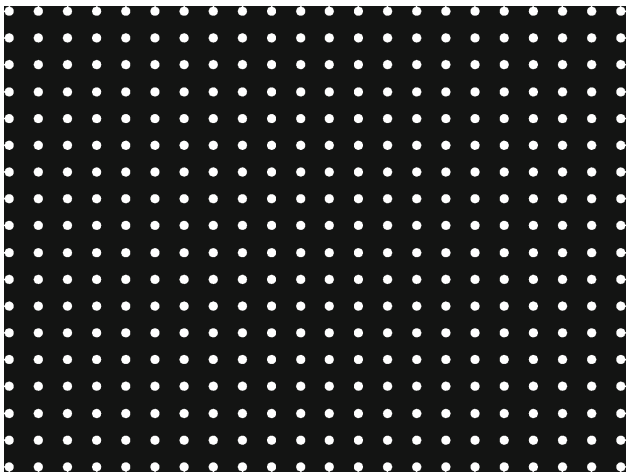


Fig. 6 Image of the projector containing $m' = 396$ feature points

vision in each pair, the depth of the points can be calculated in the coordinate system of that pair (camera or projector coordinate system). As a result, each pair can project points on the screen and calculate the 3D coordinates of the feature points. Due to the fact that the cameras overlap in adjacent pairs in their field of view, part of the points projected by the projector of a pair is also observed by adjacent cameras. So you can use these common 3D points between adjacent pairs and calibrate adjacent cameras. First, we describe the imaging scenario in a system containing n pairs and then describe how to calibrate adjacent cameras with these feature points.

3.2.1.1 Imaging To capture feature points, we have outlined the method which will be described later. Suppose we want each of the pairs to project the number of m' feature points on the screen. We create an image with the resolution of the projector and put the number of m' points uniformly on it. Figure 6 shows the image of the projector in which $m' = 396$ feature point is placed.

In this method, each projector should project the image on the screen in sequence. At the time of projection of each projector, all the cameras will capture simultaneously. This method is similar to the object (jig) motion of the calibration in the cameras' field of view which is simulated here by the projector. Finally, by examining the images provided, it will be determined that which pair is adjacent and therefore we are able to calibrate them directly. Figure 7 shows an example of images provided by system cameras. In Fig. 7a, b, projector 1 has projected feature points on the screen with the image from camera one and camera two, respectively. In Fig. 7c, d, projector 2 projected feature points on the screen and camera one and camera two provided one image, respectively.

3.2.1.2 Solving the correspondence problem Solving the correspondence problem means identifying a feature point in the camera image and projector image. In order to solve the correspondence problem, we will use the algorithm given in Reference [1] and we will explain it below.

Consider the image in Fig. 6. This image shows 396 feature points. Suppose we number the points so that the feature point placed in the position (1,1) of this 2D matrix is of number 1, the feature point placed in the position (1,2) of this 2D matrix is of numbers 2 and so on, and the feature point placed in the (2.1) position of this 2D matrix is of number 23 and so on (spatial coordinates are (row number, column number)). So we assigned a number to each of the feature points. Then, we have calculated the base-2 expression for the number of each feature point. Now, in addition to the image projection shown in Fig. 6, we have also made other images and they are also projected on the screen, in such a way that the first image only includes the feature points which in the base-2 expression, the bit is located in place one, and it has a value of 1. Similarly, the second image only includes the feature points in the base view of the two of them, the bits located in their base-2 expression have a value of one, and

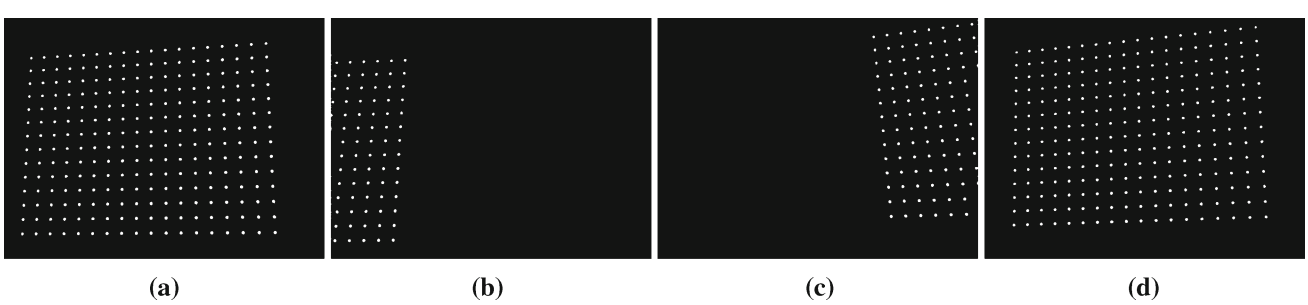


Fig. 7 Image provided by cameras of a system consisting of two pairs. **a** The number one camera has captured the feature points of the projector number one, **b** the number two camera has captured the feature points of the projector number one, **c** the number one camera has captured the feature points of the projector number two and **d** the number two camera has captured the feature points of the projector number two

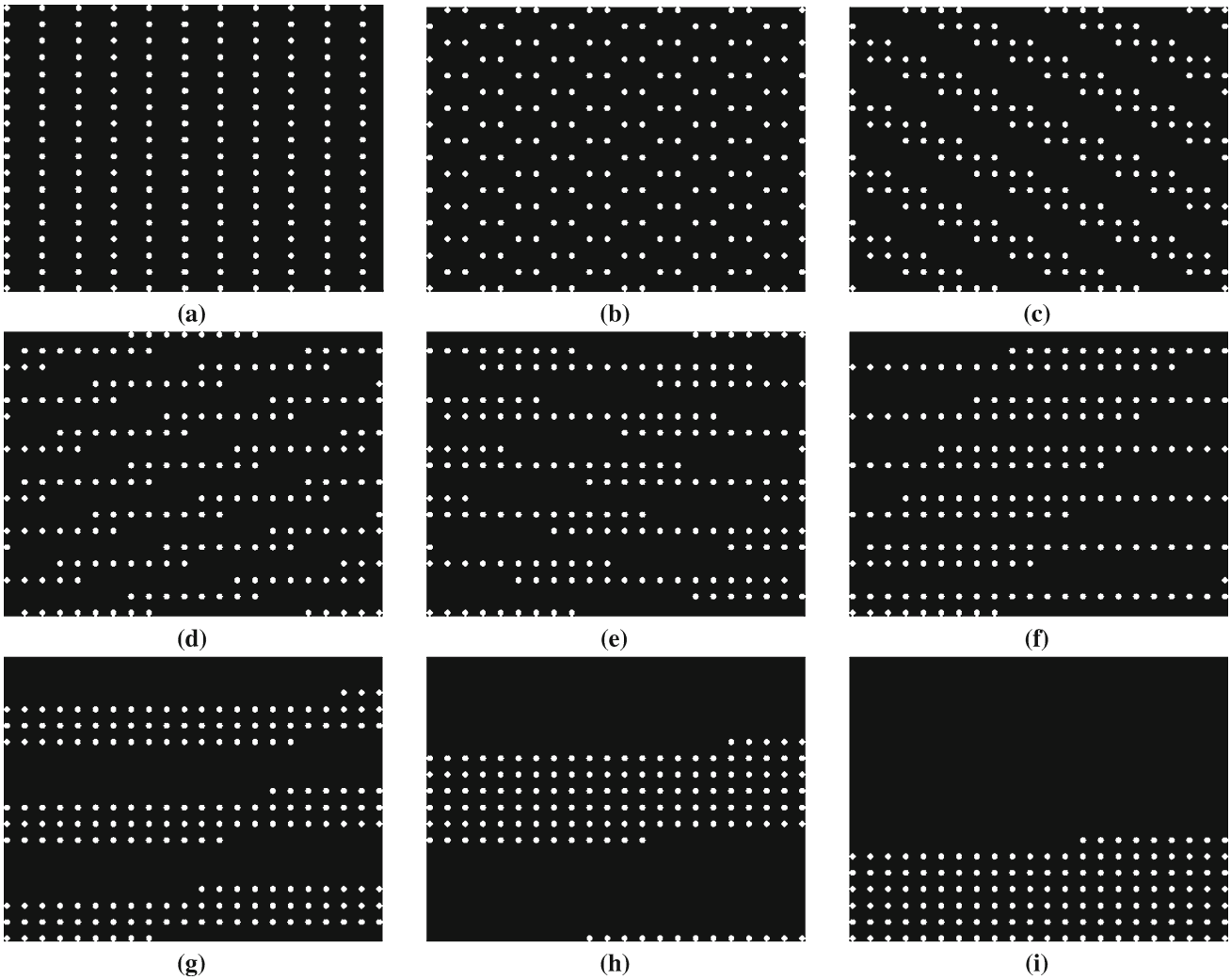


Fig. 8 Projector images for solving the correspondence problem. **a–i** Feature points with a value of one for bit number one to bit number n

so on. Thus, for a number of $396 = (110001100)_2$ points, 9 images should be projected on the screen and all the cameras should capture. At the end of the capturing stage, we will decode the points that are detected in each image. Thus, the problem of the correspondence between the feature points in the camera image and the projector image will be solved. The images to be displayed in addition to the image shown in Fig. 6, for solving the problem of correspondence, are shown in Fig. 8. Figure 8a–i shows images for the state in which bits number one to nine have a value of one, respectively.

3.2.1.3 Camera pair calibration algorithm By examining the images provided by the cameras, it will be determined which cameras have a minimal number of common feature points. In this section, we will describe an algorithm for calibrating cameras that have common feature points.

As we have already mentioned, each pair can calculate the 3D coordinates of the feature points in the coordinate system of the same pair (i.e., camera or projector). On the other hand, the adjacent pairs have special common feature points that can be used to solve the stereo calibration problem. Also, note that the correspondence problem has been solved for common feature points between adjacent camera images. As a result, the relative rotation and translation between two adjacent cameras are equal to the relative rotation and translation between the common feature points of that two cameras. Finally, by minimizing Eq. (14) for both cameras, the stereo calibration parameters will be obtained between the two cameras. The minimization of Eq. (14) has a closed-form solution method [22].

$$\min_{R,T} \sum_{i=1}^{N} \| p_i' - (Rp_i + T) \|^2 \tag{14}$$

In this equation, p_i is the 3D coordinates of the feature points in the coordinate system of a camera and p'_i is the 3D coordinates of the feature points in the coordinate system of the other camera.

3.2.2 Calculating the position of each camera relative to the reference camera

As we have already mentioned, we will not be able to directly calculate the position of each camera relative to the reference camera, because in a multi-projector display system, non-adjacent cameras do not overlap. In this section, by introducing the graph for the cameras and finding the paths in this graph from each camera to the reference camera, we will calculate the relative position of each camera relative to the reference camera. Note that the relative position of each camera has been calculated relative to adjacent cameras.

3.2.2.1 Creating adjacency graph The configuration of n cameras is shown with graph G, in which each vertex represents a camera and the presence of an edge between the two vertices indicates common space in their field of view and thus the stereo calibration between the two cameras. In this regard, we create the graph G. In this section we are looking to solve two problems. The first problem is which camera (or vertex) is considered as a reference to compute the relative position of the other cameras to this reference camera, and the second is to find the shortest path from each camera to the reference camera, because the shortest path of each camera to reference camera will have a lower error. In the next section we will provide a solution to these two problems.

3.2.2.2 Calculating the shortest path between each camera and reference camera Since we ultimately seek to express all the points in the reference coordinate system, the issue that which camera is considered as a reference will affect the error propagation from any coordinate system to the reference coordinate system. We will illustrate this with a simple example. Figure 9 shows the adjacency graph of a system containing five cameras. Due to the edges drawn on the graph, the pair of cameras that overlap in their field of view are distinguished. Suppose camera 1 has been selected as the reference camera. In this case, the total length of the shortest path between each camera and reference camera is equal to $6 = 1 + 2 + 2 + 1$. Now suppose camera 2 is selected as the reference camera. In this case, the total length of the shortest path between each camera and reference camera is equal to $5 = 2 + 1 + 1 + 1$. As you can see, the fact that which camera is considered as the reference camera will affect the length of the shortest paths from each camera to the reference camera in total. Given the importance of selecting a reference camera, it is possible in a repetitive process to count one of the vertices of the graph each time as a reference and calculate the shortest path from each vertex to the reference vertex. Finally, a vertex will be selected as reference which because of being the reference, the total length of the shortest path to the reference vertex gets the minimum. Given that the value of each edge in the adjacency graph of our system is one, the breadth-first search (BFS) algorithm can be used to find the shortest path between the two vertices of the graph. As a result, in a repetitive process, each time we consider one of the vertices of the graph as the reference and then, we find the shortest path between each vertex and the reference vertex. At the end of this process, if for ith vertex, total length of the shortest path from every vertex to the reference vertex gets the minimum, then we select ith vertex as reference and store shortest paths from each vertex to the reference vertex.

3.2.3 Global optimization and bundle adjustment problem

The method used to calculate the position of each camera relative to the reference camera causes error propagation from each camera to the reference camera. Note that in order to express points in the reference camera's coordinate system, some rotation and translation matrices, which exist in the path of every camera to reference camera, should be multiplied together. Since calculating the rotation and translation values between each adjacent pair has errors, the multiplication of rotation and transition matrices will cause an error propagation. We will have global optimization phase and bundle adjustment problem to optimize the solution till this stage. The bundle adjustment problem is usually a final step in 3D reconstruction algorithms. In this step, by writing an energy function, we seek to optimize all calibration parameters. We will write down the corresponding energy function and then describe its components.

Consider a system including N cameras and M feature points. Since all feature points are not necessarily observed by all cameras, a mask matrix called $W_{M\times N}$ is introduced, in which the matrix above contains only zero and one numbers. If the point m is observed on camera n, w_{mn} is equal to one; otherwise, its value is zero. Suppose we have $N \geq 2$ cameras. Assume that the point m is shown with $\mathbf{x}_m = [x_m, y_m, z_m]^T$, the set of all three-dimensional points with $X = \mathbf{x}_{m\,(m=1,\ldots,M)}$, the translation vector of the camera n in the reference coordinate system with $\mathbf{t}_n = [t_{n,x}, t_{n,y}, t_{n,z}]^T$, the set of all translation vectors in the reference coordinate system with $T = \mathbf{t}_{n\,(n=1,\ldots,N)}$, rotation angles of nth camera with $\phi_n = [\varphi_{n,x}, \varphi_{n,y}, \varphi_{n,z}]^T$ and the set of all rotation angles with $\Phi = \phi_{n\,(n=1,\ldots,N)}$. Also, suppose each row shows the rotating matrix of the nth camera, $R(\phi_n)$, with $\mathbf{r}_{ni}$ for $i = 1, 2, 3$. To define the energy function of the global optimization problem and the bundle adjustment problem, we first define the mth point mapping function of the three-

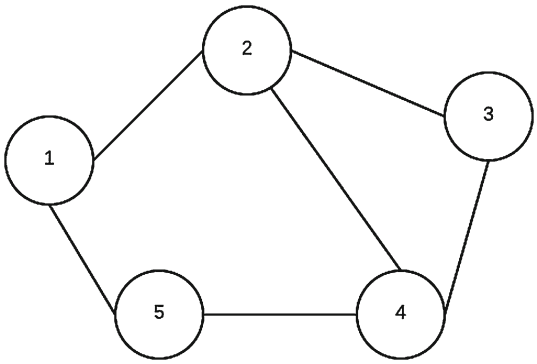


Fig. 9 Adjacency graph of a system consisting of five cameras. More information can be found in [26] and [27]

dimensional points by the nth camera with Eq. (15).

$$u_{mn} = u_n^0 + f_n \frac{\mathbf{r}_{n1}.\mathbf{x}_m + t_{nx}}{\mathbf{r}_{n3}.\mathbf{x}_m + t_{nz}},\ v_{mn} = v_n^0 + f_n \frac{\mathbf{r}_{n2}.\mathbf{x}_m + t_{ny}}{\mathbf{r}_{n3}.\mathbf{x}_m + t_{nz}} \tag{15}$$

In this relation, f_n and $[u_n^0 v_n^0]$ are the focal length and principal point of the nth camera, respectively. If we define $[u_{mn}^0 v_{mn}^0]$ as the coordinates of the mth point image in the nth camera, the energy function can be defined as the bundle adjustment problem in Eq. (16).

$$\min P = \sum_{n=1}^{N} \sum_{m=1}^{M} w_{mn} [(u_{mn} - u_{mn}^0)^2 + (v_{mn} - v_{mn}^0)^2] \tag{16}$$

In this relation, w_{mn} are mask matrix elements. Therefore, the goal of the above optimization problem is to obtain the calibrated sets X, T and Φ.

The optimization problem has been optimized using the Levenberg–Marquardt classical algorithm; more details on this algorithm are presented in Reference [23].

3.3 Creating image for projectors

After calibrating the system and, as a result, calculating the calibration parameters, we have to set up the inputs of the projectors so that a seamless and unified image is displayed on the screen. Since correspondence problem is resolved between the image of the camera and the image of the projector, and given the calibration parameters of each pair, it is possible to extract the 3D shape of screen using triangulation, but this is a time-consuming process. Note that when solving the problem of estimating the position of a number of cameras together, we calculated the 3D coordinates of some feature points in the camera coordinate system of each pair. We also calculated the position of all system cameras with respect to the reference camera. As a result, we will have point clouds creating the screen in the coordinate system of the reference camera.

In the next step, we would like to wallpaper a single image on the screen. A simple method for doing this is to create a one-to-one correspondence between the image pixels and the point clouds of the screen. If the point clouds of the screen do not have the proper resolution for generating this one-to-one correspondence, then using interpolation algorithms, you can get new points and add to the points of the screen. Thus, we assign a color to each of the points. Finally, the color of each point on the screen will be assigned to the corresponding pixel in the image of each projector.

It should be noted that the process of calibrating the system and the process of creating the image will be performed offline and online, respectively. Eventually, after the projectors' images are initialized, each will project its share on screen.

4 Experimental results

In this section, we will evaluate the proposed system. This assessment will be carried out in a variety of ways. The first evaluation we have for the system is the local alignment error. In the next section, we will review the error of the global alignment to check the error propagation. The third assessment we considered for the system relates to the error of the problem of estimating the position of a number of cameras relative to each other. Also, in this section we will examine the effect of noise on the system. In the end, we will examine the level of optimum overlapping for projectors in a multi-projector system.

The proposed system is implemented using the Matlab R2016a software. As we will discuss in the evaluation section, we will do some part of the evaluation based on actual images, and another part based on the camera and projector simulator, which is written for the desired number of pairs. With the help of this simulator, we can simulate the mode of projection of n pairs by assuming an arbitrary uneven screen. We will give further details on the simulator in the relevant section.

The webcams used in the system to evaluate the proposed system are of Logitech, C270 model, and have a resolution of 1280×960. All the webcam settings are in manual mode. The projectors used in the system are Epson EB-S18 model with a resolution of 1024×768. Also, the auto-correct setting of the rectangular image has been disabled in these projectors so that no geometric correction is made on the projector image. The processing time for calibration of a system consists of three pairs with an Intel core i5-4670 3.40GHZ personal computer and 8GB RAM is 20 seconds.

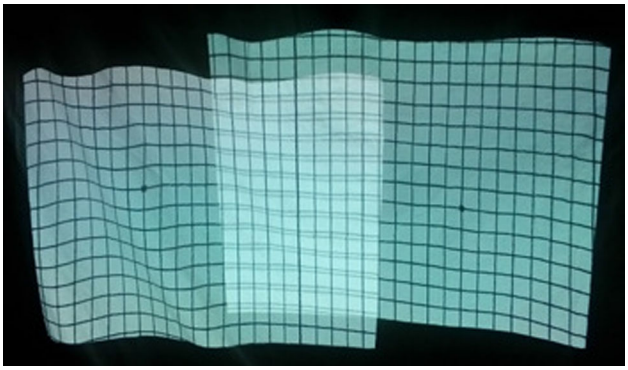


Fig. 10 An example of a lack of alignment in the overlapping area

4.1 System evaluation criteria

Most articles presented in the field of multi-projector displays do not have a benchmark, because there is no precise evaluation criterion in this regard. In some articles that calibrate the camera, the error of the camera calibration phase is announced as the error of the system, but because the error of this phase does not describe the output of a multi-projector display visually, this criterion also does not provide precise evaluation of the system error. In References [7] and [24], a system is considered for evaluation and we will use this criterion to compare our proposed system.

In this section we will present four tests to evaluate the proposed system. The first experiment is the local alignment error for adjacent pairs, the second experiment is the global alignment error for the problem of error propagation, the third experiment is the error of the problem of estimating the position of a number of cameras together, and the fourth experiment is to consider the effect of the overlapping of adjacent projectors. Also, in the third experiment, the effect of noise on the system will be investigated.

4.2 Local alignment accuracy

The most important factor that eliminates the idea of having a single integrated display is the mismatch of images in the overlapping area of projectors. If the images produced by the two neighboring projectors do not have a proper alignment, the image in that area will be duplicated. An example of a lack of proper alignment in the overlapping area is shown in Fig. 10.

Suppose Ω is the set of all feature points and Φ is the set of all projectors. We define the local alignment error criterion in Eq. (17).

$$E = \sum_{\forall p \in \Omega} \sum_{\forall (i,j) \in \Omega * \Omega} I(i,p).I(j,p).\|p_i - p_j\|^2 \qquad (17)$$

Table 1 Comparison of the proposed system with other systems based on the global alignment accuracy criterion (pixel)

Method	Number of camera view	Error
[6]	152	1.35
[25]	1	1.73
[7,24] (cam-all)	1	1.19
[7,24] (cam-2*2)	15	0.55
Proposed method	15	0.43

In this equation, p_i is the coordinate of the image of the feature point p in the camera of the ith pair and $I(.)$ will be zero or one such that, if the feature point p is observed in the image of the corresponding projector, $I(.) = 1$, and otherwise, $I(.) = 0$. This relationship measures the same non-conformance that the human eye is very sensitive to.

Measuring this criterion is time-consuming. To do this, we present an automated algorithm to measure the local alignment error, which we will outline below. Note that this experiment was performed using real images for two pairs. In this method, we first label an image of 14*18 feature points on the screen. Each of the feature points in this image is assigned a number and using the algorithm described in Section 3.2.1.2. We solve the correspondence problem for the feature points in this image and the images of each of the two cameras. Obviously, only a few of these points are in the overlapping area of the two projectors. Using the camera pair number one, we capture the feature points of the projector number one, and then the feature points of the projector number two. After extracting the corresponding points from the camera image, we calculate the Euclidean distance of these points. The results of this measurement are recorded in Table 1. In this table, the phrase cam-2*2 means the observing of a 2*2 array of projectors by each camera and cam-all means the observing of all projectors by each camera. As you can see, the error of the proposed system is considered to be better in all configurations.

4.3 Global alignment error

As we mentioned in the previous section, the human eye is very sensitive to the local alignment error criterion. In the face of large displays that the system has more than one camera, in order to express points in the reference coordinate system, the transformation matrices in the path of each camera to the reference camera are multiplied, which will cause the error to be propagated from each camera to the reference camera. The local alignment error criterion in the previous section does not address the problem of error propagation, because local alignment error is measured only between two neighboring pairs. In calculating the global alignment error, we consider the number of pairs to be more than one and we will test using

Table 2 Comparison of the proposed system with other systems based on the global alignment accuracy criterion (pixel)

Method	Number of camera view	Error
[6]	152	–
[25]	1	1.5
[7,24] (cam-all)	1	1.5
[7,24] (cam-2*2)	15	1.8
Proposed method	15	0.85

the simulator. The following is a description of the simulator, and then, we will present the results.

The camera and projector simulator is a software that simulates the performance of a system that includes a number of camera and projectors. In this simulator we will also have pairs of cameras and projector with the internal and external parameters of these two as inputs to the algorithm. Also, the position of the remaining cameras will be sequentially arranged according to the position of the last camera and considering that projectors are overlapping in adjacent pairs. Then, we can turn on a number of projectors and capture an image from one of the cameras.

Using the simulator above, a display system containing 15 pairs is evaluated. Results are presented in Table 2. As you can see, the overall alignment error for the proposed system is higher than the rest of the systems.

4.4 Effect of noise on the proposed system error

In this section, we will consider an experiment to examine the effect of noise on the proposed system. In this test, the position of all cameras is generated randomly. When it comes to the position of the cameras, it is considered that the projector of each pair has overlap with at least one of the projector of other pairs. Figure 11 is an example of an image of one of the cameras. In this image, the projector number one projects an image that includes a few feature points on a sinusoid screen. Then, the camera number two started capturing.

Suppose that the calibration process for the random data generated is related to four cameras. Given the actual position of the cameras in the simulator, difference of calculated position and real position in 3D space is calculated as root mean square error.

Figure 12 shows the results of noise effects on the error of the problem of estimating the position of several cameras relative to each other. As you can see, the error of the problem of estimating the position of several cameras relative to each other is almost identical and acceptable until the noise with a variance of less than 0.6. The added noise levels with a variance of more than 0.6 magnitudes have considerably increased. The reason for this is that the initial solution is not good enough. In fact, due to the non-convexity of the introduced energy function and not having a good initial solution and close enough to the optimal point, there is no guarantee of finding the optimal global point and the algorithm will be caught in the local optimal point.

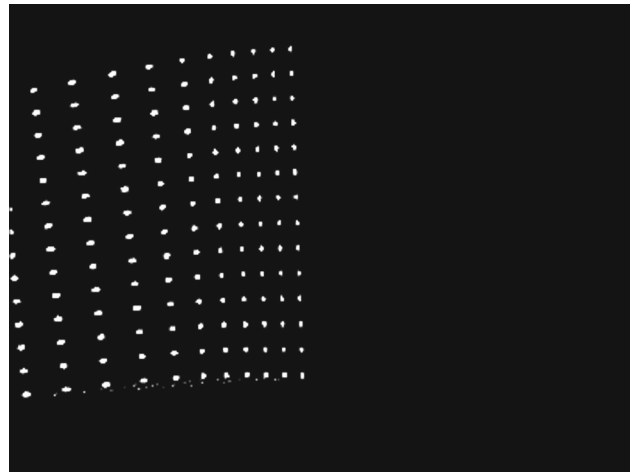


Fig. 11 Projection of projector of pair number one and camera capturing from pair two

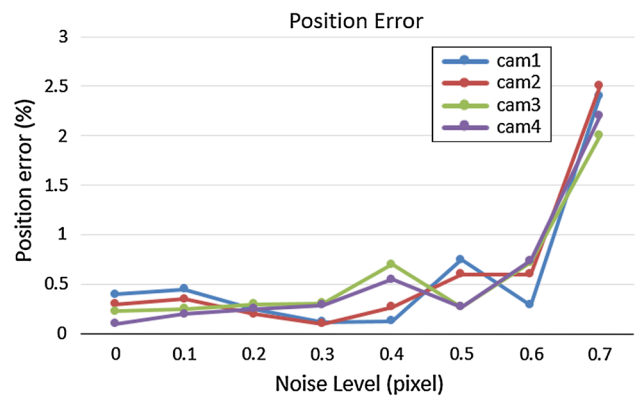


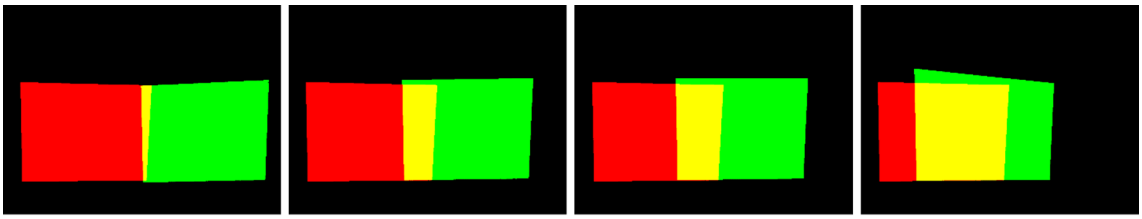
Fig. 12 Effect of noise on the error of the problem of estimating the position of a number of cameras relative to each other

4.5 The effect of projector overlapping on local alignment error

As we have already mentioned, projectors are overlapping in multi-projector display systems. This configuration is intended to prevent seams in the boundaries of projectors.

In this section, an experiment was conducted to obtain the optimum level of overlapping area for the proposed system. This experiment is based on two pairs. The way to calculate the overlap between two projectors is measured using a camera.

If n_1 is the number of pixels of the projector number one, n_2 is the number of pixels of the projector number two and m is the number of pixels located in the overlap area, the overlap percentage between the two adjacent projector can



(a) (b) (c) (d)

Fig. 13 Different overlap percentages for two projectors. **a** 3%, **b** 13%, **c** 20%, **d** 49%

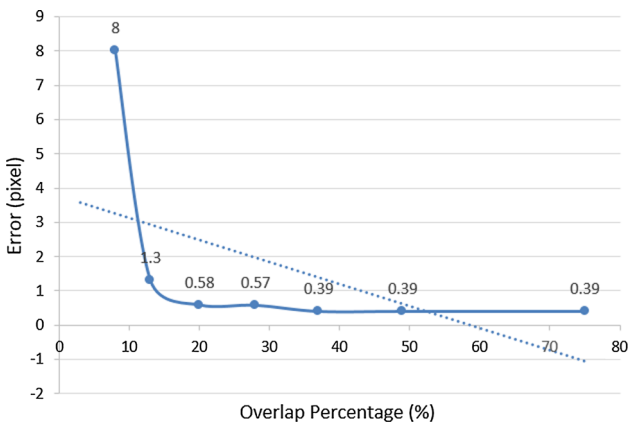


Fig. 14 Percent overlay graph relative to local alignment error

be calculated according to Eq. (18).

$$\text{overlap percentage} = \frac{m}{n_1 + n_2 - m} * 100 \tag{18}$$

Figure 13a–d shows images relating to the two projectors that were captured by a camera. These images are provided in color after the thresholding on camera images and calculating the masks of each projector to see the overlapping area. In Fig. 13a–d, overlapping percentage is 3%, 13%, 20% and 49%, respectively, according to Eq. (18).

In Fig. 14 graphs are plotted to review the results of this test. In this diagram, the horizontal axis shows the percentage of overlap between the two projectors and the vertical axis is the local alignment error. It should be noted that in the case where the overlap rate between the two projectors is 3%, the proposed algorithm has a great deal of error. In fact, in this case, the proposed algorithm failed to find the relative position of the camera number one relative to the camera number two. Also, 20% seems to be optimal for overlapping percent, as it is less than one pixel error and the overlapping ratio of projectors is lower than nice works such as [24]. On the other hand, the overlap rate of more than 40% did not have much effect on error. Note that the diagram shown in Fig. 14 can be considered as a good measure for buyers of a multi-projector display system. The overlap percentage will require more pairs to cover the entire screen, and a lower overlap percentage can reduce the number of system pairs if the alignment error is reduced.

5 Conclusion

A new method for calibrating a multi-projector display is presented in this paper. Viewpoint independence and having favorite display are the challenges in the field of multi-projector displays (as it is mentioned before). We have designed the multi-projector displays from the beginning in this paper because the other methods are not capable of presenting an approach to solve the challenges above. In this method, we designed the multi-projector displays as an optimization problem so all calibration parameters will be calculated after solving this optimization problem. Considering camera and projector as a pair not only offers the possibility of stereo vision to every pairs, but also it is in the same direction of companies which produce projectors in the world (new projectors which are sold recently in market have an internal camera). Also, proposed method is designed in a way that can be used in projection on any uneven surfaces in addition to the projection on a flat screen. In the beginning, the object should be covered by a white cloth, and different camera-projector pairs have to be placed around the object. The 3D shape of the object can be extracted by performing the calibration operation. This extraction is able to be projected. According to what has been described in Sect. 1, none of previous works done in the field of multi-projector displays were defined generally to this extent. Consider that our proposed system covers a huge area of applicable softwares which can be defined for a display. In presenting this system, we did as much as effort we could to make no need of user monitoring in setting up a multi-projector display and make the calibration operation done in a totally automatic way. Also, according to presented results in Sect. 4, the alignment accuracy of the proposed system is less than one pixel and is better than available systems.

We run the system on a spherical surface with three pairs and on an arbitrary uneven surface with two pairs, and the results are available through the links below:

1. http://mvlab.um.ac.ir/ then go to the project section.
2. https://www.youtube.com/watch?v=blzAmpoq1_4.

References

1. Majumder, A., Browne, C., Brown, M.S.: Practical Multi-projector Display Design. CRC Press, Boca Raton (2007)
2. Raskar, R., Welch, G., Cutts, M., Lake, A., Stesin, L., Fuchs, H.: The office of the future: a unified approach to image-based modeling and spatially immersive displays. In: Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques, pp. 179–188 (1998)
3. Raskar, R., Brown, M.S., Yang, R., Chen, W.C., Welch, G., Towles, H., Seales, B., Fuchs, H.: Multi-projector displays using camera-based registration. In: Proceedings of the Conference on Visualization'99: Celebrating 10 Years, pp. 161–168 (1999)
4. Brown, M.S., Seales, W.B.: A practical and flexible tiled display system. In: Proceedings of 10th Pacific Conference on Computer Graphics and Applications, pp. 194–203 (2002)
5. Ahmed, A., Hafiz, R., Khan, M.M., Cho, Y., Cha, J.: Geometric correction for uneven quadric projection surfaces using recursive subdivision of Bézier patches. ETRI J. **35**(6), 1115–1125 (2013)
6. Chen, Y., Clark, D.W., Finkelstein, A., Housel, T.C., Li, K.: Automatic alignment of high-resolution multi-projector display using an un-calibrated camera. In: Proceedings of the Conference on Visualization'00, pp. 125–130 (2000)
7. Chen, H., Sukthankar, R., Wallace, G., Li, K.: Scalable alignment of large-format multi-projector displays using camera homography trees. In: Proceedings of the Conference on Visualization'02, pp. 339–346 (2002)
8. Hereld, M., Judson, I.R., Stevens, R.: Dottytoto: a measurement engine for aligning multi-projector display systems. Argonne National Laboratory preprint ANL, MCS-P958-0502 (2002)
9. Bhasker, E., Juang, R., Majumder, A.: Registration techniques for using imperfect and partially calibrated devices in planar multi-projector displays. IEEE Trans. Vis. Comput. Gr. **13**(6), 1368–1375 (2007)
10. Raskar, R., Van Baar, J., Willwacher, T., Rao, S.: Quadric transfer for immersive curved screen displays. Comput. Gr. Forum **23**(3), 451–460 (2004)
11. Harville, M., Culbertson, B., Sobel, I., Gelb, D., Fitzhugh, A., Tanguay, D.: Practical methods for geometric and photometric correction of tiled projector. In: Conference on Computer Vision and Pattern Recognition Workshop, CVPRW'06, pp. 5–5 (2006)
12. Sajadi, B., Majumder, A.: Markerless view-independent registration of multiple distorted projectors on extruded surfaces using an uncalibrated camera. IEEE Trans. Vis. Comput. Gr. **15**(6), 1307–1316 (2009)
13. Chuang, Y.M., Hsu, S.P., Huang, K.C., Chang, Y.C., Ruan, S.H.: Implementation of image warping with application to projection onto a curved display. 5th IEEE Conference on Industrial Electronics and Applications (ICIEA), pp. 404–408 (2010)
14. May, B.B., Cahill, N.D., Rosen, M.R.: Calibration of a multi-projector system for display on a cylindrical surface. In: Image Processing Workshop (WNYIPW), Western New York, pp. 6–9 (2010)
15. Heinz, M., Brunnett, G.: Optimized GPU-based post-processing for stereoscopic multi-projector display systems. Virtual Real. **23**, 45–60 (2019)
16. Portalés, C., Orduña, J.M., Morillo, P., Gimeno, J.: An efficient projector calibration method for projecting virtual reality on cylindrical surfaces. Multimed. Tools Appl. **78**, 1457–1471 (2019)
17. Han, Y., Wang, M., Liu, X., Wang, R.: A method for multi-projector seamless tiled display system. In: 2nd International Conference on Signal Processing and Integrated Networks (SPIN), pp. 837–842 (2015)
18. Teng, WANG, WANG, Yi-gang: A Viewpoint Mapping Geometric Correction Technique for Image Display of Curved Screen, DEStech Transactions on Engineering and Technology Research, apop (2017)
19. Moreno, D., Taubin, G.: Simple, accurate, and robust projector-camera calibration. In: Second International Conference on 3D Imaging, Modeling, Processing, Visualization and Transmission (3DIMPVT), pp. 464–471 (2012)
20. Li, T., Hu, F., Geng, Z.: Geometric calibration of a camera-projector 3D imaging system. In: Proceedings of the 10th International Conference on Virtual Reality Continuum and Its Applications in Industry (ACM), pp. 187–194 (2012)
21. Zhang, Z.: A flexible new technique for camera calibration. IEEE Trans. Pattern Anal. Mach. Intell. **22**, 1330–1334 (2000)
22. Aftra, K.S., Huang, T.S., Blostein, S.D.: Least-squares fitting of two 3-D point sets. IEEE Trans. Pattern Anal. Mach. Intell **5**, 698–700 (1987)
23. Moré, J.J.: Numerical Analysis, pp. 105–116. Springer, New York (1978)
24. Bhasker, E.S., Sinha, P., Majumder, A.: Asynchronous distributed calibration for scalable and reconfigurable multi-projector displays. IEEE Trans. Vis. Comput. Gr. **12**, 1101–1108 (2006)
25. Yang, R., Gotz, D., Hensley, J., Towles, H., Brown, M.S.: Pixelflex: a reconfigurable multi-projector display system. In: Proceedings of the Conference on Visualization'01, pp. 167–174 (2001)
26. Yan, C., Li, L., Zhang, C., Liu, B., Zhang, Y., Dai, Q.: Cross-modality bridging and knowledge transferring for image understanding. IEEE Trans. Multimed. (2019). https://ieeexplore.ieee.org/abstract/document/8662712
27. Yan, C., Xie, H., Chen, J., Zha, Z., Hao, X., Zhang, Y., Dai, Q.: A fast uyghur text detector for complex background images. IEEE Trans. Multimed. **20**(12), 3389–3398 (2018)

Shahaboddin Askarian Bajestani received the bachelors degree in computer engineering from Sadjad University of Technology, Mashhad, Iran, in 2013 and the M.Sc. degree in artificial intelligence from Ferdowsi University of Mashhad, Iran, in 2016. He is currently working on designing and implementing camera–projector-based applications. His research interests include image processing and machine vision.

Hamidreza Pourreza received the M.Sc. degree in electrical engineering and the Ph.D. degree in computer engineering from Amirkabir University of Technology, Tehran, Iran, in 1993 and 2002, respectively. He is a professor with the Department of Computer Engineering, Ferdowsi University of Mashhad, Iran. His research interests include image processing and machine vision. He is a senior member of IEEE and one of the founders and active member of MVLab (mvlab.um.ac.ir).

Shayan Nalbandian received his bachelors degree in information technology engineering from Islamic Azad University, Mashhad, Iran, in 2016. His current research interest includes Internet of Things and Cloud Computing.