

## Correction of Image Deformation from Lens Distortion Using Bezier Patches

ARDESHIR GOSHTASBY

*Department of Computer Science, University of Kentucky, Lexington, Kentucky 40506 0027*

Received February 19, 1987; revised January 19, 1989

In many image analysis applications such as depth perception and motion analysis, availability of accurate image positional information is a necessity. Distortion in the lens, however, moves the image points locally and causes error in the position of image points. In this paper, a decalibration process is described which corrects image deformation from lens distortion. In this process lens distortion is first determined in terms of Bezier patches and then the patches are used to correct image deformation. © 1989 Academic Press, Inc.

### 1. INTRODUCTION

Most image analysis systems are based on ideal cameras, functioning like a pin-hole camera. A pin-hole camera, however, cannot be realized in practice. Distortion from the lens deforms the image, so it no longer represents the projection of points in the scene.

A lens distortion may be classified as radial or tangential [1]. A radial distortion causes image points to be displaced from their ideal positions toward or away from the image center while a tangential distortion causes image points to be displaced in the direction perpendicular to the line connecting them to the image center. A straight line passing through the image center will stay straight (although it might stretch or shrink) under a radial distortion, but the same straight line will become curved under a tangential distortion. Other lines in the image will be curved under both radial and tangential distortions. Radial distortion results from imperfect lens design, while tangential distortion results from decentering the lens elements when manufacturing the lens.

In this paper, a decalibration process is described that can correct image deformation obtained from lens distortion. Decalibration is the processing performed to correct a deformed image; it transforms the deformed image into an ideal one which has no distortion. To obtain such a transformation, a uniform grid is placed in front of the camera and its image is produced. The original grid is taken as the ideal image, and the obtained image is taken as its deformed image. The objective in the decalibration process is to transform the deformed image to resemble the ideal image.

Green *et al.* [2] used local affine transformations to map grid elements in the deformed image to grid elements in the ideal image. This technique, although providing continuous mapping from one grid element to the next, does not provide a smooth sampling between the grid elements because the transformation function used is linear. Yokobori, Yeh, and Rosenfeld [3] used global polynomials of degree three with least-squares to map deformed images to ideal images. The distortion

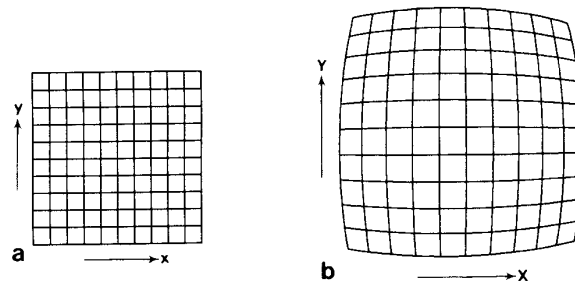


FIG. 1. (a) and (b) are, correspondingly, the ideal and the deformed images used in the decalibration process.

model used by this technique assumes that the entire center row and the entire center column in an image are not distorted. This assumption, however, is too restrictive, since it cannot represent either radial or tangential distortion.

In this paper, Bezier patches [4, pp. 215–219] will be used to describe lens distortion. A local lens distortion has a global effect on the obtained images, but its effect is vanishingly small on points far away. This behavior can be described by Bezier patches very well. A Bezier patch is described globally but remains sensitive to local data. This behavior is in contrast to least-squares techniques for fitting polynomials to local data, where a local distortion is averaged out equally all over the image.

A Bezier patch is described in terms of the vertices of its characteristic polyhedron, known as the control vertices. In this paper, an algorithm is given that can determine the control vertices of a Bezier patch fitting a uniform grid of points. A Bezier patch determined in this manner describes the behavior of a lens and can be used to correct images obtained by the lens.

In the following, in Section 2, the decalibration process is formulated in terms of Bezier patches, and in Section 3, the control vertices of a Bezier patch fitting a uniform grid of points is determined. Then in Section 4, the proposed decalibration process is applied on an off-the-shelf lens to determine its distortion and to correct images obtained by the lens.

## 2. FORMULATION OF THE DECALIBRATION PROCESS

Figure 1a shows a uniform rectangular grid. Suppose we place this grid in front of the camera in such a way that the optical axis of the camera is normal to the grid plane and passes through the center of the grid. It is not required to align the grid lines with the  $x$  and  $y$  axes because we are going to use a representation which uses an arbitrary coordinate system. Let Fig. 1b show an image obtained from the grid. Due to lens distortion, the obtained image is deformed. We need a transformation function that can transform the deformed image back to the ideal image. Such a transformation function describes the behavior of the lens and can be used to correct images obtained by the lens.

Using the correspondence between grid points in Figs. 1a and b, we determine the required transformation function as follows. Let points  $(x_{ij}, y_{ij})$  and  $(X_{ij}, Y_{ij})$  show points obtained by intersection of the  $i$ th horizontal line and the  $j$ th vertical line in

Figs. 1a and b, respectively. We desire two functions  $f$  and  $g$  that satisfy

$$\left. \begin{aligned} X_{ij} &= f(x_{ij}, y_{ij}) \\ Y_{ij} &= g(x_{ij}, y_{ij}) \end{aligned} \right\} \quad i = 0, \dots, m; \quad j = 0, \dots, n, \quad (1)$$

where  $(m + 1)$  and  $(n + 1)$  are the number of horizontal and vertical grid lines in the images, respectively. The functions  $f$  and  $g$  map grid points in the ideal image to grid points in the deformed image. Other points in the images are mapped by interpolation.

To determine  $f$ , we note that  $(x_{ij}, y_{ij}, X_{ij}) \quad i = 0, \dots, m, \quad j = 0, \dots, n$  can be considered a set of 3D points, and  $f$  can be considered a surface that passes through the points. Therefore, to determine  $f$ , we need a smooth surface that passes through the points. There are many surfaces, however, that can pass through the points. Since gridded data is available, we use Bezier patches [4, pp. 215–219] for this purpose. A Bezier patch although using the whole data, it is locally sensitive and models lens distortion very well.

In the next section, a procedure to determine function  $f$  as a Bezier patch is described. Function  $g$  can be determined similarly.

### 3. FITTING BEZIER PATCHES TO GRIDDED DATA

The power of Bezier patches in modeling 3D shapes is well known [5]. By changing the position of control vertices of a Bezier patch appropriately, it is possible to change the shape of the surface to a desired shape. In this paper, however, we do not use Bezier patches to design 3D shapes but rather to interpolate data. A Bezier patch is described in terms of the control vertices of its characteristic polyhedron [4, p. 215]. We would like to determine the control vertices of a Bezier patch that would fit a grid of points.

To determine the Bezier patch, we note that the curves bounding the patch are Bezier curve segments that pass through the bounding grid points [4, p. 217]. In the following, we first determine the bounding curves of the patch by determining the control vertices of Bezier curve segments that pass through the boundary grid points.

#### 3.1. Determining the Control Vertices of a Bezier Curve Segment Fitting a Set of Points

Given a set of  $(n + 1)$  points, we would like to determine a Bezier curve segment that passes through the points (see Fig. 2). A Bezier curve segment is determined in terms of the control vertices of its characteristic polygon. Suppose the control vertices of the Bezier curve that passes through the given points are  $V_i, i = 0, \dots, n$ . Then, the parametric representation of such a curve is given by [4, p. 113]

$$\mathbf{P}(u) = \sum_{i=0}^n V_i B_{i,n}(u) \quad u \in [0, 1], \quad (2)$$

where  $B_{i,n}(u) = C(n, i)u^i(1 - u)^{n-i}$  are the blending functions and  $C(n, i) = n!/i!(n - i)!$  are the binomial coefficients.

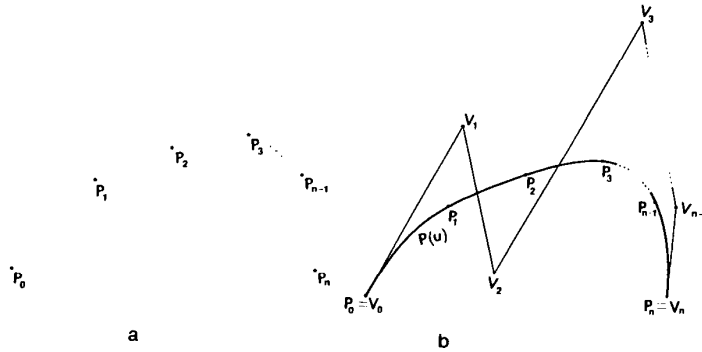


FIG. 2. Fitting a Bezier curve segment to a set of data points: (a) the given data points; (b) the Bezier curve segment passing through the points and its characteristic polygon.

Let us denote the points on the curve at  $u = j/n$  by  $P_j$  for  $j = 0, \dots, n$ . Then from relation (2) we have

$$P_j = \sum_{i=0}^n V_i B_{i,n}(j/n) \quad j = 0, \dots, n. \tag{3}$$

Now, assuming that the given points through which the curve should pass, are  $P_j$ ,  $j = 0, \dots, n$ , we determine the control vertices of such a Bezier curve segment as follows. Since a Bezier curve segment passes through the two end control vertices then  $P_0 = V_0$ ,  $P_n = V_n$ , and relation (3) may be rewritten as

$$P_j - P_0 B_{0,n}(j/n) - P_n B_{n,n}(j/n) = \sum_{i=1}^{n-1} V_i B_{i,n}(j/n) \quad j = 1, \dots, n-1. \tag{4}$$

Relation (4) shows a system of  $(n - 1)$  linear equations which can be solved to determine the control vertices  $V_i$ ,  $i = 1, \dots, n - 1$ . Knowing the boundary control vertices, next we determine the remaining control vertices of a Bezier patch to fit a grid of points.

### 3.2. Determining the Control Vertices of a Bezier Patch Fitting a Grid of Points

Given a grid of  $(m + 1) \times (n + 1)$  points, we would like to determine a Bezier patch that would fit the points. Assuming such a Bezier patch has control vertices  $V_{ij}$ ,  $i = 0, \dots, m$ ,  $j = 0, \dots, n$ , we can write the equation of the patch in parameters  $u$  and  $v$  as follows [4, p. 215]:

$$P(u, v) = \sum_{i=0}^m \sum_{j=0}^n V_{ij} B_{i,m}(u) B_{j,n}(v) \quad u, v \in [0, 1]. \tag{5}$$

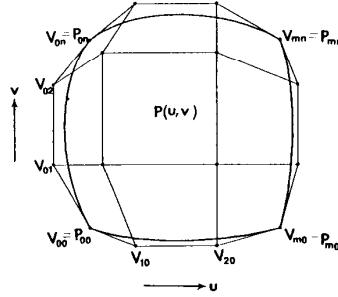


FIG. 3. A Bezier patch and its control vertices.

Let us denote the point on the patch at  $u = i/m$  and  $v = j/n$  by  $\mathbf{P}_{ij}$ , then

$$\mathbf{P}_{ij} = \sum_{i=0}^m \sum_{j=0}^n \mathbf{V}_{ij} B_{i,m}(i/m) B_{j,n}(j/n) \quad i = 0, \dots, m, j = 0, \dots, n. \quad (6)$$

We assume that  $\mathbf{P}_{ij}$ ,  $i = 0, \dots, m$ ,  $j = 0, \dots, n$  are the given points, and by an inverse algorithm determine the control vertices  $\mathbf{V}_{ij}$ ,  $i = 0, \dots, m$ ,  $j = 0, \dots, n$  of such a patch.

Since a Bezier patch passes through its four corner control vertices (see Fig. 3), then

$$\mathbf{V}_{00} = \mathbf{P}_{00}, \quad \mathbf{V}_{0n} = \mathbf{P}_{0n}, \quad \mathbf{V}_{m0} = \mathbf{P}_{m0}, \quad \mathbf{V}_{mn} = \mathbf{P}_{mn}. \quad (7)$$

Also, since the bounding control vertices of a Bezier patch are the same as the control vertices of Bezier curve segments bounding the Bezier patch, we can determine the bounding control vertices of the patch using the procedure in Section 3.1. The bounding control vertices that are determined in this manner are

$$\mathbf{V}_{mj}, \mathbf{V}_{0j}, j = 1, \dots, n-1; \quad \mathbf{V}_{i0}, \mathbf{V}_{in}, i = 1, \dots, m-1. \quad (8)$$

Substituting (7) and (8) into (6) we get

$$\begin{aligned} \mathbf{P}_{ij} &= [\mathbf{V}_{00} B_{0,m}(i/m) B_{0,n}(j/n) + \mathbf{V}_{0n} B_{0,m}(i/m) B_{n,n}(j/n) \\ &+ \mathbf{V}_{m0} B_{m,m}(i/m) B_{0,n}(j/n) + \mathbf{V}_{mn} B_{m,m}(i/m) B_{n,n}(j/n)] \\ &- \left[ \sum_{l=1}^{n-1} \mathbf{V}_{0l} B_{0,m}(i/m) B_{l,n}(j/n) + \sum_{l=1}^{n-1} \mathbf{V}_{ml} B_{m,m}(i/m) B_{l,n}(j/n) \right. \\ &+ \left. \sum_{k=1}^{m-1} \mathbf{V}_{k0} B_{k,m}(j/n) + \sum_{k=1}^{m-1} \mathbf{V}_{kn} B_{k,m}(i/m) B_{n,n}(j/n) \right] \\ &= \sum_{k=1}^{m-1} \sum_{l=1}^{n-1} \mathbf{V}_{kl} B_{k,m}(i/m) B_{l,n}(j/n) \\ & \quad i = 1, \dots, m-1, j = 1, \dots, n-1. \quad (9) \end{aligned}$$

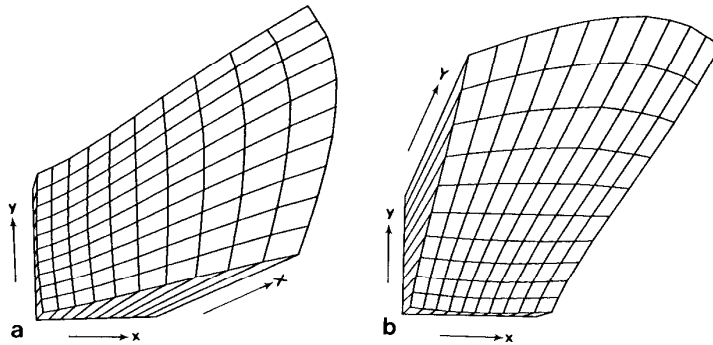


FIG. 4. (a) and (b) are, correspondingly, the  $X$ -component and the  $Y$ -component transformation functions needed to correct the deformed image of Fig. 1b to resemble the ideal image of Fig. 1a.

The left-hand side of the above relation contains only known quantities. Equation (9) thus shows a linear system of  $(m - 1) \times (n - 1)$  equations, which can be solved to determine the remaining control vertices  $V_{kl}$ ,  $k = 1, \dots, m - 1$ ,  $l = 1, \dots, n - 1$ .

As an example, let us use the grids of Fig. 1 to determine the distortion of the underlying lens. Figure 1a is the ideal grid and Fig. 1b is the deformed grid generated from a hypothetical lens. Assuming grid points in the ideal image are represented by  $(x_{ij}, y_{ij})$ ,  $i = 0, \dots, 10$ ,  $j = 0, \dots, 10$  and grid points in the deformed image are represented by  $(X_{ij}, Y_{ij})$ ,  $i = 0, \dots, 10$ ,  $j = 0, \dots, 10$ , we fit two patches to the sets of points  $(x_{ij}, y_{ij}, X_{ij})$ ,  $i = 0, \dots, 10$ ,  $j = 0, \dots, 10$ , and  $(x_{ij}, y_{ij}, Y_{ij})$ ,  $i = 0, \dots, 10$ ,  $j = 0, \dots, 10$  by the above process. The  $X$ -component and the  $Y$ -component transformation functions are shown in wireframe form in Fig. 4. A lens with no distortion would result in two flat patches. In Fig. 4, we see that the patches are curved with increasing curvature toward the patch borders. This shows that distortion increases toward the boundary of the lens. The two patches of Fig. 4 represent the distortion of the underlying lens in the  $X$  and  $Y$  directions and can be used to correct images obtained by the lens.

Next, we will apply the proposed decalibration process on images from an off-the-shelf lens.

#### 4. RESULTS

To exhibit the applicability of the proposed decalibration process on images from an off-the-shelf lens, we used a Fujinon-TV 1 : 1.2/1.25 lens obtained recently by the author for image acquisition purposes. We placed the grid of Fig. 5a in front of the camera and obtained an image from the grid. Figure 5b shows the image of the grid after being properly segmented [6].

Using the coordinates of corresponding grid points in images of Figs. 5a and b, we determined functions  $f$  and  $g$  using the process of Section 3.2. The obtained functions are shown in Figs. 6a and b. Although the patches seem to be flat, if we look closely at another representation of the patches, as shown in Figs. 6c and d, we notice moderate image deformation. Figures 6c and d are, correspondingly, the

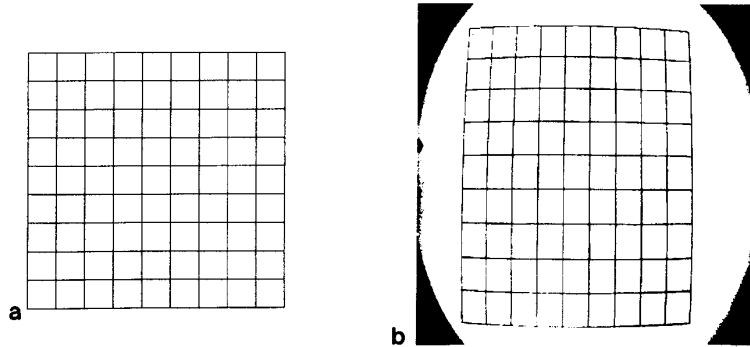


FIG. 5. (a) A  $10 \times 10$  grid. (b) Its image after segmentation.

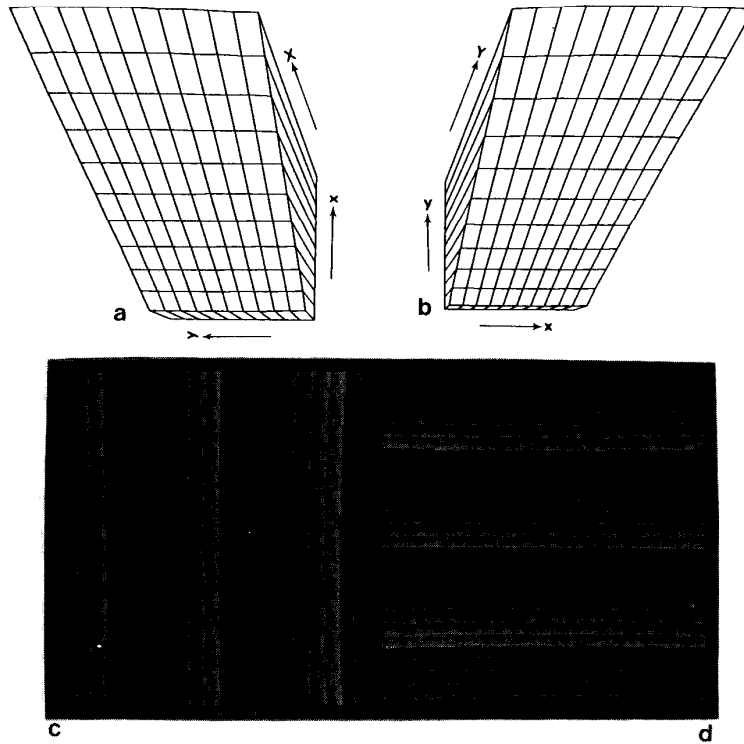


FIG. 6. (a) and (b) represent the X-component and the Y-component transformation functions, respectively. (c) and (d) are, correspondingly, the same as (a) and (b) with the third dimension shown by the intensities. Ribbons of constant intensity show columns and rows of the deformed image when mapped to the ideal image.

same as Figs. 6a and b but with the third dimension shown in intensities. The intensities are enhanced to bring out the deformation in the image. Ribbons of constant intensity show columns and rows of the deformed grid when mapped to the ideal grid. As can be seen, these ribbons are slightly curved, showing slight image deformation. Assuming that the center of the deformed grid maps to the center of the ideal grid, measurements showed that this particular lens displaces image points from their true positions by up to seven pixels.

Although image deformation obtained by this lens may not be visible by a quick glance at the images, when we measure the positions of the image points, we see that the image points are displaced. Consider determining the disparity of points in a pair of stereo images obtained by this lens for depth perception purposes. Suppose disparities in the stereo images vary from 0 to 35 pixels. Since this lens displaces a point by as much as seven pixels, the obtained disparities will be off by up to 20%.

To improve the accuracy of point positions in an image, it is necessary to correct the image before any measurement is made on it. The image correction procedure that was described above determines the control vertices of the characteristic polyhedrons of two Bezier patches that describe distortion in the  $x$  and  $y$  directions in the underlying lens. If images obtained by the lens are of size  $480 \times 512$ , then the Bezier patches may be described by two arrays of size  $480 \times 512$ , as shown in Figs. 6c and d. These arrays describe the mappings needed to transform a deformed image obtained by the lens to an ideal image that has no deformation.

Entry  $(x, y)$  of Figs. 6c and d, correspondingly, show the  $X$ -component and the  $Y$ -component of the point in the deformed image that corresponds to point  $(x, y)$  in the ideal image. Therefore, to determine the point in the deformed image that corresponds to point  $(x, y)$  in the ideal image, we look at entry  $(x, y)$  of Figs. 6c and d to read the  $X$ -component and the  $Y$ -component of the point. Knowing  $(X, Y)$ , we sample the intensity at  $(X, Y)$  of the deformed image and enter the sampled intensity into entry  $(x, y)$  of the ideal image. When we carry out this process for all values of  $(x, y)$  we sample the deformed image into the ideal image. Figure 7a shows resampling of the image of Fig. 5b with mapping functions of Figs. 6c and d. To show how well the deformed image is corrected, we have overlaid the deformed image with the original image in Fig. 7b. Figure 7c shows the correction of the deformed image using the polynomials of degree three as transformation functions and determining the parameters of the polynomials by the least-squares method. Figure 7d shows overlaying of images 7c and 5a. As can be seen, the Bezier patches have corrected the lens distortions better than the polynomials of degree three with least-squares. The least-square method cannot compensate for local distortions properly; it averages a local distortion over the entire image. This causes a local distortion to be spread all over the image. Bezier patches, on the other hand, being locally sensitive, can compensate for local image distortions without causing any side effects on the other areas of the image.

The processing time of the proposed decalibration process depends on the size of the grid. For a grid of size  $10 \times 10$  as shown in Fig. 5a, we needed a few seconds to determine the boundary control vertices by solving four systems of eight equations each, and about 2 min to determine the interior control vertices by solving a system of 64 equations on a MicroVax II computer. The resampling of an image depends on the size of the image. For an image of size  $480 \times 512$  and using the nearest neighbor method, about 2.5 min processing time was required to resample a

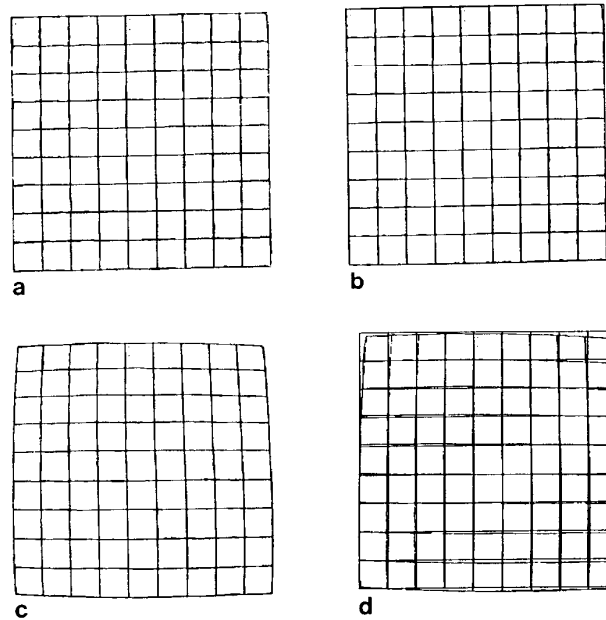


FIG. 7. Comparison of image transformation using the Bezier patches and the polynomials of degree three. (a) Image of Fig. 5b transformed using the Bezier patches. (b) Overlaying of images of Fig. 5b and (a). (c) Image of Fig. 5b transformed using the polynomials of degree three. (d) Overlaying of images of Fig. 5b and (c).

deformed image to its original form. The entire process, therefore, requires about 4.5 min on a MicroVax II computer.

The memory used in this technique is mostly for the storage of the two component mapping functions as shown in Figs. 6c and d. For images of size  $480 \times 512$ , we need two arrays of size  $480 \times 512$  each to store the  $X$ - and  $Y$ -component mapping functions. These two arrays, or the two sets of control vertices that describe the  $X$ - and  $Y$ -component patches, are the only ones that should be stored for any lens.

It should be mentioned that distortion in a lens varies with the temperature of the lens. Therefore, it is important that the above measurements be made at the temperature at which the lens will be operated. Fryer and Brown [7] have shown that lens distortion varies slightly with focusing too (by about a few percent). Therefore, it is important that the distortion measurements are made at about the focusing level at which the lens is to be operated. If the lens is to be operated at different focusing levels, then lens distortion may be measured at two different focusing levels and used in an interpolation formula to determine the lens distortion at an arbitrary focusing level [7].

##### 5. CONCLUSION

Almost every lens used for image acquisition contains some form and degree of distortion. A lens' distortion displaces points in an image, causing image deforma-

tion. In this note, a decalibration process was described that can correct image deformation caused by lens distortion. This process involves estimation of two Bezier patches that describe lens distortion in the  $X$ -direction and in the  $Y$ -direction. The patches are then used as the  $X$ - and the  $Y$ -component transformation functions that map the deformed image into the ideal image.

The control vertices of the Bezier patches are determined by solving a system of equations using the grid points through which the patches should pass. The Bezier patches obtained in this form map the grid points in the deformed image exactly to the grid points in the ideal image and interpolate the rest of the points. Examples exhibiting the proposed decalibration process on images from a hypothetical lens and a real lens were presented.

#### REFERENCES

1. *Manual of Photogrammetry* (M. Thompson, Ed.), 3rd ed., pp. 84–86, Amer. Soc. Photogrammetry, Washington, DC, 1976.
2. W. B. Green, P. L. Jespen, J. E. Kreznar, R. M. Ruiz, A. A. Schwartz, and J. B. Seidman, Removal of instrument signature from Mariner 9 television images of Mars, *Appl. Opt.* **14**, No. 1, 1975, 105–114.
3. N. Yokobori, P. Yeh, and A. Rosenfeld, Selective geometric correction of images, in *Proceedings, Computer Vision and Pattern Recognition, 1986*, pp. 530–533.
4. M. E. Mortenson, *Geometric Modeling*, pp. 113–125, 215–219, Wiley, New York, 1985.
5. B. A. Barsky, A description and evaluation of various 3D models, *IEEE Comput. Graphics & Appl.* **4**, No. 1, 1984, 38–52.
6. C. K. Chow and T. Kaneko, Automatic boundary detection of the left ventricle from cineangiograms, *Comput. Biomed. Res.* **5**, 1972, 388–410.
7. J. G. Fryer and D. C. Brown, Lens distortion for close-range photogrammetry, *Photogramm. Eng. Remote Sens.* **52**, No. 1, 1986, 51–58.