Refinement in 3D Reconstruction using Cheirality Constraints

Sunando Sengupta and Sukhendu Das
Visualization and Perception Laboratory
Department of CSE
IIT - Madras, Chennai - 600036, India
Email: sunando@cse.iitm.ernet.in and sdas@iitm.ac.in

***Abstract*—The problem of 3D reconstruction has become more and more important with the increase in demands from virtual reality, visualization, security and biomedical industries. Uncalibrated reconstruction, from a set of images taken from any ordinary CCD camera, is complicated and difficult due to the presence of non-linear constraints on the scene. Any linear solution suffers from numerical instability or it requires additional scene information and camera information to be provided. Here we present a reconstruction method which works without any additional camera parameters after we obtain the plane at infinity by using cheiral constraints or scene information. We also study the ambiguities arising due to various types of reconstructions. We show the results using synthetic data and real world data.**

Keywords: 3D Reconstruction, Epipolar Geometry, Reconstruction Ambiguity, Cheirality, Auto-calibration.

I. Introduction

The problem of reconstruction of a 3D shape from a set of images has been in attention and a subject of study for the past two decades. With the development of multimedia and virtual reality industry the need for getting a precise 3D model for any real world object becomes more and more important. Different applications in computer graphics require that a 3D model of the object be created from an array of images taken with various exterior orientations. But the problem of reconstruction suffers from the problem of projective ambiguity, which best stated as, the reconstructed model differs from the original 3D model by a projective transformation [1], [2], [3], [4], [5]. It has been shown that for metric reconstruction some additional scene information is required other than the image correspondences. This information may be presented in the form of parallelism in the scene, or in form of any information about the camera internal parameters. In addition to the scene information, presence of noise in the images and errors in correspondences make the procedure numerically unstable, and in most of the cases a unique solution does not exist, and hence a least squares approximation is often employed.

In this paper, we present a robust method for reconstruction once the plane at infinity (π_∞) is first established by cheiral inequality or through scene information. The method of reconstruction uses the (π_∞) to obtain the infinite homography, induced by π_∞, for obtaining metric reconstruction. But such methods are prone to numerical sensitivity as often the 'Absolute conic' image is not positive definite and thus the method fails. We try to overcome such numerical sensitivity by using square-root covariance which guarantees positive definiteness, making the algorithm more robust. We also study reconstruction ambiguity arising due to Triangulation method [6]. A projective reconstruction ensures that a recovered structure will be a projectivity away from the actual scene. We try to model ambiguity on simulated data and average it over a large number of iterations. The paper is organized as follows: Section 2 discusses about the general camera geometry and the ambiguity in reconstruction. Here we give the general stratified reconstruction process. In Section 3, we speak about Auto-calibration and Cheirality. In section 4 we introduce our algorithm and prove why it will work. We present the experimental results in section 5 and conclude in section 6.

II. Projective Reconstruction and Camera Geometry

A point in a 3D scene is represented by its homogenous coordinates, i.e. a 4D vector $[X, Y, Z, 1]^T$ and the 2D projection as $[x, y, 1]^T$. We consider a pinhole camera model where the image plane is defined by $z = f$. So under the pinhole camera model a 3D point with the coordinates $\mathbf{X} = [X, Y, Z, 1]^T$ is mapped into $\mathbf{x}^p = [fX/Z, fY/Z, f]^T$. So the camera can be represented by a matrix P that transforms a set of 3D space points to 2D image points, as

$$\mathbf{x}_i^p = P\mathbf{X}_i, \ i = 1, 2, ..., N \tag{1}$$

The Camera matrix contains the information of the camera; both the internal parameters, namely the zoom, focus, skew and external parameters like rotation and the translation with respect to the world coordinate frame. The camera matrix is given as

$$P = K[R|\mathbf{t}] \tag{2}$$

where, K contains the camera internal parameters, R and $\mathbf{t}$ are rotation and translation parameters.. The internal parameters are arranged in an upper-triangular matrix,

$$K = \begin{bmatrix} \alpha_x & s & x_0 \\ & \alpha_y & y_0 \\ & & 1 \end{bmatrix} \tag{3}$$

where α_x and α_x represent the focal length in terms of pixel dimension in the x and y direction respectively, s is the

camera skew and $(x_0, y_0)^T$ are the coordinates of the principal point [1]. For most of the cameras the skew parameter, s, is zero (square pixels). The camera calibration is important as it determines the Euclidean properties, such as the angle between two rays, which is important for reconstruction. An image point back-projects to a ray defined by the point and the camera center. The calibration gives the direction of the projected ray [2]. The Image of the Absolute Conic (IAC) [7] is also related to the camera calibration parameters as

$$\omega = K^{-T}K^{-1} \tag{4}$$

Thus,

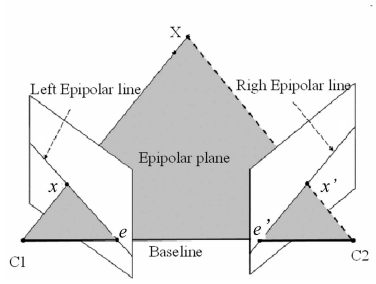
$$\omega = \frac{1}{\alpha_x^2\alpha_y^2}\begin{bmatrix} \mathbf{c}_1 & \mathbf{c}_2 & \mathbf{c}_3 \end{bmatrix} \tag{5}$$

where,

$$\begin{aligned}
\mathbf{c}_1^T &= [\alpha_y^2, \ -s\alpha_y, \ -x_0\alpha_y^2 + y_0 s\alpha_y] \\
\mathbf{c}_2^T &= [-s\alpha_y, \ \alpha_x^2 + s^2, \ \alpha_y s x_0 - \alpha_x^2 y_0 - s^2 y_0] \\
\mathbf{c}_3^T &= [-x_0\alpha_y^2 + y_0 s\alpha_y, \ \alpha_y s x_0 - \alpha_x^2 y_0 - s^2 y_0, \\
&\quad \alpha_x^2\alpha_y^2 + \alpha_x^2 y_0^2 + (\alpha_y x_0 - s y_0)^2]
\end{aligned}$$

So, once the IAC is known to a scale, the camera parameters can be determined using Cholesky Factorization [7] and hence the metric reconstruction can be performed. So restoring the IAC or its dual image (DIAC, given by $\omega^* = \omega^{-1} = KK^T$) to its canonical position will help in realizing the metric reconstruction.

Now given two views of the same object, where each view differs by some amount of rotation and/or translation, the points in one image can be correlated with another set of points in the other image. This problem is called the *correspondence problem* [8] in stereo. This concept can be generalized to the three-view correspondence problem [9] or, in general, the N-view correspondence problem [10]. The geometry involved with the intersection of the image planes with the pencil of the planes having the same baseline as axis (the baseline is the line joining the camera centers) is termed as *epipolar geometry* (see Fig 1). As the views of the object are given relative to each other, the first projective camera matrix can be written as $P_0 = K_0[I|\mathbf{0}]$ where, K_0 contains the camera internal parameters. The camera axes and origin are aligned with the world origin and axes respectively. A rotation of the camera is equivalent to matrix multiplication of the camera matrix P_0 by an appropriate rotation matrix R and translation vector t. So the camera matrix for the second camera becomes $P_1 = K_1[R|\mathbf{t}]$. In a reconstruction problem, the input is a set of images from which the scene needs to be reconstructed. Given the images the first task is to compute the correspondences, which can be done using a standard feature tracking mechanism like Harris corner or KLT feature tracker [11], [12]. Now given any image pair, the correspondences in both the images are related by the Fundamental matrix, [13], [1], [14], [3] as $\mathbf{x}_i^{'}F\mathbf{x}_i = 0$ where $\mathbf{x}_i$ and $\mathbf{x}_i^{'}$ are the pair of image correspondences. From epipolar geometry [1],



(a)

Fig. 1. Epipolar geometry for the case of 3D point reconstruction from two images.

[15] we obtain the projective camera matrix (corresponding to the image pairs) from the Fundamental matrix. Now, the correspondences $\mathbf{x}_i \leftrightarrow P\mathbf{X}_i$ and $P\mathbf{X}_i \leftrightarrow P^{'}\mathbf{X}_i^{'}$ are equivalent. Thus the equations $(\mathbf{x}_i) \times (P\mathbf{X}_i) = 0$ and $(\mathbf{x}_i) \times (P^{'}\mathbf{X}_i) = 0$ ($\times$ refers to cross product between two vectors) can be rearranged in a form to obtain 3D $\mathbf{X}_i$, as:

$$A_i\mathbf{X}_i = 0 \tag{6}$$

The right-null vector of A_i will give the desired 3D point [1], [6], [2]. But due to the presence of noise and errors in correspondences the null space cannot be guaranteed to exist. In such cases, a linear least squares solution can be obtained via the SVD [16] of A_i. For the case of uncalibrated cameras, the 3D points are computed only up to an arbitrary projective transformation in 3D space. For any invertible 4×4 matrix H representing the projective transformation of 3D space, replacing each point X_i by $H\mathbf{X}_i$ and the camera matrices P_i by P_iH^{-1}, we get the same image points as

$$\mathbf{x}_i = P\mathbf{X}_i = P_iH^{-1}H\mathbf{X}_i \tag{7}$$

Thus the reconstructed camera matrix and actual camera matrix differs by a projective transformation [1]. Hence, using only the image correspondences, the 3D reconstructed points and the corresponding camera matrices can be obtained only up to a projective scale [17], [14]. So it is essential to get some additional information about the scene to compute the 3D structure from a given pair of images. To resolve this kind of ambiguity a step-wise procedure called *Stratified Reconstruction.* [18], [1], [2] is used. In Stratified reconstruction additional scene information is fused in to refine the projective model progressively to an Affine transformation first and finally to a Metric transformation. The most common method used to resolve this kind of ambiguity, is to identify vanishing points in the scene and then identify the plane at infinity π_∞. As in case of Affine transformation, the plane at infinity π_∞ is preserved to its canonical position, but not in projective transformation. Identifying the plane π_∞ can bring down the uncertainty from projectivity to affinity. A projective reconstruction transfers π_∞ to some generic position π, resulting in the arrival of the vanishing points. So, given the plane at infinity in its projective position π the projective transformation H will act on the plane contravariantly such

that $H^{-T}\pi = (0,0,0,1)^T$. Thus the resulting transformation restoring to affinity will be

$$H = \begin{bmatrix} I & | & \mathbf{0} \\ \pi^{\mathbf{T}} & & \end{bmatrix} \tag{8}$$

The refinement from affinity to metric reconstruction can be performed by identifying the IAC [7], [18]. The Absolute Conic is equivalent to finding the camera calibration parameters. The Absolute Conic, a planar conic lying on the plane at infinity, captures the metric properties, which transferred to its canonical position ($x_1^2+x_2^2+x_3^2 = 0$ on π_∞) will restore back the metric properties [7]. The image of the absolute conic is a property of the image itself and is related to camera parameters as given in Eqn. 5. The back-projection of this conic is a cone, which intersects the plane at infinity at a conic, defining the absolute conic. The absolute conic is related to the camera internal parameters as given by Eqn. 4.

In Fig. 2 a simulated data of Rubik's Cube is reconstructed from two views differing by a rotation of 45 degrees and a translation **t** = [-1, 0, 1] using stratified reconstruction. The Affine reconstruction is obtained by identifying the vanishing points in the orthogonal direction. The metric reconstruction has been done with the knowledge of camera parameters, obtained beforehand by any standard calibration method using the calibration objects toolkit [19].

III. Auto-calibration and Cheirality

Auto-calibration is the process of determining the camera parameters from a set of image sequences of the scene. The problem with the stratified reconstruction is the dependency of external scene information (parallel lines in scene for transforming to affinity) or camera parameters (for transformation to metric). Often these information are not available and hence it becomes essential to learn these parameters from the sequence of images of the scene. The prerequisite for auto-calibration is to have projective reconstruction from the available image correspondences. For any projective camera, $P^i = [A^i|a^i]$, the auto-calibration constraint can be given as

$$\omega_i^* = K^i K^{iT} = (A^i - a^i\mathbf{p}^T)K^1K^{1T}(A^i - a^i\mathbf{p}^T)^T \text{ for i=2,..,m} \tag{9}$$

where p^T represents the plane at infinity, i represents the i^{th} camera. For $m \geq 3$, the parameters can be solved and hence the reconstruction can be performed. But as the constraints are highly non-linear arriving at a global solution becomes a very difficult task, and most of the optimization technique suffer from local minima or the initialization problem. Linear solution [1] for estimation of the camera calibration parameters exists, which needs the knowledge of the plane at infinity. This again requires either user intervention or the process of a non-linear optimization. Similar constrains are expressed through Kruppa equations on the IAC [2], [20].

A. Cheirality

Cheirality constraints arise from the fact that any point that lies in an image must lie in front of the camera producing that image [1], [2]. Using this idea, the scene can be determined

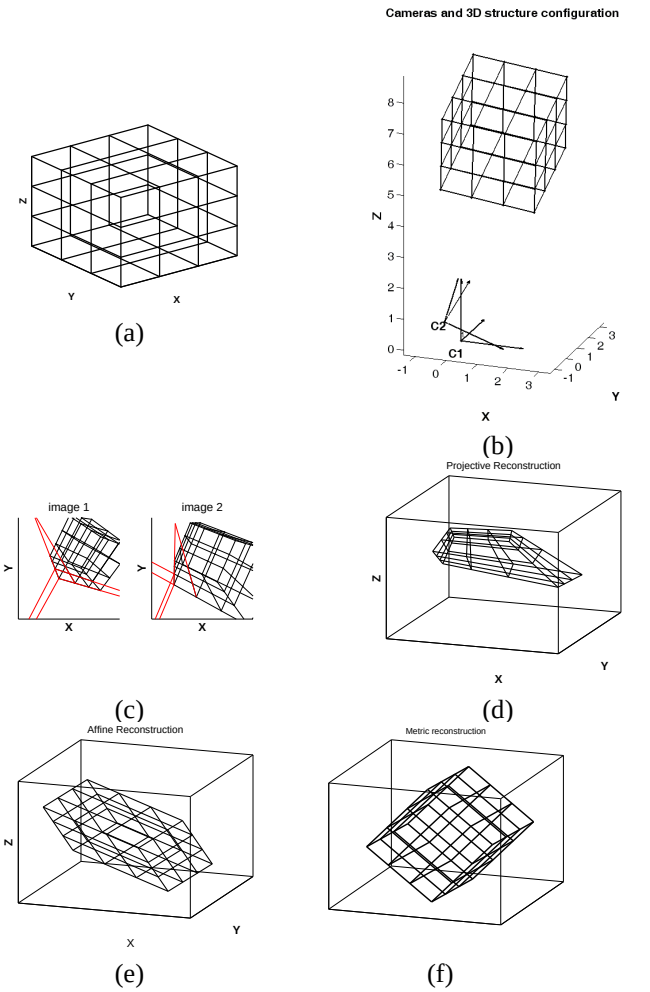


Fig. 2. The synthetic Rubik's Cube and its reconstruction. (a) The Rubik's Cube (b) The actual 3D structure of the Rubik's Cube and the camera positions. The camera positions are given by C1 and C2. (c) The images of the Cube as viewed from the two cameras and their respective vanishing points (d) Projective reconstruction of the cube from the two images (e) Affine reconstruction (f) Metric reconstruction for calibrated case.

from two views up to a more restricted class of mappings known as quasi-affine transformations, which are precisely those projectivities that preserve the convex hull of an object of interest. An invariant of quasi-affine transformation known as the Cheiral sequence of a set of points is defined, and it is shown how the cheiral sequence may be computed using two uncalibrated views. The depth of a point $\mathbf{X} = [X, Y, Z, T]$, with respect to the camera $P = [M|m]$, is given as [21], [1]

$$depth(\mathbf{X}; P) = \frac{sign(detM)w}{\mathrm{T}||\mathbf{m}^3||}$$

where $\mathbf{m}^3$ is the last row of M and $P\mathbf{X} = w\mathbf{x}$. So, the sign is given as

$$depth(\mathbf{X}; P) \doteq sign(detM)w\mathrm{T} \tag{10}$$

where $\doteq$ denotes equality of sign. The quantity $sign(depth(\mathbf{X}; P))$ is known as the cheirality of point $\mathbf{X}$ with respect to camera P. Thus for any transformation

matrix H, transforming X and P as PH^{-1} and $H\mathbf{X}$, the cheirality transforms are

$$depth(H\mathbf{X}; PH^{-1}) \doteq w(h_4^T\mathbf{X})(h_4^T\mathbf{C}_p)det(H^{-1}) \quad (11)$$

where h_4^T is the fourth row of H and C_p is the camera center. Thus if π_∞ is mapped by a projective transformation H and $\delta = sign(det(H))$, then

$$depth(H\mathbf{X}; PH^{-1}) \doteq w(\pi_\infty\mathbf{X})(\pi_\infty\mathbf{C}_p)\delta \quad (12)$$

For the point to be in the front of the camera the above mentioned quantity must be greater than zero. Thus we arrive at the cheiral inequality [21], [22], [23]

$$\begin{aligned} \mathbf{X}_i^T\pi_\infty > 0 \text{ for all points i=1,..,n} \\ \delta\mathbf{C}^{jT}\pi_\infty > 0 \text{ for all cameras j=1,...,p} \end{aligned} \quad (13)$$

Where n is the number of points in the scene and p is the number of cameras. Thus π_∞ can be computed from the above inequality for each $\delta = \pm 1$ and hence a quasi-affine transformation can be attained without any extra input scene information [15]. Once the quasi-plane is known the camera parameters can be computed from the projective camera matrices in a linear least-square based solution using the infinite homography [1], [7]. The infinite homography induced by the plane at infinity is given by

$$H_\infty^i = A^i - \mathbf{a}^i\mathbf{p}^T \quad (14)$$

H_∞^i represents the homography from camera $[I|\mathbf{0}]$ and $[A^i|a^i]$. So the dual conic in Eqn. 9 is now

$$\omega^* = H_\infty^i\omega^*{H_\infty^i}^T \quad (15)$$

This can be rearranged as a 6x6 matrix A, composed of elements of H_∞^i, forming $Ac = 0$, where c is composed from the elements the conic ω^*. A linear least squares solution will yield a solution for c. But the matrix has rank 4, so $m \geq 2$ will give a solution to ω^* and the metric reconstruction [7], [15], [21]. But the problem with the least-squares based method is that the absolute conic is numerically instable and may not be positive definite which is essential for Cholesky decomposition [24]. This means that the camera parameters cannot be decomposed always from the IAC and hence the metric reconstruction cannot be done. We try to overcome this problem in our proposed algorithm.

IV. Proposed Algorithm

The problem with using the plane at infinity, π, as estimated from quasi-affine transformation (Eqn. 13) is that when trying to obtain the DIAC, ω^*, may not be positive definite. Such class of matrices is known as the *Indefinite System* [24].

The problem with any Symmetric Indefinite Systems is that the form x^TAx can take any positive or negative value. From the SVD based linear solution of $\omega^*(Eqn.15)$ we obtain an indefinite system, which may have negative pivot. A symmetric positive definite matrix has a positive "weighty" diagonal [24] (large diagonal entries) making all the eigenvalues positive. For an indefinite system it ceases to be such a case, making it unviable for cholesky decomposition. To overcome this problem we follow the following approach.

Let $A = U\Sigma V^T$ be the Singular value decomposition of $A \in \Re^{m\times n}$. The cholesky decomposition of A is given by $A = KK^T$. Now for indefinite matrix systems, we define

$$\hat{A} = U\Sigma U^T \quad (16)$$

We use $\hat{A}$ and undergo its cholesky analysis which is guaranteed to exist as the eigenvalues are replaced by the singular values A. The singular values of A are determined through the SVD decomposition of A. This is guaranteed to be positive definite as

$$\hat{A} = U\Sigma U^T = (U\sqrt{\Sigma})(\sqrt{\Sigma}U)^T = R^TR \quad (17)$$

We take positive eigenvalues instead of positive pivots of the matrix and prove it to be positive-definite [24], [16]. Let $C = AA^T$, the eigenvalues of C are given by columns of U, obtained through SVD of A. Then

$$\begin{aligned} AA^T &= (U\Sigma V^T)(U\Sigma V^T)^T \\ &= U\Sigma V^TV\Sigma^TU^T \\ &= U\Sigma^2U^T \\ &= \hat{A}^2 \end{aligned} \quad (18)$$

This comes from the fact that V is orthogonal and Σ is a diagonal matrix of singular values. So $\sqrt{\Sigma}$ exists and columns of U are eigenvectors of AA^T. Now, from spectral theorem every real symmetric matrix can be diagonalized by an orthogonal matrix, i.e. $A = Q\Lambda Q^{-1}$, where columns of Q contain a complete set of orthonormal eigenvectors and Λ is a diagonal matrix formed from the eigenvalues of A [16]. Power of any matrix can be obtained from the spectral theorem by powering the eigenvalues as $A^k = Q\Lambda^kQ^{-1}$. Thus we see that $\hat{A}$ is *Squared Root Covariance* of A. The DIAC, obtained via the infinite homography (Eqn. 15), is rectified by Eqn. 16 to obtain $\hat{\omega}^*$ before obtaining the final metric rectifying transformation. The rectified DIAC, $\hat{\omega}^*$, guarantees robustness to noise and provides numerical stability.

So in our algorithm (see Fig. 3), the input is set of input views of the 3D scene. We compute the point correspondences using Harris corner detector [11] and compute the projective reconstruction. From projective reconstruction, we arrive at a Quasi-affine reconstruction by solving the inequalities as stated in Eqn. 13. The quasi-affine transformation gives us an approximate plane at infinity that can be used for determination of the metric reconstruction. We obtain the image of absolute conic ω^*, by linear least-square based estimation using SVD [24], [1] from Eqn. 14. The ω^* is then corrected by our proposed method described earlier in this section and finally metric rectification is done. In the following section results are described along with error analysis.

V. Experiments and Results

As we have seen the ambiguity in the projective reconstruction process, we explore a of bit detail into that before analyzing our proposed algorithm. Using simulated data we compute

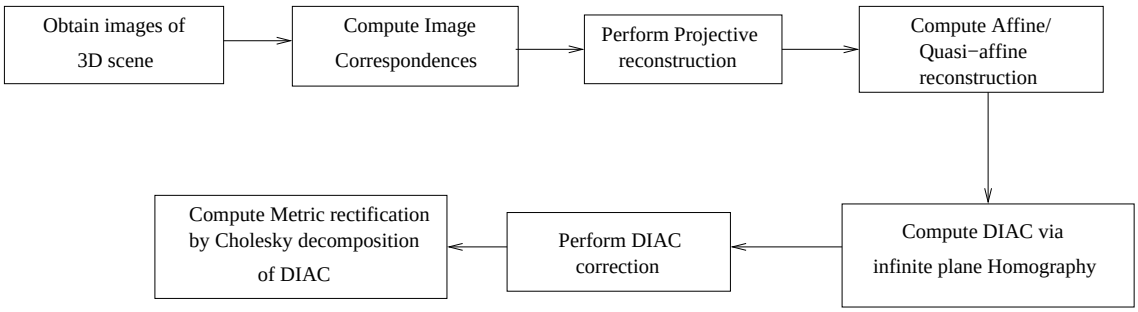


Fig. 3. Flowchart for our algorithm

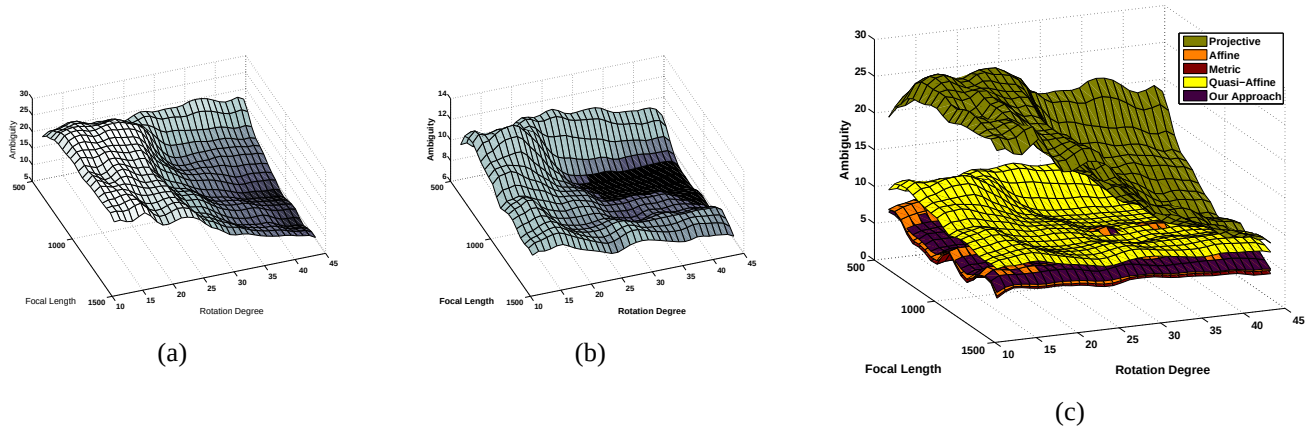


Fig. 4. Ambiguity Surface (a) Projective reconstruction (b) Quasi-affine reconstruction (c) Ambiguity surfaces for different reconstruction after data normalization.

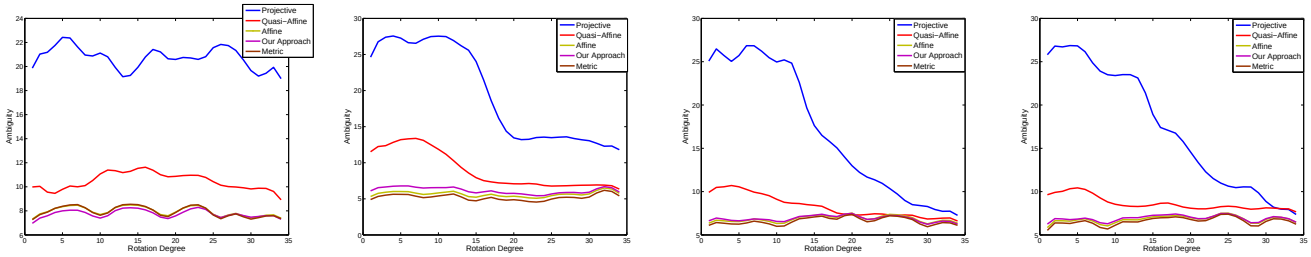


Fig. 5. Ambiguity plot at various focal length for Projective, Quasi-affine, Affine, our approach and Metric where f= (a) 500 (b) 750 (c) 1000 (d) 1250.

average ambiguity per point as the Euclidean norm between the actual and the reconstructed point. The reconstructed point clouds are normalized such that the centroid is at the origin and average distance from origin to each point is $\sqrt{3}$, that is, the average point is at location $(1, 1, 1)^T$. This is due to the fact that the reconstructed point clouds can be determined up to a similarity transform. We obtain *Ambiguity Surfaces* for projective reconstruction, affine, metric, quasi-affine, and our proposed algorithm (Fig. 4). We see in Figs. 4(a) and (b), the ambiguity is large for small focal length and rotation between the appearances. It settles down with increase in rotation and focal length. The Fig. 4(c) shows the regions (in rotation-focal length domain) where the corresponding ambiguity is high using a colormap. As expected, the projective reconstruction is most ambiguous, followed by quasi-affine surface. Affine surface and surface corresponding to our approach are very close and overlapping. Ambiguity surface corresponding to metric reconstruction is least as expected (see Fig. 4(c)). The error surface is obtained by taking the average ambiguity over a large number of iterations varying the viewing direction in each direction. As we go into more constrained reconstruction we see that the average ambiguity also reduces. The upper surface is the ambiguity for the projective reconstruction. The ambiguity is plotted for different focal length for all the reconstructions as shown in Fig 5.

Now we describe the experimentation for the proposed algorithm as described in section IV. We use synthetic 3D data and also natural images. For the synthetic case, we simulate 3D Rubik's Cube and a house [2] and view it from different positions. The various views are taken as the images,

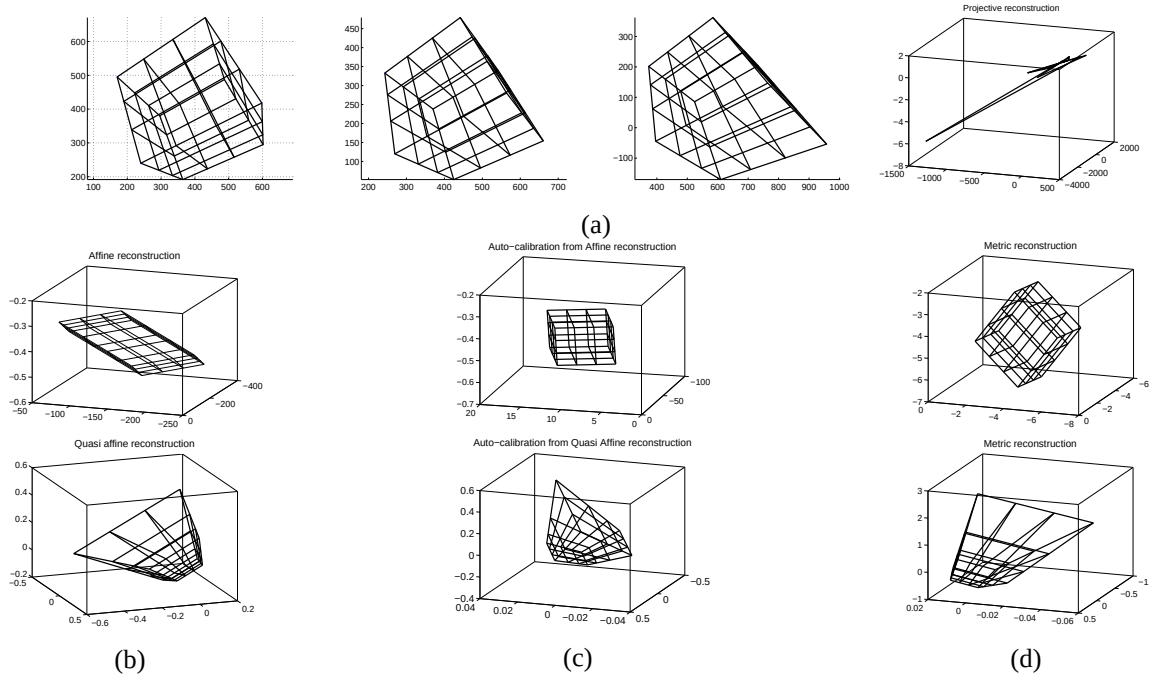


Fig. 6. Simulated Data (Rubik) (a) These are the images and the corresponding projective reconstruction achieved from the images. (b) Affine and Quasi-affine reconstruction respectively (c) Metric reconstruction through auto-calibration for both Affine and Quasi-affine (d)Metric reconstruction via known internal parameters for both Affine and Quasi-affine

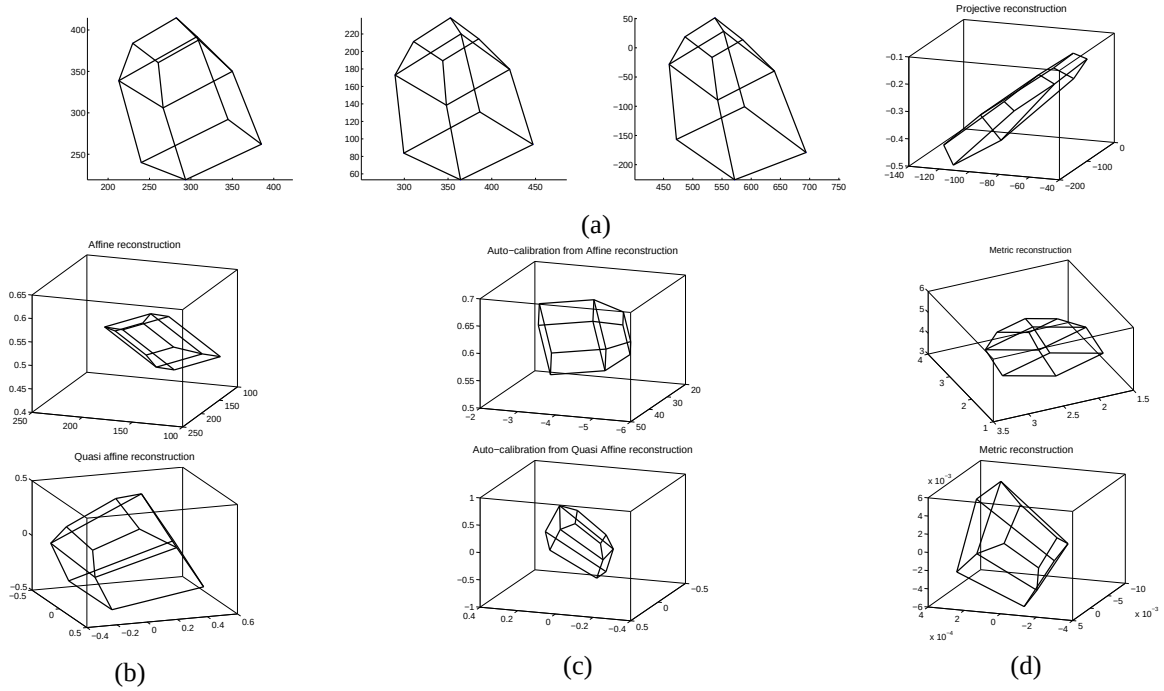


Fig. 7. Simulated Data (House) (a) These are the images and the corresponding projective reconstruction achieved from the images. (b) Affine and Quasi-affine reconstruction respectively (c) Metric reconstruction through auto-calibration for both Affine and Quasi-affine (d)Metric reconstruction via known internal parameters for both Affine and Quasi-affine

which are used for further processing. The output of projective reconstruction (Eqn. 6) is used to compute the structure for Quasi-Affine transformation and affine transformation. We show both the cases when plane at infinity π_∞ is determined using knowledge of parallel lines in the scene as well as through cheiral inequalities (Eqn. 13). Then we proceed for metric reconstruction for both the cases (see Figs. 6 and 7):(i) through auto-calibration via the plane at infinity determined through both scene parallel lines and cheiral inequality and (ii) metric reconstruction with known internal camera parameters via the plane at infinity (again determined through both cases). The projective reconstruction from the images is shown in Figs 6(a) and 7(a). Then we obtain a Affine/Quasi-affine transformation which is shown in Figs 6(b), 7(b). Our modified auto-calibrated approach is shown in Figs 6(c) and 7(c), and finally we arrive at metric reconstruction (Figs 6(d), 7(d)). The metric reconstruction is done with the knowledge of camera internal parameters from affine/quasi-affine reconstruction. We see that without the knowledge of the camera internal parameters reconstruction is close to metric rectification (which assumes knowledge of camera internal parameters). For real data we take images from IGOIL database [25], where images are taken on a turn-table sequence with an interval of 5 degrees. We have also taken chequered box images at intervals of five degrees, which we have used for training to obtain the approximate vanishing points. We compute the approximate vanishing points off-line and use it during the reconstruction for other objects. The projective reconstruction is obtained by the Eight-point algorithm [26]. The affine transformation is obtained through Eqn. 8, followed by modifying the DIAC as in Eqn. 16. Figure 8 shows the reconstruction of a book taken from the IGOIL database. The images are taken with a rotation angle of 20° between the views. We see in the final reconstruction that all the points do lie in a plane (evident from the fact that all the points approximately can be viewed to lie on a line at some viewing angle). Similarly in the second example (Fig 9) we try to reconstruct points of a basket. We see the final point cloud form an approximate curved surface. In both the cases we can see that it is an improvement over affine transformation (with respect to Figs 8(e) and 9(e)).

VI. Conclusion

We present a method of metric reconstruction through the Auto-calibration method via the plane at infinity π_∞. We show results using both the simulated case as well as for a real image. The plane at infinity π_∞ is obtained using the information of parallel lines from the 3D scene or approximated by Quasi-affine transformation. The main contribution lies in approximating the DIAC by square-root covariance, guaranteeing its positive definiteness. Thus the DIAC can be decomposed to a cholesky factorization to get a satisfactory metric reconstruction. This method removes the necessity for knowing the internal camera parameters and is robust. We see that for simulated data it is close to the actual metric reconstruction obtained through the knowledge of internal parameters. For real data we propose an approach of off-line learning of scene information where a chequered box is rotated at equal intervals of five degrees. This data is used to learn (approximately) the parallel lines in the scene given the rotation information. This information is used to get the approximate plane at infinity. The information of plane at infinity is subsequently used for our proposed auto-calibration method. Our algorithm is robust to image noises and numerical instability and it applies to a wide variety of common real world situations, such as a fixed camera with rotating object or vice-versa.

References

[1] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2002.

[2] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision*. Springer, 2004.

[3] O. Faugeras, *Three Dimensional Computer Vision*. MIT Press, 1993.

[4] P. F. Sturm and B. Triggs, "A factorization based algorithm for multi-image projective structure and motion," in *ECCV '96: Proceedings of the 4th European Conference on Computer Vision-Volume II*. London, UK: Springer-Verlag, 1996, pp. 709–720.

[5] O. Faugeras, "What can be seen in three dimensions with an uncalibrated stereo rig?" in *ECCV*, 1992, pp. 563–578.

[6] R. I. Hartley and P. Strum, "Triangulation," in *ARPA Image Understanding Workshop*, 1994, pp. 957–966.

[7] A. Fusiello, "Uncalibrated euclidean reconstruction: a review," *Image Vision Comput*, vol. 18, no. 6-7, pp. 555–563, 2000.

[8] D. Marr and T. Poggio, "Cooperative computation of stereo disparity," *Science*, vol. 194, pp. 283–287, 1976.

[9] R. Hartley, "Lines and points in three views and the trifocal tensor," *IJCV*, vol. 22, pp. 125–140, 1997.

[10] P. Torr, A. W. Fitzgibbon, and A. Zisserman, "The problem of degeneracy in structure and motion recovery from uncalibrated image sequences," *International Journal of Computer Vision*, vol. 32, pp. 27–44, August 1999.

[11] C. J. Harris and M. Stephens, "A combined corner and edge detector," in *4th Alvey Vision Conferences*, 1988, pp. 147–151.

[12] J. Shi and C. Tomasi, "Good features to track," in *IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, June 1994.

[13] Z. Zhang, R. Deriche, O. Faugeras, and Q. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry." in *Rapport de recherche 2273, INRIA*, May 1994.

[14] T. S. Huang and . D. Faugeras, "Some properties of the E matrix in two-view motion estimation," *IEEE Trans. on PAMI*, vol. 11, pp. 1310–1312, 1989.

[15] R. I. Hartley, "Euclidean reconstruction from uncalibrated views," in *Proceedings of the Second Joint European - US Workshop on Applications of Invariance in Computer Vision*. London, UK: Springer-Verlag, 1994, pp. 237–256.

[16] G. Strang, *Linear Algebra and its Applications*. Thomson Brooks, 2005.

[17] R. Hartley, R. Gupta, and T. Chang, "Stereo from uncalibrated cameras," in *CVPR*, June 1992, pp. 761–764.

[18] O. Faugeras, "Stratification of 3-dimensional vision: Projective, affine, and metric representations," *JOSA-A*, vol. 12, no. 3, pp. 465–484, March 1995.

[19] J. Bouguet, "Matlab camera calibration toolbox," in *Technical Report Caltech*, 2000.

[20] R. I. Hartley, "Kruppa's equations derived from the fundamental matrix," *IEEE Trans. on PAMI*, vol. 19, no. 2, pp. 133–135, Feb 1997.

[21] R. Hartley, "Cheirality invariants," in *DARPA93*, 1993, pp. 745–753.

[22] T. Werner and T. Pajdla, "Cheirality in epipolar geometry," in *Research Report, Czech Technical University*, 2000, pp. 548–553.

[23] R. I. Hartley, "Chirality," *Int. J. Comput. Vision*, vol. 26, no. 1, pp. 41–61, 1998.

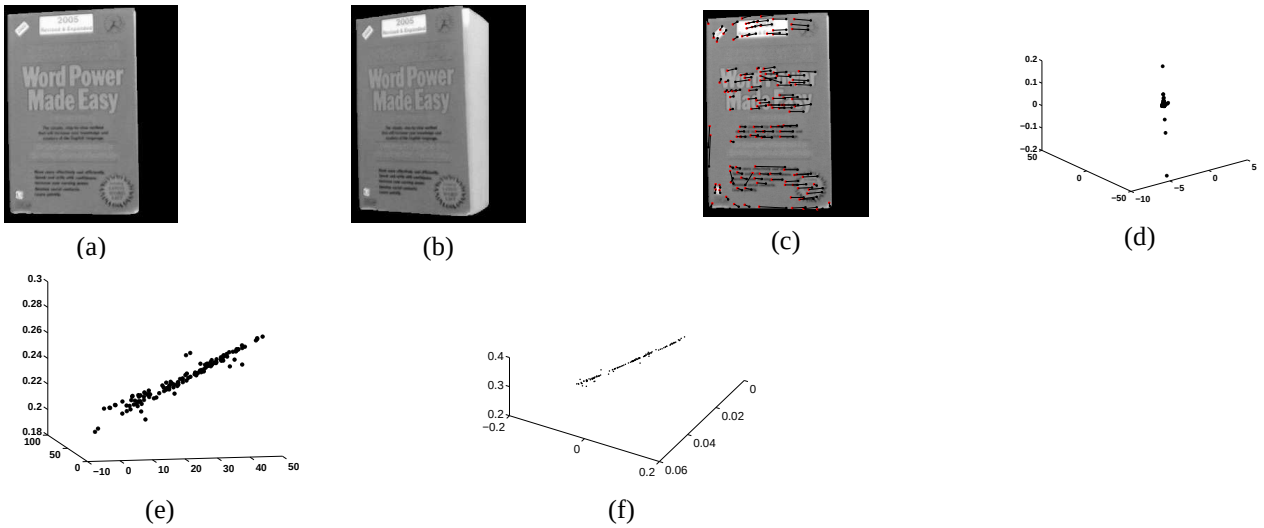


Fig. 8. The results of reconstruction of book image from IGOIL database (a) and (b) input images separated by 20° (c) correspondences in the images (d) Projective reconstruction (e) Affine reconstruction (f) Metric reconstruction

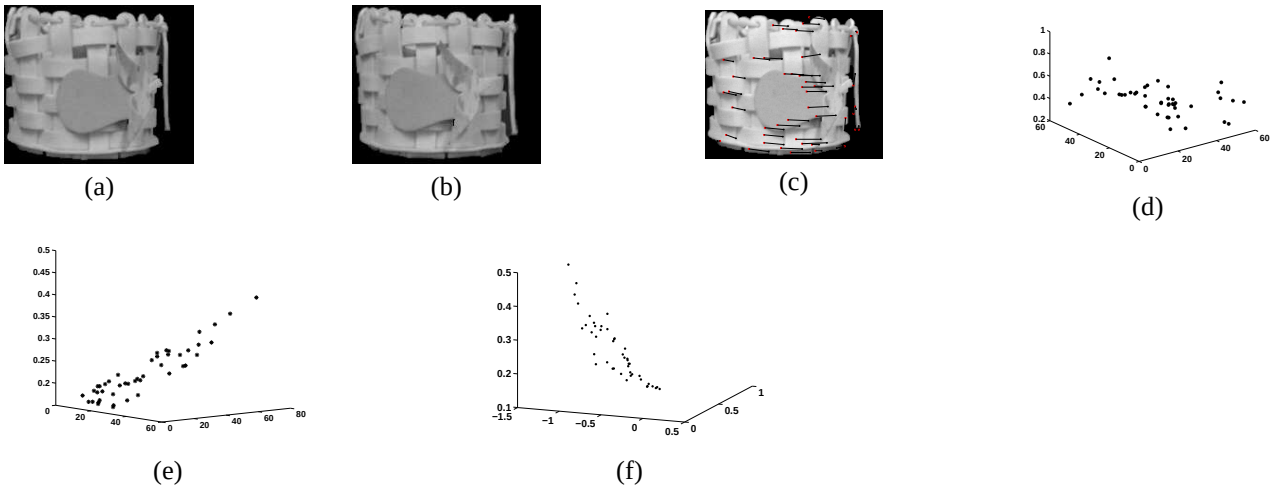


Fig. 9. Reconstruction of basket image from IGOIL databases (a) and (b) input images separated by 20° (c) correspondences in the images (d) Projective reconstruction (e) Affine reconstruction (f) Metric reconstruction

[24] G. H. Golub and C. F. V. Loan, *Matrix Computation.* John Hopkins University Press, 1996.

[25] M. Kalra and S. Das, "IITM Generic Object Image Library," 2006, http://www.cs.iitm.ernet.in/~sdas/vplab/downloads.html.

[26] R. Hartley, "In defense of the eight-point algorithm," *IEEE Trans. on PAMI*, vol. 19, pp. 580–593, 1997.