Globally Optimal Affine and Metric Upgrades in Stratified Autocalibration

Manmohan Chandraker[†] mkchandraker@cs.ucsd.edu

Sameer Agarwal[‡] sagarwal@cs.washington.edu

David Kriegman[†] kriegman@cs.ucsd.edu

Serge Belongie[†] sjb@cs.ucsd.edu

[†] University of California, San Diego [‡] University of Washington, Seattle

Abstract

We present a practical, stratified autocalibration algorithm with theoretical guarantees of global optimality. Given a projective reconstruction, the first stage of the algorithm upgrades it to affine by estimating the position of the plane at infinity. The plane at infinity is computed by globally minimizing a least squares formulation of the modulus constraints. In the second stage, the algorithm upgrades this affine reconstruction to a metric one by globally minimizing the infinite homography relation to compute the dual image of the absolute conic (DIAC). The positive semidefiniteness of the DIAC is explicitly enforced as part of the optimization process, rather than as a post-processing step.

For each stage, we construct and minimize tight convex relaxations of the highly non-convex objective functions in a branch and bound optimization framework. We exploit the problem structure to restrict the search space for the DIAC and the plane at infinity to a small, fixed number of branching dimensions, independent of the number of views.

Experimental evidence of the accuracy, speed and scalability of our algorithm is presented on synthetic and real data. MATLAB code for the implementation is made available to the community.

1 Introduction

This paper proposes practical algorithms that provably solve well-known formulations for both affine and metric upgrade stages of stratified autocalibration to their global minimum.

The affine upgrade, which is arguably the more difficult step in autocalibration, is succintly computable by estimating the position of the plane at infinity in a projective reconstruction, for instance, by solving the modulus constraints [19]. Previous approaches to minimizing the modulus constraints for several views rely on local, descent-based methods with random reinitializations. These methods are not guaranteed to perform well for such non-convex problems. Moreover, in our experience, a highly accurate estimate of the plane at infinity is imperative to obtain a usable metric reconstruction.

The metric upgrade step involves estimating the intrinsic parameters of the cameras, which is commonly approached by estimating the dual image of the absolute conic (DIAC). A variety of linear methods exist towards this end, however, they are known to perform poorly in the presence of noise [10]. Perhaps more significantly, most methods *a posteriori* impose the positive semidefiniteness of the DIAC, which might lead to a spurious calibration. Thus, it is important to impose the positive semidefiniteness of the DIAC *within* the optimization, not as a post-processing step.

This paper proposes global minimization algorithms for both stages of stratified autocalibration that furnish theoretical certificates of optimality. That is, they return a solution at most ϵ away from the global minimum, for arbitrarily small ϵ. Our solution approach relies on constructing efficiently minimizable, tight convex relaxations to non-convex programs, using them in a branch and bound framework [13, 27].

A crucial concern in branch and bound algorithms is the exponential dependence of the worst case time complexity on the number of branching dimensions. In this paper, we exploit the inherent problem structure to restrict our branching dimensions to a small, fixed, view-independent number, which allows our algorithms to scale gracefully with an increase in the number of views.

A significant drawback of local methods is that they critically depend on the quality of a heuristic intialization. We use chirality constraints derived from the scene to compute a theoretically correct initial search space for the plane at infinity, within which we are guaranteed to find the global minimum [8, 9]. Our initial region for the metric upgrade is intuitively specifiable as conditions on the intrinsic parameters and can be wide enough to include any practical cases.

While we provide a unified framework for a global solution to stratified autocalibration, in several ways, our constructions are general enough to find application in domains beyond multiple view geometry.

In summary, our main contributions are the following:

- Highly accurate recovery of the plane at infinity in a projective reconstruction by global minimization of the modulus constraints.

- Highly accurate estimation of the DIAC by globally solving the infinite homography relation.
- A general exposition on novel convexification methods for global optimization of non-convex programs.

The outline of the rest of the paper is as follows. Section 2 describes background relevant to autocalibration and Section 3 outlines the related prior work. Section 4 is a brief overview of branch and bound algorithms. Sections 5 and 6 describe our global optimization algorithms for estimating the DIAC and the plane at infinity respectively. Section 7 presents experiments on synthetic and real data and Section 8 concludes with a discussion of further extensions.

2 Background

Unless stated otherwise, we will denote 3D world points $\mathbf{X}$ by homogeneous 4-vectors and 2D image points $\mathbf{x}$ by homogeneous 3-vectors. Given the images of n points in m views, a projective reconstruction $\{\mathtt{P}, \mathbf{X}\}$ computes the Euclidean scene $\{\hat{\mathtt{P}}_i, \hat{\mathbf{X}}_j\}$ up to a 4×4 homography:

$$\mathtt{P}_i = \hat{\mathtt{P}}_i \mathtt{H}^{-1}\ ,\ \mathbf{X}_j = \mathtt{H}\hat{\mathbf{X}}_j.$$

A camera matrix is denoted by $\mathtt{K}[\mathtt{R}|\mathbf{t}]$ where $\mathtt{K}$ is an upper triangular matrix that encodes the intrinsic parameters and $(\mathtt{R}, \mathbf{t})$ constitute exterior orientation. A more detailed discussion of the material in this section can be found in [10].

2.1 The Infinite Homography Relation

We can always perform the projective reconstruction such that $\mathtt{P}_1 = [\mathtt{I}|\mathbf{0}]$. Let the first world camera be $\widehat{\mathtt{P}}_1 = \mathtt{K}_1\,[\mathtt{I}|\mathbf{0}]$. Then $\mathtt{H}$ has the form

$$\mathtt{H} = \begin{bmatrix} \mathtt{K}_1 & \mathbf{0} \\ -\mathbf{p}^\top \mathtt{K}_1 & 1 \end{bmatrix} \tag{1}$$

where $\pi_\infty = (\mathbf{p}, 1)^\top$ is the location to which the plane at infinity moves in the projective reconstruction from its canonical position $(0,0,0,1)^\top$. The aim of autocalibration is to recover the plane at infinity and the intrinsic parameters.

Let $\mathtt{P}_i = [\mathtt{A}_i|\mathbf{a}_i]$ where $\mathtt{A}_i$ is 3×3 and $\mathbf{a}_i$ is a 3×1 vector. Let $\widehat{\mathtt{P}}_i = \mathtt{K}_i\,[\mathtt{R}_i|\mathbf{t}_i]$. Then, for the case of constant intrinsic parameters ($\mathtt{K}_i = \mathtt{K}$), we have

$$\omega^* = (\mathtt{A}_i - \mathbf{a}_i\mathbf{p}^\top)\omega^*(\mathtt{A}_i - \mathbf{a}_i\mathbf{p}^\top)^\top \tag{2}$$

where $\omega^* = \mathtt{K}\mathtt{K}^\top$ denotes the dual image of the absolute conic (DIAC). Equality here is up to a scale factor. Since $\mathtt{H}^i_\infty = (\mathtt{A}_i - \mathbf{a}_i\mathbf{p}^\top)$ is precisely the homography induced between the views $\mathtt{P}_1 = [\mathtt{I}|0]$ and $\mathtt{P}_i = [\mathtt{A}_i|\mathbf{a}_i]$ by the plane at infinity $(\mathbf{p}, 1)^\top$, it is called the infinite homography.

2.2 Modulus Constraints

Noting that the infinite homography is conjugate to a rotation matrix and must have eigenvalues of equal moduli, one can relate the roots of its characteristic polynomial

$$\det(\lambda\mathtt{I} - (\mathtt{A}_i - \mathbf{a}_i\mathbf{p}^\top)) = \lambda^3 - \alpha_i\lambda^2 + \beta_i\lambda - \gamma_i. \tag{3}$$

to derive the so called *modulus constraint* [19]:

$$\gamma_i\alpha_i^3 = \beta_i^3, \tag{4}$$

where $\alpha_i, \beta_i, \gamma_i$ are affine functions of the coordinates $\{p_1, p_2, p_3\}$ of the plane at infinity. Three views suffice to restrict the solution space to a $4^3 = 64$ possibilities (actually 21, see [23]), which can in theory be extracted using continuation methods.

2.3 Chirality bounds on plane at infinity

Chirality constraints demand that the reconstructed scene points lie in front of the camera. While a general projective transformation may result in the plane at infinity splitting the scene, a quasi-affine transformation is one that preserves the convex hull of the scene points $\mathbf{X}$ and camera centers $\mathbf{C}$. A transformation $\mathtt{H}_q$ that upgrades a projective reconstruction to quasi-affine can be computed by solving the so-called chiral inequalities. A subsequent affine centering, $\mathtt{H}_a$, guarantees that the plane at infinity in the centered quasi-affine frame, $\mathbf{v} = (\mathtt{H}_a\mathtt{H}_q)^{-\top}\pi_\infty$, cannot pass through the origin. So it can be parametrized as $(v_1, v_2, v_3, 1)^\top$ and bounds on v_i in the centered quasi-affine frame can be computed by solving six linear programs:

$$\min / \max \quad v_i \quad \text{s.t.} \quad \mathbf{X}_j^{q\top}\mathbf{v} > 0,\ \mathbf{C}_k^{q\top}\mathbf{v} > 0, \tag{5}$$

where $\mathbf{X}_j^q$ and $\mathbf{C}_k^q$ are points and camera centers in the quasi-affine frame, $j = 1, \ldots, n$ and $k = 1, \ldots, m$. We refer the reader to [8, 18] for a thorough treatment of the subject.

3 Previous work

Approaches to autocalibration [6] can be broadly classified as direct and stratified. Direct methods seek to compute a metric reconstruction by estimating the absolute conic. This is encoded conveniently in the dual quadric formulation of autocalibration [12, 28], whereby an eigenvalue decomposition of the estimated dual quadric yields the homography that relates the projective reconstruction to Euclidean. Linear methods [20] as well as more elaborate SQP based optimization approaches [28] have been proposed to estimate the dual quadric, but perform poorly with noisy data. Methods such as [16] which are based on the Kruppa equations (or the fundamental matrix), are known to suffer from additional ambiguities [25].

This work primarily deals with a stratified approach to autocalibration [19]. It is well-established in literature that, in the absence of prior information about the scene, estimating the plane at infinity represents the most significant challenge in autocalibration [9]. The modulus constraints [19] propose a necessary condition for the coordinates of the plane at infinity. Local minimization techniques are used in [19] to minimize a noisy overdetermined system in the multi-view case.

An alternate approach to estimating the plane at infinity exploits the chirality constraints. The algorithm in [9] computes bounds on the plane at infinity and a brute force search is used to recover π_∞ within this region. It is argued in [18] that it might be advantageous to use camera centers alone when using chirality constraints.

Several linear methods exist for estimating the DIAC [10] for the metric upgrade, but they do not enforce its positive semidefiniteness. The only work the authors are aware of which explicitly deals with this issue is [2], which is formulated under the assumption of known principal point and zero skew. The interested reader is refered to [10] and the references therein for a more detailed overview of literature relevant to autocalibration.

Of late, there has been significant activity towards developing globally optimal algorithms for various problems in computer vision. The theory of convex LMI relaxations [15] is used in [14] to find global solutions to several optimization problems in multiview geometry, while [5] discusses a direct method for autocalibration using the same techniques. Triangulation and resectioning are solved with a certificate of optimality using convex relaxation techniques for fractional programs in [1]. An interval analysis based branch and bound method for autocalibration is proposed in [7], however the fundamental matrix based formulation does not scale well beyond a small number of views.

4 Branch and bound theory

Branch and bound algorithms are non-heuristic methods for global optimization in nonconvex problems [13]. They maintain a provable upper and/or lower bound on the (globally) optimal objective value and terminate with a certificate proving that the solution is ϵ-suboptimal (that is, within ϵ of the global optimum), for arbitrarily small ϵ.

Consider a multivariate, nonconvex, scalar-valued objective function $f(x)$, for which we seek a global minimum over a rectangle Q_0. Branch and bound algorithms require an auxiliary function $f_{\text{lb}}(Q)$ which for every region $Q \subseteq Q_0$, satisfies two properties. One, the value of $f_{\text{lb}}(Q)$ is always less than or equal to the minimum value $f_{\text{min}}(Q)$ of $f(x)$ for $x \in Q$. Two, if $|Q|$ denotes the size along the largest dimension, then for every sequence of rectangles Q_k such that $|Q_k| \rightarrow 0$, $f_{\text{lb}}(Q_k) \rightarrow f_{\text{min}}(Q_k)$ uniformly.

Computing the value of $f_{\text{lb}}(Q)$ is referred to as *bounding*, while choosing and subdividing a rectangle is called *branching*. The choice of the rectangle picked for refinement and the actual subdivision itself are essentially heuristic. We consider the rectangle with the smallest minimum of f_{lb} as the most promising to contain the global minimum and subdivide it into $k = 2$ rectangles along the largest dimension. A key consideration when designing bounding functions is the ease with which they can be estimated . So, it is desirable to design $f_{\text{lb}}(Q)$ as the solution of a convex optimization problem for which efficient solvers exist [4]. In the next two sections, we present branch and bound algorithms based on two such constructions.

Although guaranteed to find the global optimum (or a point arbitrarily close to it), the worst case complexity of a branch and bound algorithm is exponential. While this may initially appear to be discouraging, we will show in our experiments that exploiting problem structure leads to fast convergence rates in practice.

5 Globally optimal metric upgrade

For ease of exposition, we will first describe the metric upgrade step, that is, assume that the affine upgrade has already been performed. The problem of finding the plane at infinity for the affine upgrade is deferred to the next section.

5.1 Problem Formulation

Recall that when the camera intrinsic parameters are held constant, the DIAC satisifes the infinite homography relations $\omega^* = \mathtt{H}^i_\infty \omega^* \mathtt{H}^{i\top}_\infty,\ i = 1, \cdots, m$. Both ω^* and $\mathtt{H}^i_\infty$ are homogeneous quantities, so we must account for the scale in the above relation before it can be used to search for the optimal DIAC.

A necessary condition for the matrix ω^* to be interpreted as $\omega^* = \mathtt{KK}^\top$ is that $\omega^*_{33} = 1$. Thus, we fix the scale in the infinite homography relation by demanding that both the matrices on the left and the right hand side of the relation have their $(3,3)$ entry equal to 1. To this end, we introduce additional variables λ_i and pose the minimization problem:

$$\begin{aligned} \arg\min_{\omega^*,\lambda_i} \quad & \sum_i \|\omega^* - \lambda_i \mathtt{H}^i_\infty \omega^* \mathtt{H}^{i\top}_\infty\|_F^2 \qquad (6) \\ \text{s.t.} \quad & \omega^*_{33} = 1,\ \lambda_i \mathbf{h}^{i^\top}_3 \omega^* \mathbf{h}^i_3 = 1,\ \omega^* \succ 0,\ \omega^* \in \mathcal{D} \end{aligned}$$

Here, $\mathbf{h}^i_3$ denotes the third row of the 3×3 infinite homography $\mathtt{H}^i_\infty$ and $\mathcal{D}$ is some initial convex region within which the individual entries of ω^* lie.

5.2 Convex Relaxation

As discussed in Section 4, the success of a branch and bound algorithm critically depends on the quality of underestimators as well as the efficiency of their minimization. In this section, we construct a high quality convex relaxation of (6) that underestimates its minimum.

We begin by introducing a new set of variables $\nu_i = \lambda_i \omega^*$. Here each matrix ν_i is a symmetric 3×3 matrix with entries $\nu_{ijk} = \lambda_i \omega^*_{jk}$. Also let us assume that the domain $\mathcal{D}$

is given in the form of bounds $[l_{jk}, u_{jk}]$ on the five unknown symmetric entries ω^*_{jk} of ω^*. Then (6) can be re-written as

$$\begin{aligned} \arg \min_{\omega^*, \nu_i, \lambda_i} \quad & \sum_i \left\| \omega^* - \mathtt{H}^i_\infty \nu_i \mathtt{H}^{i\top}_\infty \right\|_F^2 & (7) \\ \text{s.t.} \quad & \nu_{ijk} = \lambda_i \omega^*_{jk},\ \lambda_i {\mathbf{h}^i_3}^\top \omega^* \mathbf{h}^i_3 = 1 \\ & \omega^* \succ 0,\ \omega^*_{33} = 1,\ l_{jk} \leq \omega^*_{jk} \leq u_{jk} \end{aligned}$$

The non-convexity in the above optimization problem has been reduced to the bilinear equality constraints $\nu_{ijk} = \lambda_i \omega^*_{jk}$ and $\lambda_i {\mathbf{h}^i_3}^\top \omega^* \mathbf{h}^i_3 = 1$.

Given bounds on the entries of ω^*, a relaxation of (7) is obtained by replacing the constraint $\lambda_i {\mathbf{h}^i_3}^\top \omega^* \mathbf{h}^i_3 = 1$ by a pair of linear inequalities of the form $L_i \leq \lambda_i \leq U_i$, where L_i and U_i are computed by simply inverting the bounds on $\mathbf{h}^{i\top}_3 \omega^* \mathbf{h}^i_3$. Thus, the lower bound L_i can be computed as the reciprocal of the result of the maximization problem:

$$\begin{aligned} \max_{\omega^*} \quad & {\mathbf{h}^i_3}^\top \omega^* \mathbf{h}^i_3 & (8) \\ \text{s.t.} \quad & \omega^*_{33} = 1,\ \omega^* \succeq 0,\ l_{jk} \leq \omega^*_{jk} \leq u_{jk} \end{aligned}$$

The upper bound U_i can be computed similarly by computing the reciprocal of the minimizer of the above. The relaxed optimization problem can now be stated as:

$$\begin{aligned} \arg \min_{\omega^*, \nu_i, \lambda_i} \quad & \sum_i \left\| \omega^* - \mathtt{H}^i_\infty \nu_i \mathtt{H}^{i\top}_\infty \right\|_F^2 & (9) \\ \text{s.t.} \quad & \nu_{ijk} = \lambda_i \omega^*_{jk},\ \omega^* \succ 0,\ \omega^*_{33} = 1 \\ & l_{jk} \leq \omega^*_{jk} \leq u_{jk},\ L_i \leq \lambda_i \leq U_i \end{aligned}$$

In effect, the above ensures that the introduction of an additional view does not translate into an increase in the dimensionality of our search space. Instead, the cost is limited to solving a small SDP to compute bounds on λ_i, while the number of branching dimensions remains equal to number of unknowns in ω^*, irrespective of the number of views.

Appendix A.2 discusses the synthesis of convex relaxations of bilinear equalities, which allows us to replace each bilinear equality by a set of linear inequalities. Using them, a convex relaxation of the above optimization problem can be stated as

$$\begin{aligned} \arg \min_{\omega^*, \nu_i, \lambda_i} \quad & \sum_i \left\| \omega^* - \mathtt{H}^i_\infty \nu_i \mathtt{H}^{i\top}_\infty \right\|_F^2 & (10) \\ \text{s.t.} \quad & \nu_{ijk} \leq U_i \omega^*_{jk} + l_{jk} \lambda_i - U_i l_{jk} \\ & \nu_{ijk} \leq L_i \omega^*_{jk} + u_{jk} \lambda_i - L_i u_{jk} \\ & \nu_{ijk} \geq L_i \omega^*_{jk} + l_{jk} \lambda_i - L_i l_{jk} \\ & \nu_{ijk} \geq U_i \omega^*_{jk} + u_{jk} \lambda_i - U_i u_{jk} \\ & l_{jk} \leq \omega^*_{jk} \leq u_{jk},\ L_i \leq \lambda_i \leq U_i \\ & \omega^*_{33} = 1,\ \omega^* \succeq 0 \end{aligned}$$

The objective function of the above optimization problem is convex quadratic. The constraint set includes linear inequalities and a positive semidefiniteness constraint. Such problems can be efficiently solved to their global optimum using interior point methods and a number of software packages exist for doing so. We use SeDuMi in this paper [24].

The user of the algorithm specifies valid ranges for the entries of the calibration matrix $\mathtt{K}$. From this input, we derive intervals $[l_{jk}, u_{jk}]$ for the entries ω^*_{jk} of the matrix ω^* using the rules of interval arithmetic [17].

6 Global estimation of plane at infinity

6.1 Problem Formulation

Given more than the minimal number of views and in the presence of noise, the modulus constraints must be solved in a least squares sense. Previous work [19] has attempted to estimate of the plane at infinity by minimizing $\sum_i (\gamma_i \alpha_i^3 - \beta_i^3)^2$, using non-linear local minimization techniques. This cost function is a polynomial and some recent work in computer vision [14, 5] exploits LMI relaxations to achieve global optimality in polynomial programs. However, this is a degree 8 polynomial in three variables, which is far beyond what present-day solvers [11, 22] can handle. We instead consider the equivalent formulation:

$$\min_{p_1, p_2, p_3} \sum_i (\gamma_i^{1/3} \alpha_i - \beta_i)^2 \tag{11}$$

6.2 Convex Relaxation

As an illustration of higher-level concepts, we show construction of convex underestimators for the non-convex objective in (11). The actual objective we minimize incorporates chirality bounds and is derived in Section 6.3.

Let us suppose it is possible to derive a convex underestimator $\operatorname{conv}({\gamma_i}^{1/3} \alpha_i)$ and concave overestimator $\operatorname{conc}({\gamma_i}^{1/3} \alpha_i)$ for $\gamma_i^{1/3} \alpha_i$. Then the following convex optimization problem underestimates the solution to (11).

$$\begin{aligned} \min_{p_1, p_2, p_3} \quad & \sum_i (s_i - \beta_i)^2 & (12) \\ \text{s.t.} \quad & \operatorname{conv}({\gamma_i}^{1/3} \alpha_i) \leq s_i \leq \operatorname{conc}({\gamma_i}^{1/3} \alpha_i) \end{aligned}$$

As shown in Appendix A.3, our convex and concave relaxations are piecewise linear and representable using a small set of linear inequalities. Thus the above optimization problem is a convex quadratic program that can be solved using a QP or an SOCP solver.

Given bounds on $\{p_1, p_2, p_3\}$, a branch and bound algorithm can now be used to obtain a global minimum to the modulus constraints. All that remains to be shown is that it is possible to estimate an initial region which bounds the coordinates of π_∞.

6.3 Incorporating the bounds on π_∞

One way to derive bounds on the coordinates of the plane at infinity is by using the chirality conditions overviewed in Section 2.3. Let $\mathbf{v}$ be the plane at infinity in the centered quasi-affine frame, where $\mathbf{v} = (v_1, v_2, v_3, 1)^\top$, so that we can find bounds on each v_i. However, the modulus constraints require that the first metric camera be of the form $\mathtt{K}\,[\mathtt{I}|\mathbf{0}]$ and the first projective camera have the form $[\mathtt{I}|\mathbf{0}]$, which might not be satisfiable in a centered quasi-affine frame, in general. Thus, we need to use the bounds derived in the centered quasi-affine frame within the modulus constraints for the original projective frame.

The centered quasi-affine reconstruction differs from the projective one by a transformation $\mathtt{H}_{qa} = \mathtt{H}_a \mathtt{H}_q$, where $\mathtt{H}_q$ takes the projective frame to some quasi-affine frame and $\mathtt{H}_a$ is the affine centering in that quasi-affine frame. Let $\mathbf{h}_i$ be the i-th column of $\mathtt{H}_{qa}$, then we have $p_i = \mathbf{h}_i^\top \mathbf{v} / \mathbf{h}_4^\top \mathbf{v}$. Recall that, for the j-th view, α_j, β_j and γ_j are affine expressions in p_1, p_2 and p_3 [19]. Then, for instance,

$$\alpha_j = \alpha_{j1} p_1 + \alpha_{j2} p_2 + \alpha_{j3} p_3 + \alpha_{j4} = \frac{a_j(\mathbf{v})}{d(\mathbf{v})}, \qquad (13)$$

where $a_j(\mathbf{v}) = \alpha_{j1} \mathbf{h}_1^\top \mathbf{v} + \alpha_{j2} \mathbf{h}_2^\top \mathbf{v} + \alpha_{j3} \mathbf{h}_3^\top \mathbf{v} + \alpha_{j4} \mathbf{h}_4^\top \mathbf{v}$ and $d(\mathbf{v}) = \mathbf{h}_4^\top \mathbf{v}$. Similary, let

$$\beta_j = \frac{b_j(\mathbf{v})}{d(\mathbf{v})}, \quad \gamma_j = \frac{c_j(\mathbf{v})}{d(\mathbf{v})} \qquad (14)$$

where $a_j(\mathbf{v}), b_j(\mathbf{v}), c_j(\mathbf{v}), d(\mathbf{v})$ are linear functions of $\mathbf{v}$. In the following, for the sake of brevity, we will drop the reference to $\mathbf{v}$ and just use a_j, b_j, c_j, d. Now the optimization problem (11) can be rewritten as

$$\min_{v_1,v_2,v_3} \frac{\sum_j \left(c_j^{1/3} a_j - d^{1/3} b_j \right)^2}{d^{8/3}}, \text{ s.t. } l_i \leq v_i \leq u_i \quad (15)$$

Introducing new scalar variables for some of the non-linear terms, the above is equivalent to

$$\begin{aligned} \min_{v_1,v_2,v_3} \quad & r & (16) \\ \text{s.t.} \quad & re \geq \sum_j (f_j - g_j)^2, \;\; l_i \leq v_i \leq u_i & \\ & f_j = c_j^{1/3} a_j, \; g_j = d^{1/3} b_j, \; e = d^{8/3} & \end{aligned}$$

As we did for the case of the metric upgrade, we have reduced the non-convexity in the above optimization problem to a set of equality constraints. The quadratic inequality constraint is convex and is known as a rotated cone [4]. Given bounds on v_i, it is easy to calculate bounds on a_j, b_j, c_j, d, by solving eight linear programs in three variables. Given these bounds, we can construct convex and concave envelopes of the non-linear functions, and use them to construct the following convex program that underestimates the minimum of the above problem.

$$\begin{aligned} \min_{v_1,v_2,v_3} \quad & r & (17) \\ \text{s.t.} \quad & re \geq \sum_j (f_j - g_j)^2, \; j = 1, \ldots, m & \\ & \operatorname{conv}\left(c_j^{1/3} a_j \right) \leq f_j \leq \operatorname{conc}\left(c_j^{1/3} a_j \right) & \\ & \operatorname{conv}\left(d^{1/3} b_j \right) \leq g_j \leq \operatorname{conc}\left(d^{1/3} b_j \right) & \\ & e \leq \operatorname{conc}\left(d^{8/3} \right), \; l_i \leq v_i \leq u_i & \end{aligned}$$

Notice that the convex envelope of $d^{8/3}$ is not needed. Since (17) is a minimization problem, e always takes its maximum possible value and does not require a lower bound.

Following Appendix A, our convex relaxation in (17) consists of a linear objective subject to linear and SOCP constraints, which can be efficiently minimized [24]. A branch and bound algorithm can now be used to obtain an estimate of $\{v_1, v_2, v_3\}$, which globally minimizes the modulus constraints. Thereafter, the plane at infinity in the projective frame can be recovered as $\pi_\infty = \mathtt{H}_{qa}^\top \mathbf{v}$, which completes the projective to affine upgrade.

7 Experiments

In this section, we will describe the experimental evaluation of our algorithms using synthetic and real data.

For evaluating performance metrics, we simulated a scene where 100 3D points are randomly generated in a cube with sides of length 20, centered at the origin and a varying number of cameras are randomly placed at a nominal distance of 40 units. Zero mean, Gaussian noise of varying standard deviation is added to the image coordinates. A projective transformation is applied to the scene with a known, randomly generated plane at infinity and the ground truth intrinsic calibration matrix is identity.

For various numbers of cameras and noise levels, errors in the estimation of the plane at infinity and the intrinsic parameters are tabulated in Table 1. The following metrics are defined for evaluation:

$$\Delta p = \sqrt{\sum_{i=1}^{3} (p_i / p_i^0 - 1)^2}, \quad \Delta f = \left| \frac{f_1 + f_2}{f_1^0 + f_2^0} - 1 \right|$$

$$\Delta uv = \left| (|u| + |v|) - (|u^0| + |v^0|) \right| / 2, \quad \Delta s = |s - s^0|$$

where p_i are estimated coordinates of the plane at infinity, f_1, f_2 represents the two focal lengths, (u, v) stands for the principal point and s for the skew. $p_i^0, f_1^0, f_2^0, u^0, v^0$ and s^0 are the corresponding ground truth quantities.

The accuracy of the algorithm is evident from the very low error rates obtained for reasonable noise levels. It is interesting that the algorithm performs quite well even for noise as high as 1%. In general, the accuracy improves as expected when a greater number of cameras are used.

The column $\pi_\infty(1)$ in Table 1 reports the number of branch and bound iterations using the algorithm described in 6.3. However, an additional optimization is possible: we can refine the value of the feasible point $f(q^*)$ using a gradient descent method within the rectangle that contains it. This does not compromise optimality, but allows the value of the current best estimate to be lower than the value corresponding to the minimum of the lower bounding function. The number of iterations with this refinement is tabulated under $\pi_\infty(2)$. The error metrics reported are computed using the refined algorithm, however, since both algorithms are run with the same very stringent tolerance ($\epsilon = 1e - 7$), the solutions obtained are comparable. The number of iterations for the DIAC estimation (with no refinement) are tabulated under ω^*. The metric upgrade setup was performed with $\epsilon = 1e - 5$.

Typical time for each iteration of branch and bound is a few seconds on a Pentium IV, 1 GHz computer with 1GB of RAM. For 0.5% noise, each iteration of π_∞ estimation takes about 4 seconds for 10 views and 14 seconds for 40 views. The corresponding times are 8 and 32 seconds for ω^* estimation. Our code is unoptimized MATLAB with an off-the-shelf SDP solver [24], so these timings should be understood only as rough qualitative indicators.

While the metrics above are intuitive for evaluating the intrinsic parameters, it is not readily evident how Δp must be interpreted. Towards that end, we perform a set of experiments, inspired by [19], where three mutually orthogonal 5×5 grids are observed by varying numbers of randomly placed cameras. Noise ranging from 0.1 to 1% is added to the image coordinates. Parallelism of the various grid lines is evaluated for the affine upgrade, while orthogonality and parallelism are measured in the metric reconstruction. The results are tabulated in Table 2(b).

Again, we observe that the algorithm achieves very good accuracy for reasonable noise and performs quite well even for 1% noise. With just 5 cameras, it is quite likely for the configuration to be ill-conditioned or degenerate, which causes the algorithm to break down in some cases.

To demonstrate performance on real data, we consider images of marker targets on two orthogonal walls (Figure 2(a)). Using image correspondences from detected corners, we perform a projective reconstruction using the projective factorization algorithm [26] followed by bundle adjustment. The normalization procedure and exact implementation follows the description in [10].

Bounds on the plane at infinity are computed using chirality (5). Focal lengths are assumed to lie in the interval $[500, 1500]$, principal point within $[250, 450] \times [185, 385]$ (image size is 697×573) and skew $[-0.1, 0.1]$. The plane at infinity and DIAC are estimated using our algorithm. While ground truth measurements for the scene are not available, we can indirectly infer some observable properties.

The coplanarity of the individual targets is indicated by the ratio of the first eigenvalue of the covariance matrix of their points to the sum of eigenvalues. This ratio is measured to be 3.1×10^{-6}, 4.1×10^{-5}, 6.2×10^{-5} and 4.1×10^{-4} for the four polygonal targets. The angle between the normals to the planes represented by two targets on the adjacent walls is 88.1° in our metric reconstruction. The same angle is measured as 89.8° in a reconstruction using [21]. The precise ground truth angle between the targets is unknown.

All the results that we have reported are for the raw output of our algorithm. In practice, a few iterations of bundle adjustment following our algorithms might be used to achieve a slightly better estimate.

8 Conclusions and further discussions

In this paper, we have presented globally minimal solutions to the affine and metric stages of stratified autocalibration. Although our cost function is algebraic, this is the first work that provides optimality guarantees for a scalable stratified autocalibration. We hope the methods presented here will be of use elsewhere in computer vision too.

Several important extensions to the methods introduced in this paper can be envisaged. For instance, an L_1-norm formulation will allow us to use an LP solver for the affine upgrade, making it possible to solve larger problems faster. Another important direction to pursue is the development of alternate bounding techniques for the plane at infinity besides chirality, which will allow us to use the simpler relaxation presented in Section 6.2.

Acknowledgments

The authors would like to thank Fredrik Kahl for several helpful discussions and providing data for the experiment with real images. We thank Marc Pollefeys for sharing the implementation of [21]. Manmohan Chandraker and David Kriegman were supported by NSF EIA-0303622. Sameer Agarwal was supported by NSF EIA-0321235, UW Animation Research Labs, Washington Research Foundation, Adobe and Microsoft. Serge Belongie was supported by NSF Career #0448615 and the Sloan Research Fellowship.

A Convex and Concave Relaxations

In this appendix, we briefly describe the convex and concave relaxations of the intermediate non-linear terms that were relaxed as part of the various convex relaxations in the main text. In each instance, the variables x and y take values in the intervals $[x_l, x_u]$ and $[y_l, y_u]$, respectively.

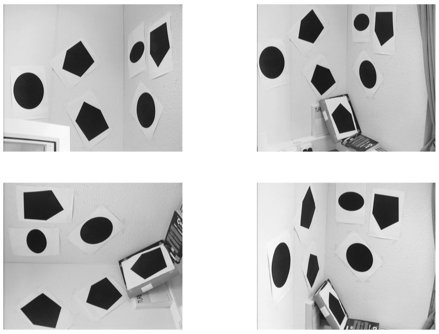
A.1 Functions of the form $f(x) = x^{8/3}$

The function $x^{8/3}$ is convex, and thus the line joining $(x_l, x_l^{8/3})$ and $(x_u, x_u^{8/3})$ is a tight concave overestimator, thus the relaxation is given by

$$z \leq x_l^{8/3} + \frac{x - x_l}{x_u - x_l}(x_u^{8/3} - x_l^{8/3}) \tag{18}$$

σ	m	Error				Iterations			Failure
(%)		Δp	Δf	Δuv	Δs	$\pi_\infty(1)$	$\pi_\infty(2)$	ω^*	(%)
0.1	5	3.13e-3	6.70e-3	4.34e-3	2.35e-3	22.5 ± 14.7	3.7± 4.4	37.5 ± 14.4	0
	10	8.45e-4	8.07e-4	1.02e-3	3.85e-4	14.5 ± 4.6	2.4 ± 2.4	40.9 ± 13.1	0
	20	8.39e-4	7.41e-4	3.90e-4	2.73e-4	14.4 ± 5.7	1.4 ± 1.2	31.8 ± 14.8	0
	40	7.65e-4	4.24e-4	2.82e-4	2.01e-4	13.8 ± 4.4	1.5 ± 3.2	27.5 ± 16.9	0
0.5	5	6.64e-3	7.50e-3	4.97e-3	2.33e-3	24.7 ± 15.3	4.5 ± 5.8	41.0 ± 16.2	0
	10	3.03e-3	2.51e-3	1.81e-3	1.20e-3	15.7 ± 5.6	3.0 ± 3.9	43.9 ± 15.9	0
	20	4.06e-3	2.13e-3	1.63e-3	1.10e-3	16.4 ± 10.5	4.1 ± 12.0	38.5 ± 11.6	0
	40	4.00e-3	1.69e-3	1.39e-3	7.09e-4	18.0 ± 9.0	9.3 ± 13.0	33.5 ± 11.7	0
1.0	5	1.09e-2	1.60e-2	7.03e-3	4.96e-3	25.0 ± 20.3	5.1 ± 10.4	42.7 ± 17.4	2
	10	5.81e-3	4.05e-3	3.01e-3	2.03e-3	18.6 ± 8.5	5.7 ± 8.8	44.3 ± 12.0	1
	20	8.16e-3	4.01e-3	2.44e-3	1.69e-3	21.5 ± 12.9	13.2 ± 17.2	43.6 ± 13.2	0
	40	7.80e-3	3.23e-3	2.37e-3	1.41e-3	24.3 ± 9.7	20.5 ± 13.7	43.3 ± 13.9	0

Figure 1. Error in camera calibration parameters for random synthetic data. The errors in the table are reported relative to ground truth for the indicated quantities. All quantities reported are averaged over a 100 trials. σ stands for percentage noise in image coordinates and m stands for number of views.



(a)

Noise	m	Affine	Metric		Failure
(%)		(Parallel)	(Parallel)	(Perpendicular)	(%)
0.1	5	0.49 ± 0.13	0.49 ± 0.12	0.45 ± 0.33	1
	10	0.31 ± 0.07	0.31 ± 0.07	0.24 ± 0.06	0
	20	0.21 ± 0.04	0.21 ± 0.04	0.17 ± 0.03	0
0.5	5	2.32 ± 0.50	2.33 ± 0.52	2.02 ± 0.87	3
	10	1.50 ± 0.30	1.50 ± 0.30	1.14 ± 0.25	0
	20	1.07 ± 0.17	1.07 ± 0.17	0.81 ± 0.15	0
1.0	5	5.34 ± 1.57	5.36 ± 1.63	5.70 ± 4.00	10
	10	3.17 ± 0.63	3.18 ± 0.63	2.48 ± 0.66	0
	20	2.05 ± 0.39	2.05 ± 0.38	1.60 ± 0.38	0

(b)

Figure 2. (a) Four of the twelve images consisting of marker targets that we use for reconstruction for our experiments with real data. (b) Error in affine and metric properties for three synthetically generated, mutually orthogonal planar grids. The table reports angular deviations from parallelism and orthogonality, measured in degrees. All quantities reported are averaged over 100 trials.

A.2 Bilinear functions $f(x, y) = xy$

We begin by considering convex and concave relaxations of the blinear function $f(x) = xy$. It can be shown [3] that the tightest convex lower bound for $f(x, y)$ is given by the function $z = \max\{x_l y + y_l x - x_l y_l, \quad x_u y + y_u x - x_u y_u\}$. Similarly, the tightest concave upper bounding function is given by $z = \min(x_u y + y_l x - x_u y_l, \quad x_l y + y_u x - x_l y_u)$. Thus, the convex relaxation of the equality constraint $z = xy$ over the domain $[x_l, y_l] \times [y_l, y_u]$ is given by

$$
\begin{aligned}
z \geq x_l y + y_l x - x_l y_l, &\quad z \geq x_u y + y_u x - x_u y_u \\
z \leq x_u y + y_l x - x_u y_l, &\quad z \leq x_l y + y_u x - x_l y_u \qquad (19)
\end{aligned}
$$

A.3 Functions of the form $f(x, y) = x^{1/3}y$

We now consider the construction of the convex relaxation for a bivariate function $f(x, y) = x^{1/3}y$.

Case 1: [$x_l > 0$ **or** $x_u < 0$**]** Suppose $x_l > 0$, then $f(x, y)$ is concave in x and convex in y. It can be shown [27] that the convex envelope for $f(x, y)$ is given by

$$
\begin{aligned}
\min \quad & z \geq (1 - \lambda) f(x_l, y_a) + \lambda f(x_u, y_b) \qquad (20) \\
\text{s.t.} \quad & x = x_l + (x_u - x_l)\lambda, \quad 0 \leq \lambda \leq 1 \\
& y = (1 - \lambda) y_a + \lambda y_b, \quad y_l \leq y_a, y_b \leq y_u
\end{aligned}
$$

Noting that $f(x, y) = x^{1/3}y$, substituting $y_p = (1 - \lambda)y_a$ and simplifying results in the following convex relaxation:

$$
\begin{aligned}
z \geq x_l^{1/3} y_p + x_u^{1/3}(y - y_p), &\quad (1 - \lambda) y_l \leq y_p \leq (1 - \lambda) y_u \\
\lambda y_l \leq y - y_p \leq \lambda y_u, &\quad \lambda = \frac{x - x_l}{x_u - x_l}. \qquad (21)
\end{aligned}
$$

A concave relaxation for $x^{1/3}y$ can be constructed by considering the negative of the convex envelope of $x^{1/3}(-y)$. This leads to

$$
\begin{aligned}
z \leq (x_u^{1/3} - x_l^{1/3}) y_p' + x_u^{1/3} y, &\quad \lambda y_u \geq y + y_p' \geq \lambda y_l \\
(1 - \lambda) y_u \geq -y_p' \geq (1 - \lambda) y_l, &\quad \lambda = \frac{x - x_l}{x_u - x_l}. \qquad (22)
\end{aligned}
$$

For the case when $x_u < 0$, we observe that a convex relaxation for $x^{1/3}y$ is given by the negative of the concave relaxation of $t^{1/3}y$, where $t = -x$. Appropriate manipulation of (21) gives us the convex and concave envelopes for this case too.

Case 2: **[**$x_l \leq 0 \leq x_u$**]** The function $x^{1/3}$ is convex for $x < 0$ and concave for $x > 0$. The derivation in (21) depends critically on the convexity of $x^{1/3}$ over its domain, and thus, cannot be used for the case when $x_l \leq 0 \leq x_u$. So instead of a one step relaxation, we will instead consider the two equality constraints $t = x^{1/3}$, $z = ty$ and relax each of them individually. Once we have the relaxation for $t = x^{1/3}$, we can then apply the bilinear relaxation to $z = ty$.

To construct a concave overestimator for $x^{1/3}$, we note that the line tangent to the curve $x^{1/3}$ and passing through $(x_l, x_l^{1/3})$ will always overestimate the curve. This line is given by $t = (4x - x_l)/3(-x_l)^{2/3}$. Further, we can show that the point of tangency is $(-x_l/8, -x_l^{1/3}/2)$. Now if $-x_l/8 < x_u$, then another line which upper bounds $x^{1/3}$ is the line passing through x_l and tangent at $(x_u, x_u^{1/3})$: $t = \frac{1}{3}x_u^{-2/3}x + \frac{2}{3}x_u^{1/3}$. Thus, the overestimator for $t = x^{1/3}$ is given by the minimum of these two lines.

Further, when $-x_l/8 \geq x_u$, we can get a better concave overestimator as simply the line passing through $(x_l, x_l^{1/3})$ and $(x_u, x_u^{1/3})$. Thus, the unified concave overestimator for $x^{1/3}$ is given by

$$t \leq \begin{cases} \min\left(\dfrac{4x - x_l}{3(-x_l)^{2/3}}, \dfrac{x + 2x_u}{3x_u^{2/3}}\right), & -x_l/8 < x_u \\ \dfrac{(x_u^{1/3} - x_l^{1/3})x + (x_u x_l^{1/3} - x_l x_u^{1/3})}{x_u - x_l}, & -x_l/8 \geq x_u \end{cases}$$

By similar arguments, we can derive the convex underestimator for $x^{1/3}$ when $x_l \leq 0 \leq x_u$ as

$$t \geq \begin{cases} \max\left(\dfrac{4x - x_u}{3x_u^{2/3}}, \dfrac{x + 2x_l}{3(-x_l)^{2/3}}\right), & -x_u/8 > l \\ \dfrac{(x_u^{1/3} - x_l^{1/3})x + (x_u x_l^{1/3} - x_l x_u^{1/3})}{x_u - x_l}, & -x_u/8 \leq x_l \end{cases}$$

References

[1] S. Agarwal, M. Chandraker, F. Kahl, S. Belongie, and D. Kriegman. Practical global optimization for multiview geometry. In *ECCV*, pages 592–605, 2006.

[2] M. Agrawal. On automatic determination of varying focal lengths using semidefinite programming. In *ICIP*, 2004.

[3] F. Al-Khayyal and J. Falk. Jointly constrained biconvex programming. *Math. Oper. Res.*, 8:273–286, 1983.

[4] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[5] M. Chandraker, S. Agarwal, F. Kahl, D. Nistér, and D. Kriegman. Autocalibration via rank-constrained estimation of the absolute quadric. In *CVPR*, 2007.

[6] O. Faugeras, Q.-T. Luong, and S. Maybank. Camera self-calibration: Theory and experiments. In *ECCV*, pages 321–334, 1992.

[7] A. Fusiello, A. Benedetti, M. Farenzena, and A. Busti. Globally convergent autocalibration using interval analysis. *PAMI*, 26(12):1633–1638, 2004.

[8] R. I. Hartley. Chirality. *IJCV*, 26(1):41–61, 1998.

[9] R. I. Hartley, E. Hayman, L. de Agapito, and I. Reid. Camera calibration and the search for infinity. In *ICCV*, pages 510–517, 1999.

[10] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[11] D. Henrion and J. B. Lasserre. GloptiPoly: Global optimization over polynomials with Matlab and SeDuMi. *ACM Trans. Math. Soft.*, 29(2):165–194, 2003.

[12] A. Heyden and K. Åström. Euclidean reconstruction from constant intrinsic parameters. In *ICPR*, pages 339–343, 1996.

[13] R. Horst and H. Tuy. *Global Optimization: Deterministic Approaches*. Springer Verlag, 2006.

[14] F. Kahl and D. Henrion. Globally optimal estimates for geometric reconstruction problems. In *ICCV*, pages 978–985, 2005.

[15] J. B. Lasserre. Global optimization with polynomials and the problem of moments. *SIOPT*, 11:796–817, 2001.

[16] R. Manning and C. Dyer. Metric self calibration from screw-transform manifolds. In *CVPR*, pages 590–597, 2001.

[17] R. E. Moore. *Interval Analysis*. Prentice-Hall, 1966.

[18] D. Nistér. Untwisting a projective reconstruction. *IJCV*, 60(2):165–183, 2004.

[19] M. Pollefeys and L. V. Gool. Stratified self-calibration with the modulus constraint. *PAMI*, 21(8):707–724, 1999.

[20] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *ICCV*, pages 90–95, 1998.

[21] M. Pollefeys, F. Verbiest, and L. J. V. Gool. Surviving dominant planes in uncalibrated structure and motion recovery. In *ECCV*, pages 837–851, 2002.

[22] S. Prajna, A. Papachristodoulou, and P. Parrilo. Introducing SOSTOOLS: a general purpose sum of squares programming solver. In *CDC*, 2002.

[23] F. Schaffalitzky. Direct solution of modulus constraints. In *ICVGIP*, pages 314–321, 2000.

[24] J. Sturm. Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. *Opt. Meth. and Soft.*, 11-12:625–653, 1999.

[25] P. Sturm. A case against Kruppa's equations for camera self-calibration. *PAMI*, 22(10):1199–1204, 2000.

[26] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *ECCV*, pages 709–720, 1996.

[27] M. Tawarmalani and N. Sahinidis. Convex extensions and envelopes of lower semi-continuous functions. *Math. Prog.*, 93(2):247–263, 2002.

[28] B. Triggs. Autocalibration and the absolute quadric. In *CVPR*, pages 609–614, 1997.