Epipolar rectification of a generic camera
Marc Deseilligny, Ewelina Rupnik

To cite this version:
Marc Deseilligny, Ewelina Rupnik. Epipolar rectification of a generic camera. 2020. hal-02968078

HAL Id: hal-02968078
https://hal.archives-ouvertes.fr/hal-02968078
Preprint submitted on 15 Oct 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Epipolar rectification of a generic camera

Marc Pierrot Deseilligny¹, Ewelina Rupnik¹

¹ LASTIG, Univ Gustave Eiffel, ENSG, IGN, F-94160 Saint-Mande, France
(marc.pierrot-deseilligny@ensg.eu, ewelina.rupnik@ign.fr)

Abstract

Keywords: epipolar rectification, generic camera, pushbroom sensor, central perspective

TODO:
1. dcrire mthode de calcul automatique des directions dans le cas sans modle;

2. Allgation : marche aussi sans modle, minimise globalement le critre d’erreur, formulation
thorique de l’ambiguit et des condition d’existence d’un crire pipolaire; mesure de ”l’pipolarabilit”
d’un couple;

1 Introduction

The epipolar geometry of images plays a central role in many applications in the field of photogrammetry and computer vision. In the stereo-reconstruction pipeline, it is used twice:

1. In the camera pose orientation step, when computing the relative orientation of a pair of images from their corresponding points. Assuming the projection follows the central perspective and the internal calibration is known, one can compute the epipolar geometry of the images using the essential matrix. Finally, the relative orientation is recovered [?].

2. In the image dense matching step, where the epipolar rectification simplifies the correspondence search because for any point \((x_1, y)\) in image \(I_1\), its correspondence is some point \((x_2, y)\) in image \(I_2\). Therefore, finding correspondences across images reduces to a 1-dimensional problem (1D).

In this paper, we only study the epipolar rectification problem and more specifically its application to a generic camera model.
1.1 Related works

Rectifying a central perspective camera stereo pair involves transforming their original epipolar geometry to a canonical form where: (a) their focal planes are coplanar, and (b) their conjugate epipolar lines are colinear, and parallel to the camera’s x-axis. From the algebraic standpoint, this is equivalent to applying two 2-dimensional (2D) projective transformations to both images of the stereo pair. Several approaches to computing such transformations have been proposed over the course of the last 30 years.

For a calibrated stereo pair (i.e. with known camera projection matrices), there exists a unique rectifying transformation, up to a rotation along the baseline \( \theta \). In an uncalibrated case, the solution is obtained by factoring out two 2D homographies from the fundamental matrix. Because there are no two unique homographies, the common practice is to parametrize these transformations such that the distortions caused by the rectification process are minimized. For instance, Loop and Zhang \([7]\) decompose the rectifying homographies to a combination of the projective, similarity and shearing transforms, with the condition that the projective transform remains (close to) affine. Hartley \([7]\) satisfies the condition that for a neighborhood of a point (e.g. the center of an image), the computed homography is a rigid transformation. Building on this work, Isgro and Trucco’s \([7]\) approach obtains a unique solution by minimizing the x-disparity without having to explicitly calculate the fundamental matrix. Instead of minimizing the disparity in the first coordinate, Wu and Yu \([7]\) recycle an idea first introduced by Hartley \([7]\) which requires that the aspect ratio of the images before and after rectification is constant. More recently, Fusiello and Irsara \([7]\) introduced the camera matrices back into the equation and proposed a quasi-Euclidean approach for uncalibrated cameras, similar to that of the calibrated cameras case. Their projective transformations are parametrized by five angles and a focal length. Monasse \textit{et al.} \([7]\) break down the one-time rotation of \([7]\) to a three step procedure, and prove increased robustness by using a geometric error measure (i.e., camera rotation angle) to reduce the rectifying error distortions.

Unlike the central projection camera model, pushbroom-like sensors acquire each image row from a different perspective center. As a consequence, the epipolar lines are neither straight lines, nor are they conjugate across the image \([7]\). One way to overcome this particularity is to simplify the projection function with a 2D affine \([7,7]\) or a parallel projection model \([7]\). Such approximations usually come at the price of precision, especially with the increasing camera field-of-view or in mountainous scenes. By extending the 2D affine model with two quartic terms, Okamonoto \textit{et al.} \([7]\) demonstrates improved performance on SPOT images. In the context of dense image matching, de Francis \textit{et al.} \([7]\) improves the precision by partitioning the images into small patches, for which independent affine rectifications are computed. Alternatively, and with equally good precision, Oh \([7]\) uses the Rational Polynomial Coefficients (RPCs) to map the epipolar curves across the full size images with lines, in a piecewise approach, followed by a global rectification transformation using a polynomial function of 3rd order.

1.2 Contributions

Our research work proposes an epipolar geometry rectification method that is not tied to any camera physical model. We demonstrate that on a range of models, including Pleiades images (pushbroom), the Corona images (panoramic pushbroom), a consumer grade camera images (central perspective) as well as Sentinel-1 radar images (pushbroom). The method resembles Oh’s \([7]\) approach in that it exploits the point correspondences to find the polynomial mapping to epipolar geometry. However, unlike the work of Oh \([7]\), we do not require that the camera geometric model is known. We demonstrate that in some circumstances, point correspondences obtained from an image processing routine, e.g. SIFT \([7]\), can serve to find the epipolar resampling.
In the remainder of this publication, we first outlay the mathematical background of the epipolar geometry and identify cases when epipolar geometry does not exist or is ambiguous (Section 2). Then, we introduce our method (Section 3), and finally present the experiments on different datasets with and without the geometric model (Section 4). The results are compared against the Oh’s [?] method with respect to the y-parallax remaining after the resampling to the epipolar geometry.

2 The mathematics of epipolar geometry in the generic case

2.1 Formalisation and notation of projections

\[ (i, j) = \pi(X, Y, Z) \]

\[ (X, Y, Z) \]

\[ B_{i,j} = \pi^{-1}(i, j) \]

Figure 1: A projection and a bundle.

We define the geometric sensor model of an image by a projection function \( \pi \), that computes, for a given 3D point, its 2D projection in the image:

**Definition 1** (Generic geometric sensor model). Illustrated in Figure 1.

A geometric sensor model \( \pi \) is a \( C^\infty \) mapping from ground space \((\mathbb{R}^3)\) to image space \((\mathbb{R}^2)\):

\[ \pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2, (X, Y, Z) \rightarrow (i, j) = \pi(X, Y, Z). \]  

Next, we define the bundles of a projection:

**Definition 2** (Bundle). For \( p_k \in I_k \) we note \( B_k(p_k) \) the bundle corresponding to \( \pi_k^{-1}(p_k) \). When there is no ambiguity, we note identically \( B_k(P) \), where \( P \in \mathbb{R}^3 \), the bundle corresponding to \( \pi_k^{-1}(\pi_k(P)) = B_k(\pi_k(P)) \).

Later, for simplicity, we will use the quasi-vertical hypothesis, which allows us to extend \( \pi \) to a bijective mapping of \( \mathbb{R}^3 \) and compute its inverse.

**Definition 3** (Quasi-vertical camera model). We say that the projection is quasi-vertical if the following mapping \( \tilde{\pi} \) is a diffeomorphism of \( \mathbb{R}^3 \):

\[ \tilde{\pi} : \mathbb{R}^3 \rightarrow \mathbb{R}^3, (X, Y, Z) \rightarrow (i, j, Z) = \tilde{\pi}(X, Y, Z), with(i, j) = \pi(X, Y, Z). \]  

Given 2 images \( I_1 \) and \( I_2 \), the knowledge of their geometric models \( \pi_1 \) and \( \pi_2 \) reduces the matching between 2 images to a 1D problem. In fact, given a point \( p_1 \) in \( I_1 \), we can compute the 3D curve \( B_1(p_1) \) of ground points that project to \( p_1 \) in \( I_1 \), and compute its homologous curve in \( I_2 \) with \( \pi_2(B_1(p_1)) \). We now define the H-compatible relation between two points by the following definition:
Definition 4 (H-Compatible, $\overset{\sim}{\pi_1,\pi_2}$). Illustrated in Figure 2.

We say that $p_1$ in $I_1$ and $p_2$ in $I_2$ are $\pi_1 - \pi_2$ H-compatible, and write $p_1\overset{\sim}{\pi_1,\pi_2} p_2$, if the following condition is satisfied:

$$(B_1(p_1) \cap B_2(p_2) \neq \emptyset) \iff (\exists P \in \mathbb{R}^3 : \pi_1(P) = p_1, \pi_2(P) = p_2).$$ (3)

In image matching, the relationship $p_1\overset{\sim}{\pi_1,\pi_2} p_2$ means that $p_1$ and $p_2$ are potentially homologous.

2.2 Definition of the epipolar geometry

In fact, the previous relationships are sufficient to implement all the matching techniques and $(\pi_1, \pi_2)$ can be used to define a matching process, taking advantage of the a priori knowledge of the scene geometry. That is, given a point in one image, we can easily follow its curve of potentially homologous points in the other image. This technique, which does not exploit the epipolar geometry, has the advantage of also being adaptable to multi-image matching. Epipolar geometry is therefore not strictly required for the image matching process.

The drawback of this approach is that it combines two different problems in the same procedure: the handling of the geometry and resampling and the matching process. When one is interested in the matching of a single image pair, the epipolar geometry can provide an elegant solution by separating the problem in two independent ones.

Definition 5 (Epipolar Geometry). Illustrated in Figures 3 and 6.

Let $\pi_1, \pi_2$ be two cameras and let $\phi_1, \phi_2$ be two diffeomorphisms of $\mathbb{R}^2$. We say that $\phi_1, \phi_2$ are epipolar resamplings iff:

$$\forall e_1 = (u_1, v_1), e_2 = (u_2, v_2) : (v_1 = v_2) \iff (\phi_1^{-1}(e_1) \overset{\sim}{\pi_1,\pi_2} \phi_2^{-1}(e_2)).$$ (4)

The matching of epipolar images is simplified because we know that the lines in two images are globally homologous.

Notation 1 (Epipolar line and curve.). We denote $\mathcal{L}_k(v)$ as the epipolar line of $E_k$ defined by $v_k = v$. We also denote $\mathcal{C}_k(v)$ as the epipolar curve of $I_k$ defined by $\mathcal{C}_k(v) = \phi_k^{-1}(\mathcal{L}_k(v))$

We can see that when epipolar geometry exists, the two curves $\mathcal{C}_1(v)$ and $\mathcal{C}_2(v)$ are globally homologous:

$$\mathcal{C}_1(v) = \pi_1(\pi_2^{-1}(\mathcal{C}_2(v))); \mathcal{C}_2(v) = \pi_2(\pi_1^{-1}(\mathcal{C}_1(v))).$$ (5)
2.3 Existence of the epipolar geometry

We now discuss the existence of the epipolar geometry. As you will see, the epipolar geometry generally does not exist, and when it does, it is not unique. It is well known that:

- for any image pair following the central projection, there exists an epipolar geometry;
- not all image pairs can be resampled to epipolar geometry. For example, a cylindrical projection, applicable to many push-broom satellites, generally does not allow for epipolar resampling.

We explain now why the epipolar geometry does not exist for any \( \pi_1, \pi_2 \) and is instead an exception. Let’s define the surface \( S^k_v \) of \( \mathbb{R}^3 \) by:

\[
S^k_v = \pi_1^{-1}(C_k(v)).
\] (6)

By the definition of epipolar geometry above, it can be seen that \( S^1_v \) and \( S^2_v \) are the same surface \( S_v \):

\[
S^1_v = S^2_v = S_v.
\] (7)

For any \( P \in S^1_v \), set \( e_1 = \phi_1(\pi_1(P)) = (u_1, v) \) and \( e_2 = \phi_2(\pi_2(P)) = (u_2, v) \). We then have \( \pi_1(P)^{\pi_1 \rightarrow \pi_2 \pi_2(P)} \) because they are projections of the same point. Then, \( v_2 = v \) according to Definition 5 and \( P \in S^2_v \). Furthermore, the \( S_v \) defines a foliation of \( \mathbb{R}^3 \), and it can be seen that:

\[
\forall v \forall P \in S_v : B_1(P) \subset S_v, B_2 \subset S_v,
\] (8)

which also leads directly from the definitions above. If \( P \in S_v \) then \( \pi_k^{-1}(P) \in C_k(v) \) (see equation (6)), then \( \pi_k^{-1}(\pi_k(P)) = B_k(P) \subset S_v \). However, in general, the existence of a stable foliation for the two bundle sets, as expressed in Equation (8), cannot be satisfied, as illustrated in Figure 4. To explain further, let \( \pi_1 \) and \( \pi_2 \) be any two projections and suppose there exists a foliation satisfying the Equation (8). Then, let:

- \( P \) be any point in 3D space, and \( S_v \) be the surface such that \( P \in S_v \);
- \( P_1 \neq P \) be a point on \( B_1(P) \), \( P_1 \in S_v \), then \( B_2(P_1) \subset S_v \);
• $P_2 \neq P$ be a point on $\mathcal{B}_2(P)$, $P_2 \in S_v$, then $\mathcal{B}_1(P_2) \subset S_v$.

As $\mathcal{B}_2(P_1)$ and $\mathcal{B}_1(P_1)$ are included in the same surface $S_v$, they must intersect somewhere in a point $Q$. In the general case, the above is a contradiction because there is no reason that the condition $\mathcal{B}_2(P_1) \cap \mathcal{B}_1(P_2) \neq \emptyset$ is satisfied for any two sets of bundles (see Figure 4, right).

Figure 4: Path closure. Left: in the epipolar case, the bundles are on the same level of the foliation and the intersection. Right: in the generic case, the paths don’t intersect and no epipolar geometry exists.

2.4 A local characterization of the epipolar existence

In this section, we compute a local formula (i.e. a differential equation) that provides conditions for the existence of an epipolar geometry. This section is rather theoretical and can be omitted by readers mainly interested in practical applications.

Analogously to the proof in Section 2.3, we will make a computation of two-way paths, $\mathcal{B}_1$ then $\mathcal{B}_2$, as well as $\mathcal{B}_2$ then $\mathcal{B}_1$. Then, we express the Taylor expansion of the intersection distance between these two paths. Let’s consider the following (see Figure 5):

• we make the quasi-vertical assumption (note that we could use curvilinear abscissa when this assumption can not be satisfied);

Figure 5: Notation for local characterization of the epipolar existence
• let \( P \) be any point in \( \mathbb{R}^3 \);
• consider the first path \((P, P_1, Q_1)\) following \( \mathcal{B}_1 \) then \( \mathcal{B}_2 \), making a progression \( d_{z_1} \) on \( \mathcal{B}_1 \) and \( d_{z_2} \) on \( \mathcal{B}_2 \);
• consider a second path \((P, P_2, Q_2)\) following \( \mathcal{B}_2 \) then \( \mathcal{B}_1 \), making a progression \( d_{z_2} \) on \( \mathcal{B}_1 \) and \( d_{z_1} \) on \( \mathcal{B}_2 \);
• we denote \( t_1(P) = (x, y, 1) \) as the tangent to the bundle \( \mathcal{B}_1 \) in point \( P \) (and similarly \( t_2(P) \));
• when we write \( \frac{\partial F}{\partial z_1} \), we will refer to the coordinate system \((i_1, j_1, z) = \pi_1^{-1}(x, y, z)\), idem for \( \frac{\partial F}{\partial z_2} \), and obviously as they are two different coordinate systems, we have in general \( \frac{\partial F}{\partial z_1} \neq \frac{\partial F}{\partial z_2} \).

Now, for any pair of "small" values \((\delta_1, \delta_2)\), we compute \((\delta_1', \delta_2')\) which minimize the distance \(|Q_1, Q_2|\) and express the canceling of the second degree Taylor expansion of this distance (the first degree can always be canceled out as we will see). Noting \( \delta \) the max of all \( \delta \), the second degree Taylor expansion gives:

\[
P_1 = P + \delta_1 t_1(P) + \frac{\delta_1^2}{2} \frac{\partial t_1}{\partial z_1}(P) + O(\delta^3)
\]

\[
Q_1 = P + \delta_2 t_2(P_1) + \frac{\delta_2^2}{2} \frac{\partial t_2}{\partial z_2}(P_1) + O(\delta^3)
\]

\[
t_2(P_1) = t_2(P) + \delta_1 \frac{\partial t_2}{\partial z_1}(P) + O(\delta^2)
\]

Putting together Equations (9), (10), (11) we can perform a Taylor expansion of the path \( P \) to \( Q_1 \):

\[
Q_1 = P + \delta_1 t_1(P) + \delta_2 t_2(P) + \frac{\delta_1^2}{2} \frac{\partial t_1}{\partial z_1}(P) + \frac{\delta_2^2}{2} \frac{\partial t_2}{\partial z_2}(P) + \delta_1 \delta_2 \frac{\partial t_2}{\partial z_1}(P) + O(\delta^3)
\]

And similarly for \( P \) to \( Q_2 \):

\[
Q_2 = P + \delta_2 t_2(P) + \delta_1 t_1(P) + \frac{\delta_2^2}{2} \frac{\partial t_2}{\partial z_2}(P) + \frac{\delta_1^2}{2} \frac{\partial t_1}{\partial z_1}(P) + \delta_1 \delta_2 \frac{\partial t_1}{\partial z_1}(P) + O(\delta^3)
\]

The first degree Taylor expansion of \( Q_2 - Q_1 \) gives:

\[
Q_2 - Q_1 = (\delta_1' - \delta_1) t_1(P) + (\delta_2' - \delta_2) t_2(P) + O(\delta^2)
\]

(14)

To minimize \(|Q_2 - Q_1|\), the first step is to cancel the first degree terms of \( Q_2 - Q_1 \). We assume that \( t_2(P) \) and \( t_1(P) \) are independant vectors\(^1\) and we must then make \( \delta_2' - \delta_2 \) and \( \delta_1' - \delta_1 \) second degree terms:

\[
\Delta_1 = \delta_1' - \delta_1 = O(\delta^2) \quad \Delta_2 = \delta_2' - \delta_2 = O(\delta^2)
\]

(15)

To develop \( Q_2 - Q_1 \) we can use the following identities that are direct consequences of Equation (15):

\[
\delta_1 \delta_2 - \delta_1' \delta_2' = O(\delta^3) \quad \delta_1^2 - \delta_1'^2 = O(\delta^3) \quad \delta_2^2 - \delta_2'^2 = O(\delta^3)
\]

(16)

\(^1\)Otherwise, it would be a degenerate case for stereovision
Subtracting Equation (12) from Equation (13), and using Equation (16), we can write:

\[ Q_2 - Q_1 = \Delta_1 \overrightarrow{t_1}(P) + \Delta_2 \overrightarrow{t_2}(P) + \delta_1 \delta_2 (\partial_{t_2} \overrightarrow{t_2}(P) - \partial_{t_1} \overrightarrow{t_1}(P)) + \mathcal{O}(\delta^3) \]  

(17)

We now translate the intersection of paths by canceling the second degree Taylor expansion in \( Q_2 - Q_1 \). We have three vectors, and their weighted sum can be null iff they are colinear.

**Theorem 1** (Existence of epipolar). The epipolar geometry exists iff the following determinant is null:

\[ \begin{vmatrix} \overrightarrow{t_1} & \overrightarrow{t_2} & \partial_{\overrightarrow{t_2}} \overrightarrow{t_2} - \partial_{\overrightarrow{t_1}} \overrightarrow{t_1} \end{vmatrix} = 0 \]  

(18)

**Remark 1** (Epipolar equation with central perspective camera). As an illustration in an easy case, we can see that this condition is trivially satisfied for a pair of central perspective cameras as we have the canceling of both terms as shown in Equation (19). This is because for a given point \( P \), for any point \( P_1 \) on \( B_1(P) \), \( t_2P_1 \) belongs to the epipolar plane \( \mathcal{P} \). We have \( t_2P_1 \in \mathcal{P} \), so \( \partial_{\overrightarrow{t_2}} \overrightarrow{t_2} \in \mathcal{P} \), and as we have also \( t_1(P) \in \mathcal{P}, t_2(P) \in \mathcal{P} \), the collinearity between \( t_1(P) \), \( t_2(P) \) and \( \partial_{\overrightarrow{t_2}} \overrightarrow{t_2}(P) \) is thus proven.

\[ \begin{vmatrix} \overrightarrow{t_1} & \overrightarrow{t_2} & \partial_{\overrightarrow{t_2}} \overrightarrow{t_2} \end{vmatrix} = \begin{vmatrix} \overrightarrow{t_1} & \overrightarrow{t_2} & \partial_{\overrightarrow{t_1}} \overrightarrow{t_1} \end{vmatrix} = 0 \]  

(19)

### 2.5 Ambiguity of the epipolar geometry

When the epipolar geometry exists, the epipolar resampling is not unique. To demonstrate that our rectification method handles this ambiguity rigorously, we first describe it formally.

Let \( \phi_1, \phi_2 \) and \( \phi'_1, \phi'_2 \) be two epipolar resamplings, then, for any \( v \) consider the pair of lines \( \mathcal{L}_1(v), \mathcal{L}_2(v) \):

- \( \phi_k^{-1}(\mathcal{L}_k(v)) \) is the curve \( \mathcal{C}_k(v) \) by definition of epipolar resampling;
- and \( \phi_k(\mathcal{C}_k(v)) = \phi_k(\phi_k^{-1}(\mathcal{L}_k(v))) \) is a line, also by definition of epipolar resampling;
- and still by definition \( \phi'_1(\phi_1^{-1}(\mathcal{L}_1(v))) = \phi'_2(\phi_2^{-1}(\mathcal{L}_2(v))) \);

Consequently we have the following constraint between two pairs of epipolar resampling:

- \( \phi'_1 \phi_1^{-1} \) and \( \phi'_2 \phi_2^{-1} \) are diffeomorphisms transforming lines into lines;
- \( \phi'_1 \phi_1^{-1} \) and \( \phi'_2 \phi_2^{-1} \) define the same global transformation on lines (i.e. if \( \phi'_1(\phi_1^{-1}(\mathcal{L}_1(v))) = \phi'_2(\phi_2^{-1}(\mathcal{L}_2(v))) \)).

*Vice versa*, let \( \phi_1, \phi_2 \) be an epipolar resampling and let \( \Lambda_1, \Lambda_2 \) be diffeomorphisms that are stable for lines and make globally the same transformation on lines. We can thus note that \( \Lambda_1 \circ \phi_1 \) and \( \Lambda_2 \circ \phi_2 \) are also an epipolar resampling.

Having devised the exact ambiguity, we can now define two constraints to impose on a unique epipolar resampling:

1. Constraint on the uniqueness of the deformation inside each line . For instance, one can impose that the columns remain constant (i.e. the deformation is only made on \( y \)), as given in Equations (20) and (21);

2. Constraint on the global deformation of lines\(^2\). For instance, by fixing the transformation of one image, as given in Equation (6).
Figure 6: Ambiguity of the epipolar geometry: two possible epipolar resamplings for a single stereopair.

**Theorem 2** (Unique epipolar constraint). If the epipolar geometry exists, there exists a unique epipolar resampling $\phi_1, \phi_2$ satisfying the following three constraints:

\[ \phi_1(x, y) = (x, y') \]  \hfill (20)
\[ \phi_2(x, y) = (x, y') \]  \hfill (21)
\[ \phi_1(0, y) = (0, y) \]  \hfill (22)

3 Proposed method for epipolar geometry resampling

3.1 Hypothesis and layout

3.1.1 Principles

The principle of the method is to use *H-Compatible* points $p_1, p_2$ to calculate a pair of functions $\phi_1, \phi_2$ that comply with the epipolar constraint, i.e. "$\phi_1(p_1)$ and $\phi_2(p_2)$ are on the same line". As these epipolar functions are not unique, we parameterize the $\phi_k$ in Theorem 2 accordingly:

\[ \phi_k(i, j) = (i, V_k(i, j)); \quad V_k : \mathbb{R}^2 \rightarrow \mathbb{R} \]  \hfill (23)

This parametrization implements the constraints of Equations (20) and (21). We will account for the constraint of Equation (22) in Section 3.2.1\(^2\). To compute $V_1, V_2$, for any pair of *H-Compatible*

\[^2\text{i.e. where each line is transformed globally to another line}\]

\[^3\text{See Equations (29), (30).}\]
points, we add an observation that constrains $V_1$ and $V_2$:

$$V_1(p_1) = V_2(p_2).$$

### 3.1.2 Hypothesis

The method takes two camera models $\pi_1$ and $\pi_2$ as inputs. These models are considered black-boxes that satisfy Equation (1), and for which no specific assumption is made on the physical model of the camera. In our C++ implementation, the cameras are considered to be pure virtual classes offering the interface to Equation (1). In this paper, the examples processed by our method are pushbroom satellite models known by their RPCs, the central perspective and radar models. However, the only restriction imposed on the generic nature of the model is that the projection function is "smooth", i.e.:

- $\hat{\pi}$ are $C^\infty$ functions, and
- the directions of epipolar curves vary within a limited range (for example, less than $\frac{\pi}{2}$).

Figure 7 illustrates the latter constraint. The left image presents a set of epipolar lines with too large direction variations. The right image represents a pair of epipolar lines whose directions change within a small range, therefore suitable for the proposed resampling method.

### 3.1.3 Estimation of the center and the global direction

To begin with, the method estimates the centers $C_1, C_2$ of a set of points $p_1$ and $p_2$. This is done by calculating the average of all points’ coordinates. Then, the computations continue in the coordinate systems centered at $C_1, C_2$. After this "normalisation", the constraint in Equation 22 is applied at these centers.

Then we need to compute a coordinate system where epipolar lines are globally horizontal. This requirement is a consequence of equation (23), and is illustrated by Figure 8:

- left image of figure 8 presents a case where epipolar curve are quasi vertical and for which an epipolar correction, without initial rotation, according to equation (23) would be impossible;
middle image of figure 8 presents a case where epipolar curve are oblique, in this case epipolar correction according to equation (23) would be possible but would lead to important distortion in the image, as can be seen on left image.

So for each image we estimate the average direction $\vec{D}_k$ of its epipolar lines, and using Equation (25), a rotation $R_k$ is applied to the input point.

$$R_k(p) = \frac{p - C_k}{\vec{D}_k}$$

The epipolar lines are now globally horizontal and the subsequent epipolar deformation can be computed on the rotated data points.

3.1.4 Layout

The layout of the method follows three steps: (1) estimate the global direction of epipolar lines; (2) estimate $F_1, F_2$ as the local epipolar rectification in the coordinate system linked to the global direction; (3) estimate the final epipolar rectification as a composition of $F_1, F_2$ and the rotation. A more formalized description of the algorithm is given in Algorithm 1.

Algorithm 1 Epipolar($\pi_1, \pi_2$). Layout of the algorithm for computing the epipolar rectification from camera models

Use $\pi_1, \pi_2$ to estimate a set of H- Compatible points $\mathcal{H} = \{(p_1, p_2)\}$:

- Estimate centers $C_1$ and $C_2$;
- Estimate global direction of epipolars $\vec{D}_1$ and $\vec{D}_2$;
- Estimate rotations $R_1, R_2$ according to Equation (25)

for all $p_1, p_2 \in \mathcal{H}$ do

- set: $q_1 = R_1(p_1)$, $q_2 = R_2(p_2)$
- add equation: $V_1(q_1) = V_2(q_2)$

end for

estimate with the least squares method $V_1$ and $V_2$

set $F_k(x, y) = (x, V_k(x, y))$

set $\phi_k = F_k \circ R_k$

return $(\phi_1, \phi_2)$

3.1.5 Why does our method work?

Intuitively, it may not be obvious that the system of equations in Equation (24) is well posed. In fact, if there was a functional relationship between $p_1$ and $p_2$, as $p_1 = F(p_2)$, an infinity of solutions for $(V_1, V_2)$ would exist, because for any function $V : \mathbb{R}^2 \rightarrow \mathbb{R}$ we can generate a solution $(V, V \circ F)$. However, note that due to the 3D aspect of $p_1$ and $p_2$, there is no functional relationship between them and, consequently, there are more constraints on $(V_1, V_2)$. Instead of a functional relationship, we can generate "one to many" (and "many to one") correspondences as illustrated in Figure 9. For example, for a given point $p_1$, following the curve $\pi_2(B_1(p_1))$, we can generate several points on the bundle (potentially an infinity) which results in many correspondences. To illustrate, if we take $p_2^k$ to be multiple homologous points of $p_1$, we then have the equation:

$$V_1(p_1) = V_2(p_2^1) ; V_1(p_1) = V_2(p_2^2) ; V_1(p_1) = V_2(p_2^3) \ldots ,$$

(26)
Marc Pierrot Deseilligny, Ewelina Rupnik

Figure 9: Left: for each \( p_1 \), we generate several 3D points on \( B_1(p_1) \). Middle: the multiple correspondences in \( I_2 \). Right: a dense network of curves in \( I_2 \).

which in fact enforces this constraint:

\[
V_2(p^1_1) = V_2(p^2_1) = V_2(p^3_1) \ldots
\]  

(27)

If we now look at left image of Figure 9, we see that Equation (27) imposes the constraint that a "piece of curve" is horizontal. In Section 3.3.1, we will see a more detailed analysis explaining how the method can work even with configurations different than those depicted in Figure 9.

3.2 Detailed implementation

3.2.1 Choice of a parametric functional space

We need to select a space of parametric functions to represent \( V_1, V_2 \). The only constraint is that \( V_1, V_2 \) are \( C^\infty \) functions, and that the additional constraint in Equation (22) is valid.

Classically, when parameterizing a set of functions \( C^\infty \), a "natural" candidate is the set of polynomials of a given degree. We know that the function will be \( C^\infty \) and, according to the Stone-Weierstrass theorem [?, ?] (which says that the space of polynomials is dense in the space of continuous functions), with a sufficiently high degree we will be able to accurately approximate any continuous function. A possible limitation of selecting high degree polynomial is over-fitting, which may lead to unwanted high frequency behavior. In our case, this problem should never arise as the measurements are synthesized from the projection functions \( \pi_1, \pi_2 \), which provides sufficient redundancy (for instance, hundreds of times more measurements than constraints). Note, this will be a different issue, and we will have to take care of degree, when we use the method with tie points and without model in section 3.3.1.

If \( d \) is the selected degree, we have two vectors of unknowns \( C^1_{a,b}, C^2_{a,b} \), corresponding to coefficients of the polynomials:

\[
V_k(p) = V_k(i,j) = \sum_{a=0}^{d} \sum_{b=0}^{d-a} C^k_{a,b} i^a j^b.
\]  

(28)

3.2.2 Imposing constraints on global lines deformation

When applying the constraint of equation (22) to equation (28), we have \( i = 0 \), thence we can suppress all terms \( i^a \) for \( a \neq 0 \). The constraint equation then reads:

\[
V_1(0,j) = j = \sum_{b=0}^{N} C^1_{0,b} j^b
\]  

(29)

In Equation (29), \( j \) on the left and the sum on the right are both polynomials, so if their functions are equal on a segment, they must be equal term by term. The constraint then comes to force a
number the unknowns $C^i_{0,k}$ which have known values: 1 for $C^i_{0,1}$ and 0 otherwise. Using the Kronecker delta, we can write:

$$C^i_{0,k} = \delta_{1,k}$$ (30)

### 3.2.3 Generation of points, computation of the direction and centers

The points from $\pi_1$ and $\pi_2$ are generated twice, using each image as the master. The bundles are always generated from the master images. The Algorithm 2 presents the generation of the points with $I_1$ as the master, as well as the computation of the global direction and the points’ centers.

**Algorithm 2** GenerateData(). Compute a list $L_{1,2}$ of $\pi_1 - \pi_2$ $H$-compatible pairs with $I_1$ as the master image. Compute also the center $C_1$ of points in $I_1$ and the global direction $\vec{D}_2$ for epipolar curves of $I_2$.

\[
\begin{align*}
L_{1,2} & \leftarrow () ; C_1 \leftarrow (0,0) ; \vec{D}_2 \leftarrow (0,0) ; N \leftarrow 0 \\
& \text{for } p_1.x = 0 \text{ to } X_1 \text{ Step } \delta_{x,y} \text{ do} \\
& \quad \text{for } p_1.y = 0 \text{ to } Y_1 \text{ Step } \delta_{x,y} \text{ do} \\
& \quad \quad \text{for } z = Z_0 \text{ to } Z_1 \text{ Step } \delta_z \text{ do} \\
& \quad \quad \quad p_2 = \pi_2(\pi_1^{-1}(p_1, Z)) \\
& \quad \quad \quad p'_2 = \pi_2(\pi_1^{-1}(p_1, Z + \delta_z)) \\
& \quad \quad \quad \text{if } p_2 \in I_2 \text{ and } p'_2 \in I_2 \text{ then} \\
& \quad \quad \quad \quad L_{1,2}.append((p_1, p_2)) \\
& \quad \quad \quad \quad C_1 \leftarrow C_1 + p_1 \\
& \quad \quad \quad \quad \vec{D}_2 \leftarrow \vec{D}_2 + \frac{p_2 - p'_2}{|p_2 - p'_2|} \\
& \quad \quad \quad N \leftarrow N + 1 \\
& \quad \quad \text{end if} \\
& \quad \text{end for} \\
& \text{end for} \\
& \text{end for} \\
& C_1 \leftarrow \frac{C_1}{N} ; \vec{D}_2 \leftarrow \frac{\vec{D}_2}{N}
\end{align*}
\]

Once the centers $C_1, C_2$, directions $\vec{D}_1, \vec{D}_2$ and the list $L_{1,2}$ are computed, they are used to normalize the measurements and make the direction globally horizontal by applying Equation (25) to all elements of the list.

### 3.2.4 Estimating the rectification

As the measurements are synthetic and without outliers, we can directly solve the equations with the linear least squares method, thus merging all previous steps:

- let $d$ be the degree of the polynomials;
- the unknowns are the coefficient of the polynomials $V_1$ and $V_2$. There are \(\frac{(d+1)(d+2)}{2}\) unknowns for $V_2$ and \(\frac{(d+1)(d+2)}{2} - (d + 1)\) for $V_1$, taking into account the constraint in Equation (30);
- for each pair of normalized points $q_1, q_2$ we add the Equation (28) to the least squares equation system.

We then estimate the $V_1, V_2$ and obtain:

$$\varphi_k(p) = \varphi_k(i,j) = (i, V_k(i,j)) ; \phi_k = \varphi_k \circ R_k$$ (31)
Estimating the inverse function The natural way to resample $I_k$ in $E_k$ is to write:

$$E_k(p) = I_k(\varphi_k^{-1}(p)).$$  \hspace{1cm} (32)

Therefore, to rectify an image, we also need to calculate the inverse function. The inverse of $R_k$ is obvious. For computing the inverse of $\varphi_k$, we exploit the fact that if $\varphi$ is invariant for the column, then $\varphi^{-1}$ is invariant too. Consequently, we can parametrize it with a function $W : \mathbb{R}^2 \rightarrow \mathbb{R}$ as:

$$\varphi_k^{-1}(p) = \varphi_k^{-1}(u, v) = (u, W_k(u, v)) \hspace{1cm} (33)$$

To estimate $W$, we follow the same rationale as in Section 3.2.1, and use the base of a polynomial function. Once the $V_k$ are known, we generate for each point $p_k = (i, j)$ in $L_{1,2}$ an observation:

$$W_k(i, V_k(i, j)) = j \hspace{1cm} (34)$$

If we want to ensure that the computed inverse is sufficiently close to the ”real” inverse, we can increase the polynomial’s degree (in our implementation, we typically use the degree of $d + 4$). It has no side effects as long as we maintain high redundancy.

3.3 Epipolar resampling without the geometric model – not too happy with this section me neither (;-)

3.3.1 Possibility with only tie-points

Ideally, there is no bullet-ing but full sentences.

"Is it possible to use the proposed method to compute the epipolar geometry if we have point correspondences between the image pairs but we don’t know the geometric model?" There is NO straightforward answer to whether it is possible or not. The answer is that is not possible in general, but it becomes possible when the relief is not smooth and we add the contraint that the ressampling is smooth enough.

**Pro:** Algorithm 1 uses only point correspondences and a direction to compute the epipolar geometry. It does not matter if the point correspondences are extracted from the geometric model (as with Algorithm 2) or from an image processing method, e.g. SIFT [?], which does not require any a priori information on image geometry. Hence, as long as we know the directions, our method can be used.

**Cons:** When the point correspondences are computed with an image processing routine, there exists some functional relationship between them. Suppose that:

- the 3D scene can be described by a function $Z = Z(X, Y)$, and denote $S^Z$ as the corresponding surface;
- for a point $p_1$ of $I_1$, denote $\tilde{\pi}^Z_1(p_1)$ as the intersection of $B_1(p_1)$ and the surface $S^Z$; $\tilde{\pi}^Z_1$ is the inverse of the restriction $\pi_1$ as function from $I_1$ to $S^Z$.

We now see that there exists a functional relationship between all the point correspondences $(p_1, p_2)$ and it follows the equation:

$$p_2 = (\pi_2 \circ \tilde{\pi}^Z_1)(p_1) = F^Z(p_1). \hspace{1cm} (35)$$

Therefore, as discussed in Section 3.1.5, in the most general case, it is impossible to recover the epipolar geometry from a set of correspondences.

**Pro & Cons:** If we go back to Section 3.1.5, we can see that we missed the fact that in the proposed method, the functions $V_1$ and $V_2$ have to be "smooth". Let’s reason again:
• suppose we have computed an epipolar geometry: \( e_k = (u_k, v_k) = \phi_k(p_k) = (x_k, V_k(y_k)) \), with \( v_1 = v_2 \) and \( V_k \) being a "smooth" function and we try to analyze if the geometry was ambiguous;

• we have \( e_2 = (\phi_2 \circ \pi_2 \circ \hat{\pi}_1^Z \circ \phi_1^{-1})(e_1) = P(e_1) \)

• we write the previous equation as \( (u_2, v_2) = P(u_1, v_1) = (u_1 + p_x(u_1, v_1), v_1) \), where \( p_x \) is what is usually called the "parallax function";

• as in Section 3.1.5, for any function \( W_2 : \mathbb{R}^2 \rightarrow \mathbb{R} \), let \( W_1 \) be the function defined by \( W_1 = W_2 \circ P \);

• then, for any \( e_1, e_2, (u_1, W_1(u_1, v_1)) \) and \( (u_2, W_2(u_2, v_2)) \), we also satisfy the epipolar constraint as \( W_1(u_1, v_1)) = W_2(P(u_1, v_1)) = W_2(u_2, v_2) \).

Is it possible that \( W_2 \) and \( W_2 \circ P \) are both smooth functions? This depends on the smoothness of \( P \). If \( P \) is itself a smooth function, then obviously for any smooth \( W_2 \), \( W_2 \circ P \) will also be smooth and the epipolar resampling will be ambiguous. If we take the canonical example of a flat 3D scene, then \( p_x = 0, P = Id \) and as \( W_1 = W_2, W_1 \) is smooth if \( W_2 \) is smooth. In this case the epipolar geometry is also ambiguous. However, if the scene has high frequency depth changes, then \( P \) also has high frequencies, and \( W_2 \) and \( W_2 \circ P \) cannot both be smooth. This can be seen more formally by the following equation:

\[
\frac{\partial W_2 \circ P}{\partial u} = \frac{\partial P}{\partial u} \frac{\partial W_2}{\partial u} \circ P \tag{36}
\]

As an archetype case, for any point where the scene is not differentiable, we have \( \frac{\partial P}{\partial u} = \infty \) and Equation (36) which lead to \( \frac{\partial W_2}{\partial u} = 0 \) because \( W_2 \circ P \) is supposed to be smooth. If \( W_2 \) is a polynomial of limited degree, and we have sufficient number of points with \( \frac{\partial W_2}{\partial u} = 0 \), then it will conduct to the fact that \( W_2 \) depends only of \( v \), adding finally equation (22), we have the \( W_2 = Id \), which show the uniqueness of the solution.

3.3.2 Estimating the directions without model

For using our methods, with tie-points, we need also a way to estimate the global directions, as it is done in section ?? when we have the geometric models.

In the implementation we provide, we give two options to the user: provide it or let the program compute it automatically. Obviously the providing option can be safer when the user has sufficient knowledge on the acquisition; it can be used for example with like satellite along the track, like in the example we provide with Corona acquisition. The automatic option is based on a discretization of possible directions and combinatorial exploration of all the pair of possible direction; for a given pair, a quality criterion is computed on residual of epipolar resampling with degree 0 TO CHECK !!!, using a \( L_1 \) norm (to be more robust to outliers).

3.4 Experiments

Datasets:

• Corona, +MPD, les courbes (teorique fait par Benjamin)

• Pleiades, residu
Table 1: Maximum value of the remaining y-parallax [pix] for acquisitions from a single orbit. $Z_{buff}$ corresponds to half the depth of the volume used in the resampling calculation.

| $Z_{buff}$ [m] | 0.1       | 0.15      | 0.25      | 0.45      | 0.1       | 0.15      | 0.25      | 0.45      |
|----------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 50             | 0.0070    | 0.0142    | 0.0156    | 0.0084    | 0.0017    | 0.0017    | 0.0017    | 0.0012    |
| 100            | 0.0065    | 0.0144    | 0.0157    | 0.0092    | 0.0019    | 0.0018    | 0.0017    | 0.0012    |
| 200            | 0.0057    | 0.0151    | 0.0164    | 0.0098    | 0.0027    | 0.0022    | 0.0022    | 0.0014    |
| 270            | 0.0055    | 0.0156    | 0.0144    | 0.0010    | 0.0036    | 0.0026    | 0.0026    | 0.0020    |

Table 2: Maximum value of the remaining y-parallax [pix] for acquisitions from multiple orbits. $Z_{buff}$ corresponds to half the depth of the volume used in the resampling calculation.

| $Z_{buff}$ [m] | 0.13      | 0.2       | 0.3       | 0.4       | 0.13      | 0.2       | 0.3       | 0.4       |
|----------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| 50             | 0.0297    | 0.0059    | 0.0178    | 0.0112    | 0.0021    | 0.0021    | 0.0021    | 0.0021    |
| 100            | 0.0284    | 0.0088    | 0.0183    | 0.0134    | 0.0029    | 0.0037    | 0.0034    | 0.0034    |
| 200            | 0.0325    | 0.0196    | 0.0252    | 0.0232    | 0.0064    | 0.0010    | 0.0086    | 0.0088    |
| 270            | 0.0381    | 0.0311    | 0.0343    | 0.0340    | 0.0101    | 0.0171    | 0.0143    | 0.0154    |

Table 3: Epipolarability indices for acquisitions from a single and multiple orbits.

| $Z_{buff}$ [m] | Single orbit |                      | Multiple orbits |
|----------------|--------------|----------------------|-----------------|
| 50-270         | $1.6e^{-5}$  | $0.8e^{-5}$          | $2.0e^{-5}$     |
| 50-270         | $0.3e^{-5}$  | $0.1e^{-5}$          | $0.7e^{-5}$     |
| 50-270         | $0.1e^{-5}$  | $0.2e^{-5}$          | $0.5e^{-5}$     |
| 50-270         | $0.2e^{-5}$  | $0.2e^{-5}$          | $0.2e^{-5}$     |

4.1 Satellite images
C'est la que c'est vraiment utile ....

4.2 Radar images
?

4.3 Images without the geometric model
Corona KH-4B images Expos precis avec modele analytique:
Epipolar rectification of a generic camera

* calcul des points homologues, en 3D = direcction moyenne * resolution par moindres carress avec degres lev, degr sup pour l’inverse * Utilit du 3D, precision en fonction de la nappes, possibilit d’utiliser un modle 3D grossier.

Figure 10: Corona KH-4B stereopair and the point correspondences (top); the epipolar curves in the original geometry of the stereopair (middle); and the stereopair resampled to the epipolar geometry, rotated by 90° (bottom).

Pléiades images

Consumer grade camera

5 Discussion and perspective

Avantage de la methode : modeles analylique, minimise deformation et residu, peut tre appliquee si on a que les points homologues (photo escalier ?).

Inconvenient ? Cas comme 7 pas gere, peut tre le sont il par Oh ?

Future work = use MNT ? Test sur configuration plus compliques : orbites differentes ? Radar ? Radar/visible ?
Figure 11: dddd consumer grade cam