Photogrammetric Techniques for Panoramic Cameras

Richard Hartley

GE - CRD, Room K1-5C39,
P.O. Box 8, Schenectady,
NY 12309
email: hartley@crd.ge.com

ABSTRACT

Panoramic cameras are used in aerial surveillance for the rapid coverage of large areas of terrain. Different Panoramic camera designs vary substantially, so that there is no such thing as a generic panoramic camera similar in generality to the pinhole model of a frame camera. This is true for instance for methods of forward motion compensation for which different methods are used. Nevertheless, a panoramic camera may be modelled as a camera with a cylindrical focal surface in which the image is acquired by sweeping a slit across the focal cylinder. This model fits most styles of panoramic camera.

In this paper, a general model for panoramic cameras is described including options of different methods of forward motion compensation. A Levenberg-Marquardt based parameter estimation program is used to estimate camera parameters from ground control points or image correspondences. The model has 16 parameters including ones for describing the orientation and location of the camera, the velocity of motion of the camera, forward motion control parameters as well as internal parameters such as scale, principal point offsets and digitizing parameters (for use if the panoramic image is digitized from film).

It is a peculiarity of the moving camera model that no closed form solution exists to determine the location of the image of a point in space -- unless the sweep axis is parallel with the direction of motion. If one permits "crabbing" due to cross winds the world-to-image mapping must be determined by iteration. This complicates the task of parameter solving.

An important feature of the parameter solution method is that no initialization of the camera parameters is necessary, except knowledge of the sweep direction, which is usually obvious since the image is far wider in the sweep direction than the cross-sweep direction. The parameter solving program will automatically find an accurate initial parameter estimation and refine it by iteration to the best solution.

This program has been used to orthorectify panoramic images for subsequent mosaicing.

1. PARAMETRIZED CAMERA MODELS.

A camera model describes a mapping from a three-dimensional world to a two-dimensional image. In denoting coordinates in 2 or 3-dimensional space column vectors will be used. For ease of in-line notation they will be written as transposed row vectors, such as $(x, y, z)^T$. The superscript T denotes transpose. Thus, a camera model describes a mapping from world coordinates $(x, y, z)^T$ to image coordinates, $(u, v)^T$. Usually the mapping depends on a number of numerical quantities called parameters. Parameters include such information as the location and orientation of the camera, as well as certain internal characteristics of a camera, such a focal length. Given specific values of the parameters, it is possible to compute the coordinates of the image of any world point $(x, y, z)^T$. Let the parametrized mapping function be denoted by F_P, where $P = \{p_j\}$ is the set of parameters. Then for a given choice of parameters, the F_P maps a world point $(x, y, z)^T$ to an image point $(u, v)^T = F_P(x, y, z)^T$.

Determination of the camera parameters is the inverse problem, given several correspondences of world coordinates $(x_i, y_i, z_i)^T$ to image coordinates $(u_i, v_i)^T$, of finding the best set of parameters to fit this measured data. A standard method of parameter determination is the Levenberg-Marquardt algorithm. We have developed an implementation of the Levenberg-Marquardt algorithm derived from the description in "Numerical Recipes in C"[1] and particularly optimized for determining the parameters of camera models.

Our implementation allows estimated or measured values to be specified for each of the world coordinates x_i, y_i or z_i, image coordinates u_i and v_i and parameters p_j. In addition, weights (related to estimated standard deviations) may optionally be assigned to each of these data entities. The program then seeks to find values of the world coordinates (x_i, y_i, z_i), image coordinates (u_i, v_i) and parameters p_j such that $(u_i, v_i)^T = F_P (x_i, y_i, z_i)^T$ and such that the weighted sum of squares

$$\sum_c w(c)(c - c^\circ)^2 + \sum_j w(p_j)(p_j - p_j^\circ)^2$$

where c is any one of the world or image coordinates, w(c) is its specified weight, c° is its specified estimated value and the coordinate c runs over all possible world or image coordinates, u_i, v_i, x_i, y_i and z_i. The second sum runs over all the parameters, p_j, and similar notation is used.

The weights may take value zero, effectively letting the data entity vary freely, or infinity, which fixes the value of the data to the specified value exactly. They may also take any intermediate value. By specifying estimated values of the parameters and corresponding weights (however small), one avoids the problem of instability due to over-parametrization of the problem, which can occur in such parameter estimation problems.

1.1. Determination of Initial Values.

As usual with iterative parameter estimation problems, the algorithm will converge to a minimum provided a sufficiently close initial estimate is provided for the parameters. If the user is required to estimate initial values for camera parameters, this can represent a considerable burden, since a good initial guess is not easy to find in many cases. In general, finding an initial estimate for camera parameters must be treated differently for each different type of camera model.

In the case of panoramic cameras, it proves possible to find a good initial guess by assuming a simplified camera model. This initial guess can then be used as a starting point for iteration using the full camera model. In particular, one assumes that the camera is stationary during the image acquisition and that the digitizing transform is a simple scaling. One may then apply non-iterative techniques to find the placement and orientation of the camera. The other parameters may then take default (usually zero) values as a basis for iteration. This approach has been used with consistent success. Details of the simplified camera model and the method used for estimation of the parameters of this model are given in section 3 of this paper.

The parameter estimation program and the panoramic model is part of a system for building navigational databases, the TARGET system developed at GE-CRD and GE-SCSD, the TARGET system developed at GE-CRD and GE-SCSD. It will handle several different types of image (such as panoramics, perspective images and pushbroom images) simultaneously, using image correspondences and ground control points to find the best fit for mosaicking the images.

2. PANORAMIC CAMERAS

There are several different models and styles of panoramic cameras. The Manual of Photogrammetry[2] shows a number of different types, which differ mechanically and optically. Nevertheless, panoramic cameras may all be modelled, at least to a first approximation in a consistent way. The basic panoramic camera may best be explained by comparing it with the standard perspective (pinhole) camera. Fig 1 shows the standard model of a perspective camera, in which the world is projected from a point (the centre of projection) onto a planar focal surface. Details of this model may be found in the Manual of Photogrammetry[2]. It is clear from this model that a perspective camera has a field of view less than 180°, and in fact usually much less.

The panoramic camera on the other hand may be modelled by projecting from a point, the centre of projection, onto a cylindrical focal surface. The centre of projection must lie on the axis of the ellipse. Usually, panoramic cameras have a field of view of less than 180°, so the focal surface is a half cylinder. The basic panoramic camera model is shown in Fig 2.

Although in fact the focal surface must lie behind the centre of projection (the lens), it is convenient for purposes of mathematical analysis to show the focal surface in front of the camera.

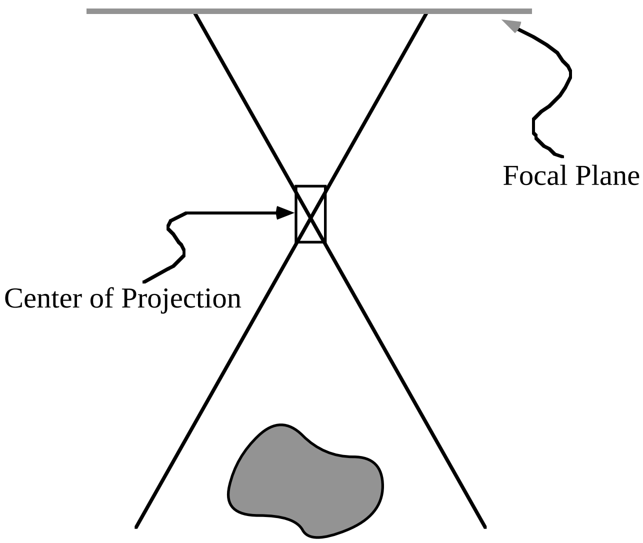

Fig 1. Ordinary perspective camera with planar focal surface.

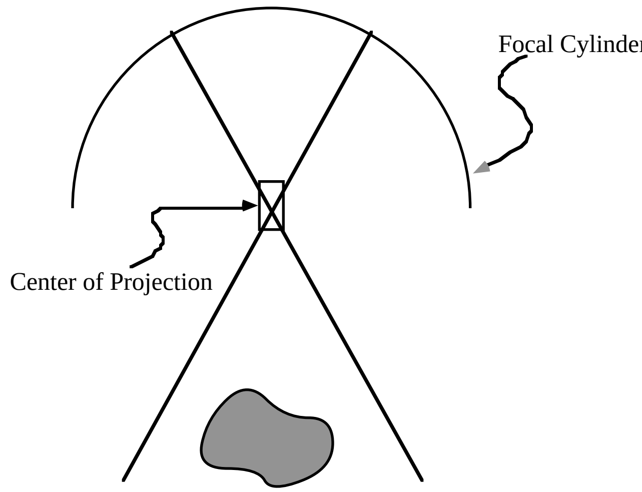

Fig 2. Panoramic camera has a cylindrical focal surface.

Actual panoramic cameras differ from this model in the details of their construction. The Manual of Photogrammetry[2] contains diagrams of different types of panoramic camera models. In some models the image is indeed projected onto a film placed against a cylindrical focal surface. In other models, a rotating prism is used. In some models the film moves during the acquisition of the image. Nevertheless, it may be verified, at least for all the model types described in the Manual of Photogrammetry that the imaging principle is equivalent to that shown in Fig 2.

A panoramic camera does not acquire the whole of its image at one time, but rather one line at a time as a lens assembly rotates around the axis of the focal cylinder. During the time of image acquisition, the camera platform (perhaps an aircraft) moves. This introduces distortions to the image. In addition various sorts of forward motion compensation, necessary to keep the image from smearing, introduce further distortions. These will be considered later. For the present, however, we will consider a simplified model in which it is assumed that the image is acquired instantaneously.

3. SIMPLIFIED PANORAMIC MODEL.

In analyzing the simplified panoramic model, we will use four sets of coordinate axes as follows.

Ground Coordinates: The first set of axes are the (X, Y, Z) axes, which are the coordinate axes for object space, and are fixed axes in which the positions of ground points are measured. These form a right handed orthonormal coordinate system. The Z-axis is thought of as pointing "down". Distances are measured in feet or metres.

Camera Coordinates: The second axes are the (X', Y', Z') axes, which are the axes of the panoramic camera fixture. The Y' axis is the axis of the focal cylinder and the other two axes are oriented to make a right-hand coordinate system. The Z' axis is somewhat arbitrarily chosen, but is thought of (somewhat vaguely) as pointing towards the centre of the image, in the same way that for perspective cameras, the Z' axis is usually chosen as the principal axis of the camera. The origin of the Camera Coordinates is at the centre of projection of the camera.

Focal Plane Coordinates: The third set of axes (U', V') define a two-dimensional rectilinear coordinate system wrapped around the focal cylinder with the V axis aligned with the Y' axis (the axis of the cylinder). The origin of the Focal Plane Coordinates is the point where the Z' axis meets the focal cylinder.

Image Coordinates: Finally, the image axes are denoted (U, V).

It will be assumed that the first three coordinate systems use the same measurement units (feet, metres, etc), whereas image coordinates are measured in pixels. Coordinates will be denoted in lower case using the same letter as the correponding axis labels. The relationships between the axes and their corresponding coordinates will now be considered. The goal is to define the mapping from Ground Coordinates to Image Coordinates.

First, Camera Coordinates are related to Ground Coordinates by a rigid rotation and displacement. This can be represented by a 3×4 matrix, M such that $(x', y', z')^T = M (x, y, z, 1)^T$. The matrix M may be decomposed into blocks as $M = (R \mid -Rt)$, where R is a 3×3 rotation matrix and t is a displacement vector.

Next, we consider the relationship of Camera Coordinates to Focal Plane Coordinates. Assume that the radius of the focal cylinder is *r* units. A point $(x', y', z')^T$ lying on the focal cylinder or radius r has Focal Plane Coordinates $(u', v')^T$ where

$$u' = r \arctan(x'/z') \quad (1)$$
$$v' = y'.$$

Now, consider a point $(x', y', z')^T$ in space, expressed in Camera Coordinates, and consider the ray from the origin to that point. This ray pierces the focal cylinder at the point $(x', y', z')^T \cdot r/\sqrt{x'^2+z'^2}$. Combining this with (1) gives the basic equation for the panoramic camera optics

$$u' = r \arctan(x'/z') \quad (2)$$
$$v' = r\, y'/\sqrt{x'^2+z'^2}.$$

Here and in general, the notation arctan(x'/z') is meant to imply a four quadrant arctangent function such as the standard programming function atan2(x, z), and does not require a division by z'.

Now, writing $\theta = \arctan(x'/z') = u'/r$, we see that $x'/z' = \tan(\theta)$ and $v'/r = y' / \sqrt{x'^2+z'^2} = (y'/z') \cos(\theta)$. From this it follows that $(x', y', z')^T = z'. (\tan(\theta), v'/(r \cos(\theta)), 1)^T$, or

$$(x', y', z')^T = z' (\tan (u'/r), v'/(r \cos(u'/r)), 1)^T. \quad (3)$$

Relation of Image Coordinates to Focal Plane Coordinates. Focal Plane Coordinates and Image Coordinates are measured in different units (metres and pixels), and so their relative scale needs to be known. For the simple panoramic model, it will be assumed that the two coordinate systems are related by scaling and translation. In particular, we assume $k\, u = u'$ and $k(v - v_0) = v'$, where k is a scaling factor and v_0 is an offset. We do not need to consider an offset u_0, since it may be assumed that the line u=0 coincides with the origin of Focal Plane Coordinates. This is because the origin of Focal Plane Coordinates is the point where the Z' axis meets the focal cylinder, and the Z' axis is not defined unambiguously. In fact, we define the Z' axis by this condition.

Therefore, from (3) we obtain a relation between Camera Coordinates and Image Coordinates.

$$(x', y', z')^T = z' (\tan (u/k), (v-v_0)/(k \cos(u/k)), 1)^T \quad (4)$$

where the constant r has been subsumed into k.

The constant k and the offset v_0 will be called "internal parameters" of the camera. The Ground Coordinate to Camera Coordinate transformation defines (and is defined by) the "external parameters" of the camera.

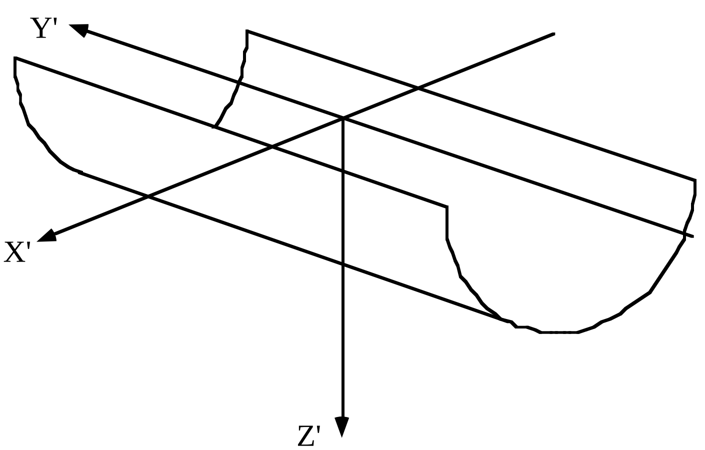

Fig 3. Camera Coordinates. For convenience, focal surface is shown in front of the camera

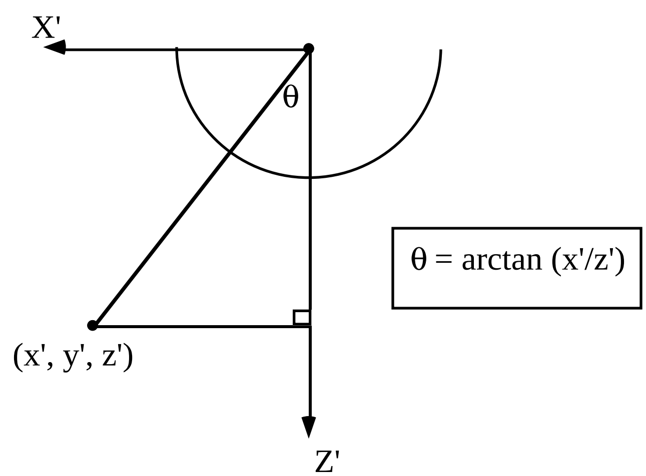

Fig 4. Imaging trigonometry.

4. DETERMINATION OF EXTERNAL CAMERA PARAMETERS.

We consider the problem of determining the external parameters of a basic panoramic camera given a number of Ground Coordinate to Image Coordinate correspondences. For the present, it will be assumed that the internal parameters of the camera are known.

Denoting the coordinates of the points in the unknown world coordinate frame by (x, y, z), we may write $(x', y', z')^T = M (x, y, z, 1)^T$ where M is a 3×4 matrix of the form (R | –Rt). From this it follows that

$$w\ (\tan(u/k),\ (v-v0)/(k\ \cos(u/k)),\ 1)^T = M\ (x, y, z, 1)^T \qquad (5)$$

where w is an unknown scale factor. The task is to find the matrix M given sufficient corresponding values of (u, v) and (x, y, z). If we write $u° = \tan(u/k)$ and $v° = (v-v_0)/(k\ \cos(u/k))$, then this equation becomes

$$w\ (u°, v°, 1)^T = M\ (x, y, z, 1)^T \qquad (6)$$

which is identical with the equation for a perspective camera. In other words, by the change of coordinates

$$u° = \tan(u/k) \quad ; \quad v° = (v-v_0)/(k\ \cos(u/k)) \qquad (7)$$

the problem of solving for the external parameters of a panoramic camera is reduced to the problem of solving for the external parameters of a perspective camera. There are several methods known in the literature for the solution of this problem. One method that provides good results is described by Sutherland[4]. In equation (6) the values of x, y, z, u° and v° are known, whereas w is unknown and the matrix M is to be found. If the rows of M are denoted $\mathbf{m}_1$, $\mathbf{m}_2$ and $\mathbf{m}_3$, and the vector $(x, y, z, 1)^T$ is denoted by $\mathbf{x}$ then (6) can be written as three equations

$$wu° = \mathbf{m}_1\mathbf{x},\ \ wv° = m_2\mathbf{x}, \text{ and } w = \mathbf{m}_3\ \mathbf{x}.$$

From this we may eliminate w, obtaining two equations

$$\mathbf{m}_3\ \mathbf{x}\ u° = \mathbf{m}_1\mathbf{x} \quad \text{and} \quad \mathbf{m}_3\mathbf{x}\ v° = \mathbf{m}_2\ \mathbf{x}.$$

These are linear equations in the entries of M. Given at least 6 control points, we may solve for the matrix M. If more than 6 points are given, then the least-squares solutions may be found by standard techniques[2].

In the presence of inexact data, the matrix M found by this technique will not necessarily be of the form (R | –Rt) with R a rotation matrix. However, the matrix of the desired form closest to M may be found as follows. If M' is the left-hand 3×3 submatrix of M, and $M' = U\ D'\ V^T$ is its Singular Value Decomposition, then M' is replaced by the matrix $M'' = U\ D''\ V^T$ where D'' is the diagonal matrix with equal diagonal entries equal to the average of the diagonal entries of D'.

4.1. Solution when all the ground-control points are coplanar.

If all the ground-control points $\mathbf{x}_i$ lie in a plane, then method of Sutherland for determining the external camera parameters does not work. Similarly, if the points lie close to a plane, then the method is numerically unstable. For the case of coplanar ground control points, a different method must be used.

Let $\mathbf{x}_i$ be a set of ground control points lying in or close to a plane, and let $\mathbf{u}°_i$ be the corresponding set of image coordinates. The problem is to find the best matrix M = (R | –Rt) such that $w_i\ \mathbf{u}°_i = M\ \mathbf{x}_i$.

As a first step, the points $\mathbf{x}_i$ are transformed to lie in or near the plane z = 0. This is done in two steps. First, points $\mathbf{x}_i$ are translated so that their centroid lies at the origin. Then, the unit normal vector $\mathbf{n}$ is found to minimize the sum $\sum(\mathbf{n}.\mathbf{x}_i)^2$. This is done using straight-forward linear techniques. Finally, the points and the normal vector are rotated so that $\mathbf{n}$ is parallel with the Z axis. At the conclusion of the algorithm, the matrix M can be corrected to take account of this transformation. Therefore, we shall assume that all the points $\mathbf{x}_i$ lie close to or on the plane z=0.

Now if we attempt to solve the equation $w_i\mathbf{u}°_i = M\mathbf{x}_i$ using the method of Sutherland to find the matrix M, then since the z-coordinate of each $\mathbf{x}_i$ is zero, it is clear that the third column of M may be arbitrarily chosen. Instead, we add the restriction that the third column of M is zero, and solve for the remaining 9 entries of M. That is, we find the best solution to (6) subject to the additional restrictions that $m_{11} = m_{21} = m_{31} = 0$. This determines the first two columns of the matrix M. It remains to find the correct third column of M so that the left hand 3×3 sub-matrix of M is a scalar multiple of a rotation matrix. This may be done as follows. Let M' be the left hand 3×3 sub-matrix of M, with third column zero. Let

$M' = U D' V^T$ be the Singular Value Decomposition of M'. Set $M'' = U D'' V^T$ where D'' is the scalar diagonal matrix having diagonal entries equal to the average of the diagonal entry of D'. Then M'' is the matrix closest to M' that is a scalar multiple of a rotation matrix.

Since the first two columns of M' may be changed by the above procedure, the fourth column of M should be recomputed using (6) and setting the entries of M' to the newly computed values. In the case where the points $\mathbf{x}_i$ do not lie precisely in a plane, it is also advantageous to negate the third column of M' and repeat the computation, selecting the one of the two solutions that gives the smaller error.

In practice, the points $\mathbf{x}_i$ will rarely lie exactly on a plane. Therefore, it is not clear which of the two methods used here should be applied to compute the matrix M. The strategy used, therefore, is to use both methods and select the solution that gives the smaller error. The methods described here have been used with good results for computing external parameters of both panoramic and perspective cameras.

4.2. Solution when internal camera parameters are unknown.

In the case where the internal parameters k and v_0 of the panoramic camera are unknown, the method described above may not be used directly. Nevertheless, in practice, it is still possible to find the placement of the panoramic camera as well as the internal camera parameters. The following method has been used with success.

If the internal camera calibration is known, then the method described above for determination of the camera placement is non-iterative and extremely rapid, involving only the solution of sets of linear equations. Therefore, it is feasible to repeat the computation for several choices of internal camera parameters and select the solution that gives the smallest error. This is the method that is used in the existing software implementation. In the present implementation, we iterate on values of k only, selecting v_0 to be the average of the v-coordinates of the control points, or the centre of the image. To estimate the range of k values over which to search, it is assumed that the extent of the image does not exceed 180°. This is true of the images taken by most panoramic cameras. The pixel width of the image is also known. Notice that k measures the number of pixels per radian, and hence 1/k is the number of radians per pixel. Therefore 1/k must lie in the range 0 to π/N, where N is the width of the image in pixels. The search is carried out in two steps. For each selected value of 1/k, the resection algorithm for a calibrated camera is carried out and the error is computed. First, a broad range search is carried out with several values of 1/k in the range. This locates the general location of the minimum error. The optimal value for 1/k is then refined by a Fibonacci search to find the minimum error. This method has been found to be effective and relatively rapid.

5. FULL PANORAMIC MODEL

The model of a panoramic camera described in the previous sections is only a simplified model of a more complex image acquisition process. Because of the differences between panoramic camera models, one approach[3] is to model individual cameras differently. In this paper, a general panoramic model will be used that applies to most different types of panoramic cameras.The full panoramic model will now be described.

5.1. Effect of Platform Motion.

The main simplification in the simple panoramic model so far considered relates to the fact that for general panoramic cameras the image is not acquired instantaneously, and the camera platform may be moving during the image acquisition. In fact, panoramic cameras are often mounted on aircraft that may be moving quite rapidly. The image is acquired by some mechanism equivalent to a rotating lens casting an image on a slit, so that one line of the image is acquired at a time, the line sweeping in the cross-axial direction (the u' direction in Focal Plane Coordinates. At any instant, only points lying in the plane containing the camera axis and the sweep line are imaged. This instantaneous plane will be called the sweep plane. The time required for acquisition of the image may be a few seconds. During this time the aircraft will move a considerable distance. In general, the axis of the camera will be mounted to be parallel with the body of the aircraft, and hence almost parallel with the direction of motion of the camera. However, because of possible cross winds and altitude changes in the aircraft during flight it may not be assumed that the direction of motion is perfectly parallel with the camera axis. Nevertheless, it is assumed in our model that the orientation of the aircraft is fixed during the image acquisition. A sensible alternative is to assume that the aircraft is undergoing uniform rotation.

The effect of the velocity of the aircraft is related to the rate of sweep during image capture. A faster rate of sweep is equivalent to a slower aircraft velocity. For this reason, the unit of time will be the time taken for the sweep plane to sweep through one radian in the cross-axial direction during image acquisition.

5.2. Forward Motion Compensation.

As mentioned, a panoramic image is obtained by focussing the image from a rotating lens on a slit so that just one line is imaged at a given time. Since the slit must be of finite non-zero width in order to obtain sufficient exposure, each point on the film will be exposed for a small, but significant time period. Meanwhile, the platform is moving forwards. If the image is not to be smeared, therefore, some correction is necessary in order that the image of a world point remain fixed with respect to the film during the exposure time. This correction is known as forward motion compensation. There are two common techniques for forward motion compensation (FMC).

If no compensation is made, then the forward motion of the aircraft will cause the image of any stationary point in space to move forwards with respect to the film. This image motion can be compensated for by moving the film forward in the axial direction of the camera during image acquisition. The speed at which the imaged point moves is dependent on the distance of the world point from the line of flight of the aircraft. If the terrain is relatively flat and the aircraft's altitude and velocity are known, then a rate of axial film motion may be chosen. In fact, the motion of the film should not be at a constant rate during the acquisition sweep, but should be sinusoidal, but this is not usually done.

A different method of forward motion compensation is to rotate the camera slowly about the X' axis during the acquisition sweep so that the Z' axis of the camera coordinate frame is always pointing to the same point on the ground. Once again, the rate of rotation depends on the altitude and velocity of the aircraft and the rate of sweep.

Although these methods of forward motion compensation serve to keep the image focussed on the film, they do introduce further distortion into the image.

5.3. Digitizing Parameters.

Whereas some panoramic cameras provide direct digital images, others capture their image on film. For computer analysis and correction, this image must be digitized. The transformation to image coordinates depends on several parameters. For instance, the film may be slightly rotated with respect to the digitizer, and hence with respect to the axes of the digitized image. Further, the placement of the origin of pixel coordinates with respect to the image (principal point offset) must be determined. Finally, the size of the pixels, and possible unequal scaling in two directions must be taken into account. In general, a full affine transformation of the image is possible during digitization. This affine transformation has 6 degrees of freedom, and parameters describing the transform may be chosen in various ways.

6. MATHEMATICAL FORMULATION OF THE PANORAMIC MODEL.

6.1. Notation:

Rotations: We use the following notation for rotation matrices:

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & c & -s \\ 0 & s & c \end{pmatrix} \quad , \quad R_y(\theta) = \begin{pmatrix} c & 0 & s \\ 0 & 1 & 0 \\ -s & 0 & c \end{pmatrix} \quad , \quad R_z(\theta) = \begin{pmatrix} c & -s & 0 \\ s & c & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

where c represents $\cos(\theta)$ and s represents $\sin(\theta)$. These are the rotation matrices which represent positive rotations about the X, Y, and Z axes respectively. Specifically, if $(x, y, z)^T$ are coordinates of a point in 3-space, then $(x', y', z')^T = R_x(\theta)\,(x, y, z)^T$ are the coordinates of the point rotated an angle θ about the X axis.

Translations.

If v is a 3-dimensional vector, then we denote the translation by the vector v by the symbol T(v). The result of applying the translation to a point with coordinates $(x, y, z)^T$ can be denoted by

$$(x', y', z')^T = T(v).\ (x, y, z)^T$$

As is well known, it is possible to represent a translation as a 4×4 matrix acting on homogeneous vectors. If rotations are represented as 4×4 matrices as well, then translations and rotations may be multiplied. We will choose to be somewhat careless about the dimension of matrices being multiplied together. In fact, we think of rotations and translations as abstract affine transformations rather than as matrices, and their combination in any order presents no problem.

6.2. Coordinates.

We distinguish five sets of axes and coordinates. The positions of the various axes to be described are dependent on a time parameter, t. We may arbitrarily scale time and fix the origin. We do this in a way such that the sweep rate of the sweep plane with respect to the camera frame is one radian in unit time and it passes through the Z' Camera Coordinate axis at time t=0.

Ground Coordinates: The first set of axes are the fixed Ground Coordinate axes (X, Y, Z) axes, defined as for the simple panoramic camera model.

Camera Coordinates: The second axes are the Camera Coordinate axes (X', Y', Z') attached to the camera, as in the simple model. In this case, however, the Camera Coordinate Axes are moving with respect to Ground Coordinates.

Sweep Coordinates: We introduce a third set of axes (X", Y", Z") called the Sweep Coordinate Axes. This frame rotates with respect to the (X', Y', Z') Coordinates and is aligned with these coordinates at time t=0. It is assumed to rotate in the positive direction about the Y' axis of the Camera Coordinates.

Focal Plane Coordinates: These are defined in the same way as for the simple model.

Image Coordinates: Finally, the coordinates of the image will be denoted by (u, v). The relation of Image Coordinates to the Focal Plane Coordinates is determined by the digitization parameters and is not assumed to be of the simple form assumed in the simple model. In fact, the transition from Focal Plane Coordinates to Image Coordinates will be an arbitrary 2-dimensional affine transform.

Our goal is to define the panoramic imaging mapping by tracing the location of a world point and its corresponding image point through the various coordinate frames from its original position in Ground Coordinates to its ultimate destination in Image Coordinates.

6.3. The Imaging Process:

Let M(t) be the coordinate transformation describing the transition from Ground Coordinates to Sweep Coordinates at time t. Thus, if $(x, y, z)^T$ are the coordinates of a ground point, then

$$(x'', y'', z'')^T = M(t) \, . \, (x, y, z)^T$$

are the coordinates of the point in the Sweep Coordinates at time t.

According to our model, the panoramic camera will image the point only at the time when it lies on the sweep plane. Now, $(x, y, z)^T$ will lie on the sweep plane exactly when x" = 0 in the above equation. Then, it will image the point at the point (t, y"/z") in Focal Plane Coordinates.

A further transformation will be applied by the transition from focal plane coordinates to image coordinates by the process of digitizing the image. This will be a two-dimensional affine transformation.

Our strategy for determining the image coordinates of a point $(x, y, z)^T$ in ground coordinates is therefore as follows.

- Determine the equation for the coordinate transition M(t) at time t. This can be done once for all the points imaged by the same camera.
- For the point (x, y, z) solve for that value of t such that $M(t).(x, y, z)^T = (0, y'', z'')^T$.
- The point will be imaged at the point (t, y"/z") in focal plane coordinates.
- Now compute the transform from focal plane to image coordinates to determine the image coordinates of the point.

These four tasks will be considered in the next four sections.

6.4. Determination of the Transition Matrix, M(t).

Transition from Ground Coordinates to Camera Coordinates : The transition from Ground Coordinates to Camera Coordinates depends on various parameters describing the camera location.

The location and orientation of the camera fixture at a time t are specified by a number of parameters.

Location of the camera at time t=0.

C_x^0, C_y^0, C_z^0 — the location of the camera at time t=0.

We denote by C^0 the vector $(C_x^0, C_y^0, C_z^0)^T$, relative to ground coordinates.

Orientation of the camera at time t=0 is specified by three angular rotations:

$\theta_z^0, \theta_y^0, \theta_x^0$ — the orientation of the camera at time t=0.

Velocity of the camera : We separate the camera velocity into horizontal and vertical components. The horizontal component of velocity is represented in polar coordinates by a direction angle relative to the Y axis of the camera, and a speed.

v_z is an angle representing the direction of motion of the camera in a horizontal direction. The angle v_z is measured relative to θz^0.

S represents the speed of the camera in the horizontal direction.

dC_z represents the vertical component of camera motion.

We constrain v_z to lie between $-\pi/2$ and $\pi/2$, and allow S to be positive or negative.

The velocity vector is therefore $(S.\cos(\theta_z{}^0 + v_z), S.\sin(\theta z^0 + v_z), dC_z)^T$, which will be denoted by dC.

Normally, if the aircraft is not "crabbing" (moving in a transverse direction), the velocity of the camera will be in the direction of the sweep axis, that is the Y axis. In this case, v_z, which represents the deviation of the sweep axis from the direction of camera motion will be zero. Similarly, the vertical component of camera velocity will be zero.

Rotation of the camera. The camera fixture is allowed to rotate during the sweep. This rotation is denoted by three quantities

$d\theta_z$, $d\theta_y$ and $d\theta_x$ — rates of change of the camera orientation.

For simplicity, $d\theta_z$ and $d\theta_y$ are assumed to be zero. However, $d\theta_x$ may vary in one style of camera due to Forward Motion Compensation. This method adjusts for forward motion of the camera by rotating the sweep axis of the camera backwards during the sweep.

In terms of these parameters, the transition from Ground Coordinates to Camera Coordinates at time t is given by $(x', y', z')^T = R(t) . C(t) \ (x, y, z)^T$ where C(t) is the translation $C(t) = T(-C^0 - t\ dC)$ and R(t) is the rotation $R(t) = R_x(-\theta_x - t\ d\theta_x) . R_y(\theta_y) . R_z(-\theta_z)$.

Transition from Camera Coordinates to Sweep Coordinates : The sweep plane rotates about the Y' axis of the Camera Coordinates at a constant rate of 1 radian per time unit. Hence, the Sweep Coordinates of a point are related to the Camera Coordinates by the equation

$$(x'', y'', z'') = R_y(-t)\ (x', y', z').$$

Now, starting with a point with ground coordinates $(x, y, z)^T$, by combining the two transformation we see that

$$(x'', y'', z'')^T = R_y(-t) . R_x(-\theta_x - t\ d\theta_x) . R_y(\theta_y) . R_z(-\theta_z) . C(t)\ (x, y, z)^T.$$

We want to put the transformation into a form in which all the time dependent terms are to the left and the time independent terms are to the right. We note that $C(t) = T(-C^0 - t\ dC)$ can be written in the form $T(-t\ dC) . T(-C^0)$. Accordingly,

$$\begin{aligned} M(t) &= R_y(-t) . R_x(-\theta_x - t\ d\theta_x) . R_y(-\theta_y) . R_z(-\theta_z) . C(t) \\ &= R_y(-t) . R_x(-t\ d\theta x) . R_x(-\theta_x) . R_y(-\theta_y) . R_z(-\theta_z) . C(t) \\ &= R_y(-t) . R_x(-t\ d\theta_x) . R^0 . C(t) \\ &= R_y(-t) . R_x(-t\ d\theta_x) . T(-t\ R^0\ dC) . R_0 . T(-C^0). \end{aligned}$$

This completes the first step of the planned strategy for computing the camera parameters (see above).

6.5. Solving for t.

The next step is, given a point with ground coordinates $(x, y, z)^T$ is to determine the time t at which it is imaged. This will be the time at which $M(t)\ (x, y, z)^T = (0, x'', z'')^T$. If the camera is undergoing crabbing motion, then there is (apparently) no closed-form formula for doing this. Instead, it is necessary to proceed by successive approximation. The method is as follows:

- Given the parameters described above, compute $R_0 . T(-C^0)$ and $R_0.dC$. These now apply to all points that this camera acts on.
- Now, given coordinates $(x, y, z)^T$, compute $(x^\circ, y^\circ, z^\circ)^T = R_0 . T(-C^0)\ (x, y, z)^T$.
- Now the problem can be cast as solving for t the following equation :

$$R_y(-t) . R_x(-t\ d\theta_x) . T(-t\ R^0\ dC)\ (x^\circ, y^\circ, z^\circ)^T = (0, y'', z'').$$

We can write solve this iteratively as follows: Write

$H(t) = R_x(-t\ d\theta_x)\ T(-t\ R_0\ dC)$.

We need to solve $R_y(-t)\ H(t)\ (x°, y°, z°)^T = (0, y'', z'')$. The algorithm is

$t = 0;$

do {

$(\alpha, \beta, \gamma)^T = H(t)\ (x°, y°, z°)^T$

$t = \arctan(\alpha/\gamma)$

} until convergence.

Note that the condition $t = \arctan(\alpha/\gamma)$ is exactly the condition for $R_y(-t).(\alpha, \beta, \gamma)^T = (0, y'', z'')$. Thus, this iterative solution starts with t=0, uses this to compute $(\alpha, \beta, \gamma)^T = H(t)\ (x°, y°, z°)^T$ then finds the new value of t such that $R_y(-t)\ (\alpha, \beta, \gamma)^T = (0, y'', z'')$, and so on. When the algorithm converges to a fixed value of t, we know that

$$(0, y'', z'')^T = R_y(-t)\ (\alpha, \beta, \gamma)^T = R_y(-t)\ H(t)\ (x°, y°, z°)^T$$

as required.

In carrying out this algorithm, note that the value of β is not needed. Suppose that $R_0.dC$is the vector (τ_x, τ_y, τ_y). Explicit computation gives the following algorithm:

$t = 0$

do {

$\alpha = x' - t\ \tau_x$

$\gamma = \cos(t\ d\theta_x).(z' - t\ \tau_z) - \sin(t\ d\theta_x).(y' - t\ \tau_y)$

$t = \arctan(\alpha/\gamma)$

} until convergence.

If the motion of the camera is directly along the direction of the Y' axis and the camera is not undergoing rotation, then the above algorithm will converge in one step. Otherwise, iteration is necessary.

6.6. Computing the Focal Plane Coordinates.

Once the value of t is known, we may compute $(0, y'', z'') = R_y(-t)\ H(t)\ (x°, y°, z°)^T$. The point $(x, y, z)^T$ is now imaged at the point $(t, y''/z'')$ in focal-plane coordinates.

6.7. Transition to Image Coordinates.

Once we have mapped the point onto the focal plane, there remains the question of determining the actual pixel coordinates of the image point. The transformation from Focal Plane Coordinates to Image Coordinates is in general an affine transformation. General affine transformations of the plane have six degrees of freedom, since they may be represented by a 2×3 matrix. One parametrization of the group of affine transformations is given by the values of the entries of this matrix. Instead, we give a set of more meaningful parameters below.

Forward Motion Compensation: One form of forward motion compensation is done by moving the film relative to the camera lens in a direction along the axis of the camera, that is, in a directional orthogonal to the sweep direction. This means that the (u', v') focal plane coordinates undergo an initial transformation of the form

$$(u', v') \rightarrow (u', v' + t\ dv)$$

where dv is a parameter measuring the rate of film, or equivalently lens motion.

Film Speed Correction: In certain types of panoramic cameras (rotating prism, optical bar) the film is moving. The speed of this motion should be regulated so as the create square pixels at the centre of the view. This is necessary to prevent blurring of the image, caused by motion of the image in the focal plane in the sweep direction. However, since it is a mechanical effect, it may be wise to model the camera to allow inaccuracies in this film speed. The effect of incorrect film speed will be to compress or expand the image in the sweep direction. This, therefore, gives a further correction:

$$(u', v') \rightarrow (u', v' + t\, dv) \rightarrow (u' . fsc , v' + t\, dv),$$

where fsc is a parameter measuring the film speed correction.

Rotation of the image: The difference between panoramic and perspective cameras is that with panoramic cameras, the two orthogonal directions in the image plane are not optically equivalent. This means that in order to distinguish them in a digitized image, the image should be aligned so that two Focal Plane Coordinate axes are aligned with the axes of the digitizer. The extent to which inaccuracies occur in this alignment must be corrected by a postulated rotation of the image. For perspective cameras, this rotation is equivalent to a camera rotation, and is taken care of by the camera orientation parameters. In the case of panoramic cameras, this is not so. Therefore, we need a further rotation parameter, called ρ.

Image Magnification: We assume that the magnification is the same in both directions of the image, except that we assumed above a stretching in the direction along the sweep axis (determined by the parameter dv described above). To be perfectly general, we could instead have assumed a stretching along some arbitrary direction, at the cost of introducing one other parameter which specified that direction. However, for the sake of simplicity, we do not do that. Therefore, we introduce a magnification factor, k to be applied to the whole image.

Coordinate Offset: Finally, we need to specify the location in image plane coordinates of the pixel origin, otherwise known as the principal point offset. In some cases, the principal point offset in the sweep direction will be redundant, particularly in the case where the camera orientation is constant. The fact that we are allowing camera rotations about the X' axis, but not about the axis Z' serves to determine the X' (and hence the Z') direction. This makes the concept of principal point offset in the u' direction meaningful. In any case, the u-offset is included as a separate parameter. As long as restraints are put on its value, in the form of a standard deviation of its value, this will not cause instability of the solution, even in the case where $d\theta_X = 0$.

Combined Internal Camera Parameters:

Taken all together, the internal parameters can be expressed in the form of a matrix. Denoting by $(u, v)^T$ the focal plane coordinates of the image point, the formulation is

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} k & 0 & p_u \\ 0 & k & p_y \end{pmatrix} . \begin{pmatrix} \cos(\rho) & -\sin(\rho) & 0 \\ \sin(\rho) & \cos(\rho) & 0 \\ 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} fsc & 0 & 0 \\ dv & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} . \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix}$$

$$= \begin{pmatrix} fsc.k.\cos(\rho) - k.dv.\sin(\rho) & -k.\sin(\rho) & -p_u \\ fsc.k.\sin(\rho) - k.dv.\cos(\rho) & k.\cos(\rho) & -p_v \end{pmatrix} . \begin{pmatrix} u' \\ v' \\ 1 \end{pmatrix}$$

It may be seen that this describes an arbitrary 2×3 matrix, representing an arbitrary affine transformation.

Of course, the internal camera parameters apply to every point that is imaged by the camera, and so the matrix shown just above will be computed just once, and then applied to every point.

7. SUMMARY OF CAMERA PARAMETERS FOR PANORAMIC CAMERAS.

We summarize here the camera parameters.

7.1. External Camera Parameters.

C_X^0, C_y^0, C_Z^0	the location of the camera at time t=0.
$\theta_Z^0, \theta_y^0, \theta_X^0$	Orientation of the camera at time t=0.
v_Z	direction of motion of the camera. in a horizontal direction relative to θ_Z^0.
S	speed of the camera in the horizontal direction.
dC_Z	vertical component of camera motion.
$d\theta_X$	rates of tilt of the camera, used as one form of Forward Motion Compensation.

7.2. Internal Camera Parameters.

k	magnification
fsc	film speed correction (scale adjustment in sweep direction)
ρ	angle of rotation of digitized image.
dv	film speed in axial direction for Forward Motion Control.
p_u , p_v	Principal point offsets. Pixel locations of the principal point.

7.3. Simplified Camera Models.

This makes a total of 16 parameters, and hence a minimum of 8 control points are necessary. However, we may normally assume that some of these parameters are zero, giving a simplified camera model. In particular

- If there is no crabbing, then we assume that v_z and dC_z are zero.
- We may assume fsc = 0 in most cases.
- Normally, one of $d\theta_x$ and dv will be zero, since these represent alternative methods of FMC.
- If the principal point is accurately known then we may assume that p_u and p_v are zero, or some known value.
- If $d\theta_x$ is zero, then $\theta_y{}^0$ is indistinguishable from internal camera corrections, and it can be assumed that $\theta_y{}^0 = 0$.

8. EXPERIMENTAL RESULTS

The above algorithm has been coded and tested on a number of synthetic and real examples. Despite the presence of an iterative loop in the computation of the camera model, convergence of the parameter estimation program was rapid, taking no more than about 30 seconds on a SPARK-2. This is largely because the initial parameter estimation using the simplified panoramic model gave a close initial estimate to the actual parameters.

Once the model has been parametrized, it may be used to correct the image by removing the panoramic distortion. As an example of this, Fig 5 shows a well known panoramic image of Manhattan (reproduced courtesy of ITEK corporation). Fig 6 shows the corrected image with panoramic distortion removed.

REFERENCES

1. William H. Press, Brian P. Flannery, Saul A. Teukolsky and WIlliam T. Vetterling, “Numerical Recipes in C— The Art of Scientific Computing”, Cambridge University Press, Cambridge, 1988.

2. “The Manual of Photogrammetry, 4th edition”, Published by the American Society of Photogrammetry, Falls Church, Va, 1980.

3. Michael Jackson, Clifford Greve, George Hoffman and David Ackerman, “A Parametrization of the Itek KA-80A Panoramic Camera”, Photogrammetric Engineering and Remote Sensing, Vol 48, No. 5, May 1982, pp. 761-769.

4. Ivan E. Sutherland, “Three-dimensional data input by tablet”, Proceedings of IEEE Vol 16, No 4, pp. 453-461, April 1974.

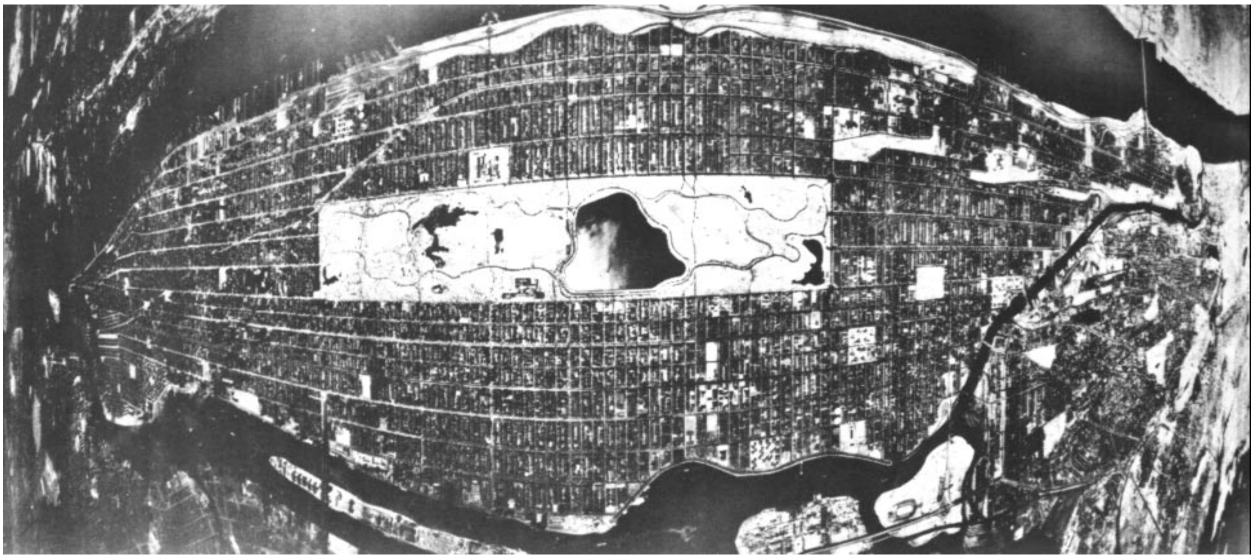

Fig.5. Panoramic Photograph of Manhattan, (courtesy of ITEK corporation).

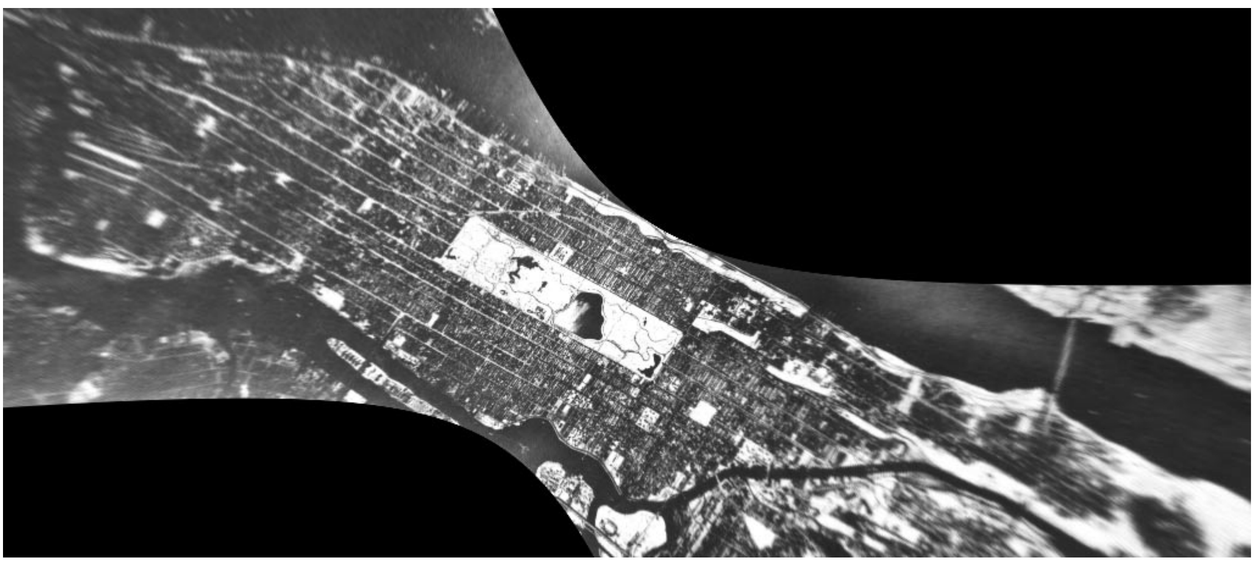

Fig.6. Corrected Manhattan Image.