Acquisition and registration of aerial video imagery of urban traffic

Rohan C. Loveland and Edward Rosten

Los Alamos National Laboratory, Los Alamos, NM, USA

ABSTRACT

The amount of information available about urban traffic from aerial video imagery is extremely high. Here we discuss the collection of such video imagery from a helicopter platform with a low-cost sensor, and the post-processing used to correct radial distortion in the data and register it. The radial distortion correction is accomplished using a Harris model. The registration is implemented in a two-step process, using a globally applied polyprojective correction model followed by a fine scale local displacement field adjustment. The resulting cleaned-up data is sufficiently well-registered to allow subsequent straight-forward vehicle tracking.

Keywords: tracking, video, traffic, urban, transportation

1. INTRODUCTION

The increasingly widespread availability of high-resolution video sensors, coupled with improvements in accompanying storage media, processors, etc... have made the collection of aerial video for the purposes of tracking automobiles and monitoring traffic much more feasible in recent years.[1] Corresponding efforts for the acquisition of aerial video imagery of traffic data using helicopters for the flight platform have been conducted by Angel and Hickman,[2] and Hoogendoorn.[3] More recently, a data collect has been conducted in a joint effort between Los Alamos National Laboratory (LANL) and the University of Arizona, an uncorrected frame which is shown below in Fig. 1.

Further author information: (Send correspondence to R.C.L.)
R.C.L.: E-mail: rohan@lanl.gov, Telephone: 1 505 667 4892

Figure 1. This shows an uncorrected frame of the original imagery taken from the helicopter hovering over Tucson. The radial distortion is clearly evidenced by the curved streets.

One of the distinguishing characteristics of this data collect is the focus on extremely low cost, primarily accomplished through the usage of a readily available COTS sensor, and the lack of specialized stabilization mounts or gimbals. In this paper, we demonstrate that useful data for analyzing traffic can be attained in this way, if the appropriate post-processing is conducted. In the following, we will briefly discuss the specifics of the data collect, and then present the post-processing algorithms applied to the data in order to make it usable, specifically addressing radial distortion correction and registration.

2. DATA COLLECTION

The motivation for taking the data was to be able to study the behavior of traffic in an urban setting. Accordingly, it was desirable to obtain imagery covering an area of approximately 1 km^2, with sufficiently high spatial and temporal resolution to allow reasonably reliable vehicle tracking. Prior work with data from other sources indicated that at least 5 minutes of video at a frame rate of $\approx 5Hz$ would be desirable, with a spatial resolution of $\approx 1ft./pixel$. The available budget for the data collect's total costs, including the helicopter time, sensor, and mount, was extremely low (i.e. $\leq$ \$5,000); this provided incentive for minimizing the data collect hardware cost and compensating for it with post-processing.

The data was collected in a six passenger Bell JetRanger helicopter, hovering above Tucson, Arizona, with one door off to allow for cables to be passed through. The sensor used was an IQEye 705 rgb camera with the V10 optics package. The sensor was rigidly mounted to the helicopter strut, with an ethernet interface to a laptop for data storage. A marine battery was carried along in the back seat as a power source.

A brief description of some of the data collect parameters is given in tbl. 1.

Table 1. Data Parameters.

Image Size	2560x1920 pixels
Sampling Rate	$\sim$ 5 frames/second
Spatial Resolution	$\sim$ 0.75 ft./pixel
Helicopter Height (AGL)	$\sim$ 3000 ft.
Field of View	$\sim 70^\circ$

3. RADIAL DISTORTION CORRECTION

As is apparent from Fig. 1, the radial distortion present in the raw imagery, seen in the curvature of the grid of streets, is significant enough to pose severe problems if left uncorrected.

A general equation for distortion correction is given by

$$I_1(\mathbf{x}) = I_0(\mathbf{x}'), \quad \mathbf{x}' \equiv \mathbf{D}(\mathbf{x}), \tag{1}$$

where I_1 is the output (corrected) image, I_0 is the input (distorted) image, and $\mathbf{x}'$ is a pixel location in the input image that is mapped to location $\mathbf{x}$ in the output image by distortion correction function $\mathbf{D}(\mathbf{x}) : \mathbb{R}^2 \Rightarrow \mathbb{R}^2$.

If the distortion is assumed to be isotropic, or strictly radial, then $\mathbf{D}(\mathbf{x})$ can be written as

$$\mathbf{D}(\mathbf{x}) = f(\rho)\hat{\mathbf{r}} + \mathbf{c}, \tag{2}$$

where $\mathbf{c}$ is the center pixel coordinate, the normalized radius vector $\hat{\mathbf{r}} = (\mathbf{x} - \mathbf{c})/\rho$, $\rho = \|\mathbf{r}\|$, and f is a scalar function of ρ.

Several models for distortion correction have been used previously, including the polynomial model,[4,5] where

$$f(\rho) = \rho(1 + k_1\rho^2 + k_2\rho^4 + k_3\rho^6 + ...), \tag{3}$$

Figure 2. This shows a radially corrected version of a frame; lines have been superimposed to allow comparisons for straightness.

the Harris model,[6] where

$$f(\rho) = \frac{\rho}{\sqrt{1+\gamma\rho^2}}, \tag{4}$$

and more recently the rational function model, where polynomial and perspective transforms are combined into a rational polynomial.[7]

All of these models can have adequate performance depending on the characteristics of the camera optics. In this case we chose the Harris model for its reasonable performance, single parameter, and ease of inversion (achieved by simply negating γ).

A variety of methods exist for estimating camera distortion correction model parameters, ranging from the simple to the complex, and depending on the content of the available imagery from the sensor to be calibrated. In this case, however, based on prior knowledge of the content of the imagery, where the streets are known to generally be straight, it was possible to simply apply a range of γ's to a single image and manually select the value corresponding to the image with the straightest streets. This was done and resulted in a value of $\gamma = 0.7$. A radially corrected resulting image is shown in Fig. 2.

4. REGISTRATION

4.1 Motivation

It is evident when viewing the imagery that a significant amount of motion occurred during the data collection, present in both the translation and orientation components of the helicopter pose. The translation had less of an overall effect because the magnitude* of it was not large with respect to the distance from the sensor to the scene, which was in this case $\sim$ 3000 ft. The angular deviations, however, were significant and resulted in large amounts of both high and low frequency jitter.

It is often the case with video sequences that it is necessary to transform it into a single stationary coordinate system, in order to make the data more usable. This *registration* process produces imagery where the location of a given stationary object in the scene also remains stationary in the transformed imagery sequence. This is particularly important as a precursor to tracking, which is largely based on assumptions of motion smoothness and continuity; any motion remaining in the sequence after registration can be very difficult to separate from the motion of objects in the scene relative to one another.

*The pilot estimated the overall helicopter translation to be within a few hundred feet.

4.2 Registration Algorithm Overview

An extremely large variety of approaches to registration are possible.[8,9] The approach taken to registration, and particularly the correction model, will generally be problem specific, depending on the type of data and distortion. In this case the distortion remaining following the radial distortion correction is due to:

- residual optical artifacts remaining after radial distortion,
- perspective projection,
- the interaction of the rolling shutter of the sensor combined with the helicopter motion, with the latter resulting from the particularly complex system of pilot control, servo feedback adjustment, and turbulence effects,
- the assumption that Tucson is planar.

In the light of these complex sources of distortion, a fairly flexible model is necessary. As is the case with all flexible models, however, care must be taken in order to ensure that the solution ultimately arrived at is the correct one, and not simply the equivalent of the closest local minimum to the starting point. For this reason a combination of transformations were used, starting with those which are fairly inflexible, and proceeding towards models allowing greater flexibility as the true solution is homed in on. In this case a combination of global transformations, enforcing some smoothness, were followed by a local transformation, which allowed maximum flexibility.

In this implementation, for both the global and local approaches, only a pair of frames is considered at a time. These are the input frame, and the base frame to which it is to be registered. The approach uses control points, rather than all the points in the image. For the transform selected at the particular stage of the registration, optimal parameters are arrived at using an optimization process, over an objective function which is based on the differences between the locations of the base control points and their transformed counterparts from the input image. The control points are filtered in this process to eliminate the inevitable false matches.

The choices of base and input frames, transforms, types of control point detectors and descriptors, and optimization and filtering methods are dependent on the stage of registration. Here we choose to initially register everything to an underlying orthorectified base image of the area acquired from Google Maps™. Because the base frame was taken with a different sensor, the registration of it to the helicopter imagery (i.e. *inter*sensor registration) was more difficult than registering different frames taken with the same sensor (i.e. *intra*sensor registration). Reasonable intersensor performance was in fact only possible with frames having orientations approximately the same as in the Google Maps image. As a result, the first registration step taken was an intersensor registration done between a single frame of the helicopter imagery and the Google Maps base image. This allowed matching control points to be found in both images. A *global* transform (i.e. a transform acting on the entire image) was used in this step.

This was followed with a second registration step, which was intrasensor registration, done between all the other frames of helicopter imagery and the single previously transformed helicopter base frame. The transformation used in this case was also a global transform.

The nature of the artifacts in the helicopter imagery was such that performance could be improved following this with a further *local* transform (i.e. a transform acting on separate parts of the image). This led to a final registration step, based on a local displacement field correction.

This process is illustrated in the diagram shown in Fig. 3.

A more detailed discussion of this process is given in the following sections. Several tools are used repeatedly, the full description of which would exceed the scope of this paper. Some of the more well-known of these will therefore be used in the following with no further explanation, but for more detail on:

- Perspective projection, the transform used to describe perspective, see Zisserman,[10]
- SIFT, used for detecting and describing control points, see Lowe,[11]

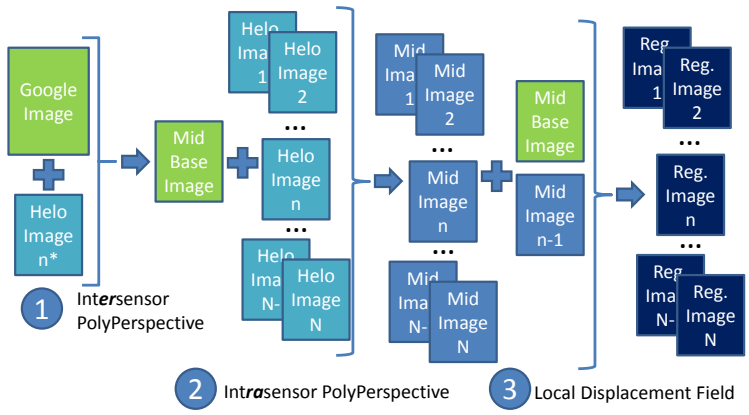

Figure 3. Diagram of registration transformations. The first is an intersensor global polyperspective, followed by an intrasensor global polyperspective, followed by a local displacement field correction.

- RANSAC, used to find models uninfluenced by outliers, see Fischler and Bolles,[12]
- REQCON, also used to find models uninfluenced by outliers, see Loveland,[13]
- Nelder-Mead Optimization, a multi-dimensional gradient-free unconstrained optimization method, see Nelder and Press.[14,15]

4.3 Intersensor Registration

The intersensor registration is done between a single input frame taken from the beginning of the helicopter sequence and the Google^{TM}base image. Due to the number of differences in the two images resulting from the usage of different sensors, the process was broken into two steps, consisting of a projective transform executed at a coarse scale, which provided the starting point for the optimal parameter search for a polyprojective transform executed at the full image scale.

4.3.1 Coarse Intersensor Registration

Initial robustness at the coarse scale was achieved by using SIFT features at the sub-sampled coarse scale, which ensured that:

- the scale of the features (e.g. buildings) was large enough to promote distinctiveness,
- the overall number of features was low enough.

The first consideration is important because it's initially necessary to have features that are distinct over the entire image. The second consideration is generally important with algorithms which ensure discriminance by requiring that the best match be significantly better than the second best match; this fails when there are so many features that the probability of a spurious, but seemingly good, secondary match becomes high.

The specific steps of the coarse transform registration were:

1. convert both the base and input images to gray-scale,
2. sub-sample both images to $\frac{1}{4}$ scale,
3. manually identify a single corresponding point in both images,
4. calculate SIFT keypoints in both images,
5. find matches between the keypoints,

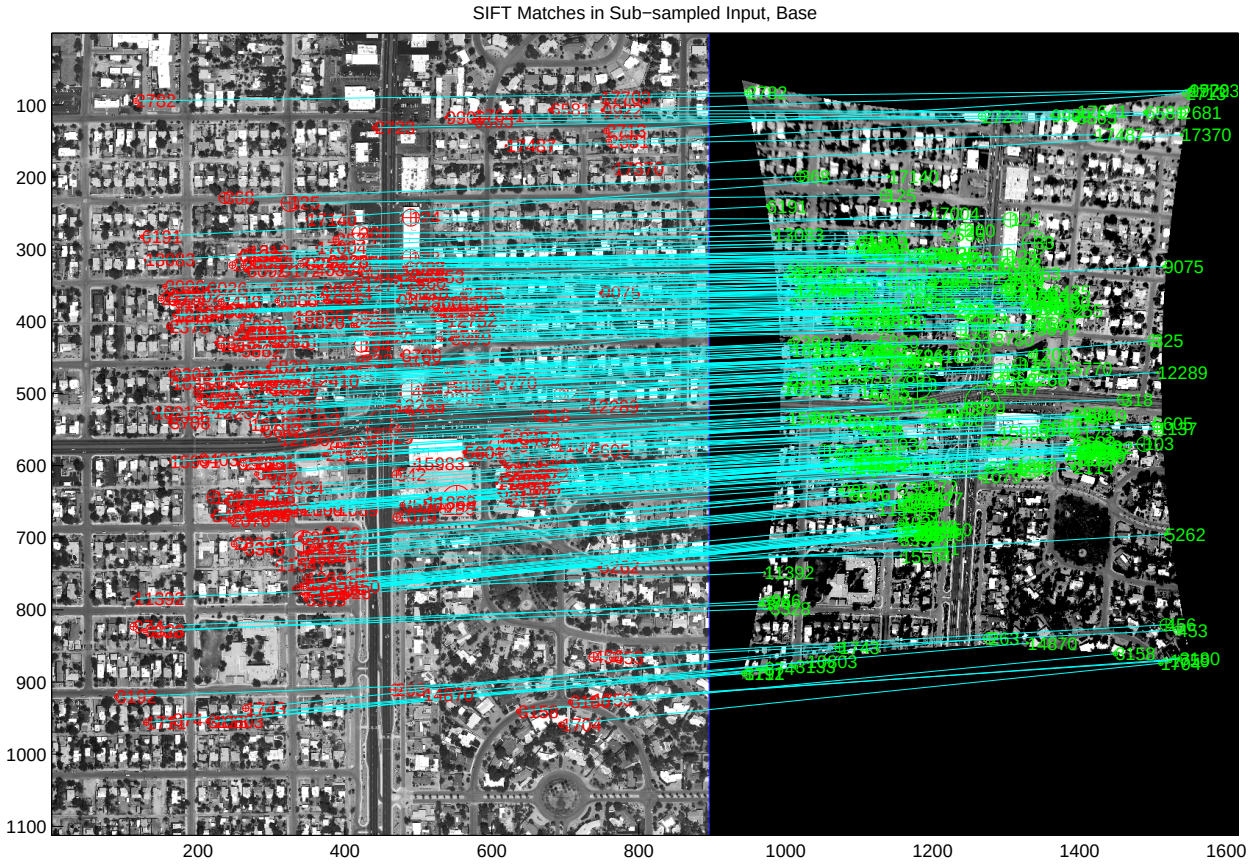

Figure 4. SIFT matches between sub-sampled input and base images.

6. use RANSAC with Nelder-Mead optimization to find the best projective transform.

The selection of $\frac{1}{4}$ scale was motivated by selecting as small an image as possible while preserving enough image detail. The single manually identified corresponding point allowed location based filtering of the SIFT features, which helped to significantly reduce the numbers of false matches. The SIFT keypoints with corresponding matches are shown in Fig. 4. At this sub-sampled scale, the base image had 17,974 SIFT keypoints, while the input image had 8,807. Of these, 324 matches were found. The majority of the matches appear to be correct, as evidenced by similar connecting line directions between the left and right sides, but careful scrutiny shows some obviously false matches.

RANSAC was used here to find the best projective model in the presence of these false matches. Four points were used to calculate each model, with an inlier threshold of five pixels and a maximum number of 1000 iterations. The resulting projective transform corrects both the orientation difference of $\approx 10°$ and the scale difference, and provides a good starting value for the polyprojective transform parameter optimization of the fine transform.

4.3.2 Fine Intersensor Registration

As mentioned previously, the expected complex nature of the distortion necessitated a more flexible model than just the projective. A model which combines the flexibility of the polynomial transform with the perspective

projection is the 'polyprojective', referred to in Claus[7] as the 'rational polynomial'. This can be expressed as

$$\begin{bmatrix} u_2 \\ v_2 \end{bmatrix} = \psi \left(\begin{bmatrix} a_j & a_{j-1} & ... & a_k & a_{k-1} & ... & a_2 & a_1 & a_0 \\ b_j & b_{j-1} & ... & b_k & b_{k-1} & ... & b_2 & b_1 & b_0 \\ c_j & c_{j-1} & ... & c_k & c_{k-1} & ... & c_2 & c_1 & 1 \end{bmatrix} \begin{bmatrix} u_1^n \\ u_1^{n-1} v_1 \\ ... \\ v_1^n \\ u_1 v_1^{n-1} \\ ... \\ u_1 \\ v_1 \\ 1 \end{bmatrix} \right). \quad (5)$$

where u_1 and v_1 are the input coordinates, u_2 and v_2 are the output coordinates, and $\psi(.)$ is the function used to "de-homogenize" the output by dividing the 1^{st} two components resulting from the matrix multiplication by the 3^{rd}.

It can be seen that this reduces to the projective transform if all the a_m, b_m and c_m are set to 0, where $m > 2$. Likewise, this reduces to a polynomial transform if the $c_j = 0, m \in [1, j]$. Note that the usage of homogeneous coordinates has once again allowed us to eliminate 1 degree of freedom by setting the lowest right-hand element of the transform matrix equal to 1.

The subscripts 'n' and 'j' have been used above; n is the highest single variable degree, which makes $j = (n+1)(n+2)/2$. The total required number of coefficients is then $3 * ((n+1)(n+2)/2) - 1$. So for a 2^{nd} degree polynomial, we are already up to 17 coefficients, while a 3^{rd} degree polynomial would require 29. For this dataset the 2^{nd} degree was used.

The 'fine' registration used the the polyprojective transform to account for the extra distortions beyond perspective which are discussed above. SIFT keypoints didn't work well at this scale, because of the relatively large size of distinct objects (e.g cars, buildings) relative to individual pixels. For this reason straightforward pixel templates were used instead here. A grid of templates on 50 pixel center displacements, with a template size of 75×75 pixels, was extracted throughout the input image, and then an appropriately rescaled version of the coarse transform was used to find corresponding search starting locations in the base image. Dithered normalized cross-correlation over a 20×20 neighborhood was then used to refine these starting location estimates to the final matching locations. This direct correlation took place in spite of the $\approx 10°$ orientation difference and the $\approx 20\%$ scale difference, so a relatively large number of mismatches were to be expected.

These grid center matches were then used with REQCON, to find the best polyperspective transform model through successive optimizations with mismatch removal. Each iteration removed the worst 5% of the points until the average error of the remaining transformed points was less than 2 pixels.

The specific steps of the fine registration transform were:

1. extract a grid of pixel templates from the input image,
2. rescale the projective transform up to a full-scale polyperspective transform
3. use the rescaled transform to find corresponding search starting locations in the base image,
4. use DNCC to find actual matching locations in the base image,
5. use REQCON with Nelder-Mead optimization to find the best polyperspective transform.

This process produced the image referred to in Fig. 3 as the "Mid Base Image", which was then used as the base image for the intrasensor registration of all of the following helicopter imagery.

4.4 Intrasensor Registration

The intrasensor registration process is inherently more forgiving than the intersensor process. This in turn allows the usage of different subalgorithms which can provide much denser coverage of the image with better matching accuracies than were previously available. The registration processes are still similar, however, in that a coarse to fine approach is still used, as is the basic framework of finding an optimal set of transform parameters from an iteratively refined set of control point matches.

In the intrasensor case, the single registered frame arrived at previously using the intersensor process is used as the base frame. An important difference between this frame and the Google^{TM} frame, however, is that drift in the helicopter imagery can eventually result in only a small area of intersection between the areas covered in a particular helicopter frame and this base frame.[†] As a result, this base frame was used in conjunction with the registered frame immediately preceding the one currently being registered. Since changes in coverage area are relatively slow in the helicopter imagery, this joint base frame approach ensures that a large number of control points will still be available in the overlapping area.

The fine (i.e. full-scale) registration transform in this case was also arrived at using the coarse registration transform as a starting point. Where SIFT keypoints hadn't worked well previously, here they could be found in sufficient numbers over the whole image. Location gating was still necessary, however, due to the extremely large number of keypoints that are present in the full scale image. Another grid was accordingly used, where SIFT keypoints were found in 50×50 templates distributed over the image on 100 pixel center displacements. In this case, the keypoints were only allowed to be matched to their opposite numbers in the appropriate corresponding template. The numbers of keypoints arrived at in this case were sufficiently large so that the combined base and previous image list was decimated by a factor of 3.

At this point, as in the intersensor registration process, the lists of SIFT keypoints from the input image was then used in conjunction with the list from the combined base and previous images to find the best polyperspective transform model through successive optimizations with mismatch removal using REQCON.

The specific steps of the fine registration transform were:

1. create a grid of centerpoints in the input image,
2. extract the SIFT keypoints from the immediate neighborhood of each centerpoint in the input image,
3. rescale the projective transform up to a full-scale polyperspective transform
4. use the rescaled transform to find corresponding grid centerpoints in the base/previous images,
5. extract the SIFT keypoints from the immediate neighborhood of each centerpoint in the base/previous images,
6. match the keypoints from each grid centerpoint,
7. use REQCON with Nelder-Mead optimization to find the best polyperspective transform.

5. DISPLACEMENT FIELD LOCAL REGISTRATION REFINEMENT

As mentioned previously, a multi-stage process, with increasing transform flexibility, is often used for registration. Here the final stage of the registration is based on local displacement corrections, rather than the global transforms used previously. No smoothing of the local displacement vectors is done, which allows maximum response to local variation. The input and base images in this case are both helicopter images, resulting from the intra and inter sensor registration done previously.

The local displacement field is calculated by once again breaking the image into a grid, in this case of 200×200 neighborhoods with centers 200 pixels apart. Since the input and base images were from the same sensor, SIFT

[†]Recall that the Google^{TM} frame completely encompasses the helicopter imagery area, while the registered frames cover only a part of that window.

keypoints could be found and matched for each corresponding neighborhood. From these matches, REQCON was used to arrive at a single consensus based displacement vector for that grid location. This field of displacement vectors was then interpolated down to the level of individual pixels. The horizontal and vertical components of this interpolated field for a particular image are shown in Fig: 5.

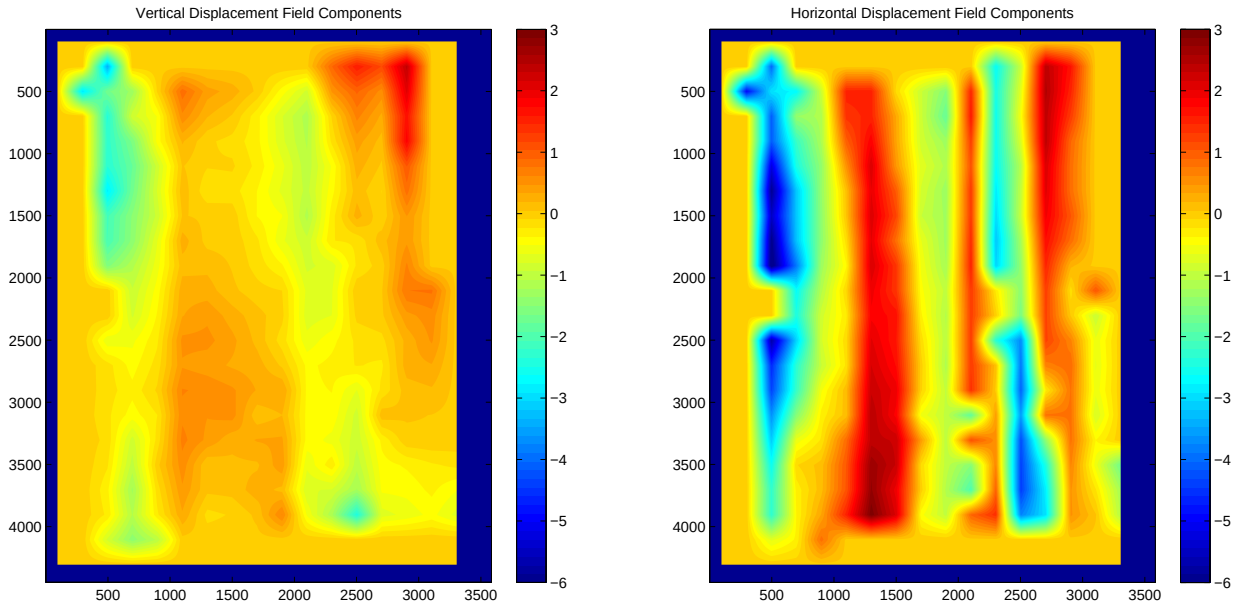

Figure 5. Interpolated vector components of displacement field.

The specific steps of the displacement registration were:

1. create a grid of centerpoints in both the base and input images,
2. extract the SIFT keypoints from the immediate neighborhood of each centerpoint in both images,
3. match the keypoints in each neighborhood,
4. use REQCON to arrive at a single consensus-based displacement vector for each neighborhood,
5. interpolate the displacement field up to the full image resolution,
6. create a registered output image by interpolating at each pixel plus displacement vector in the input image,

Overall, the choice of the neighborhood size was relatively important, in that it had to be significantly larger than the road widths. This was made evident by initial experimentation which showed that too small a neighborhood size resulted in one displacement vector resulting entirely from moving cars, which all had very similar speeds. The choice of grid center offset was driven by a trade-off between needing fine enough spacing to capture the spatial frequencies of the remaining distortion, and maximizing computational efficiency.

6. RESULTS

The result of the intrasensor registration for a pair of frames separated by 9 other frames in the sequence is shown in Fig. 6. The image shows the two frames semi-transparently superimposed on each other. Careful examination of the borders shows that in most cases the error is around a pixel or less.

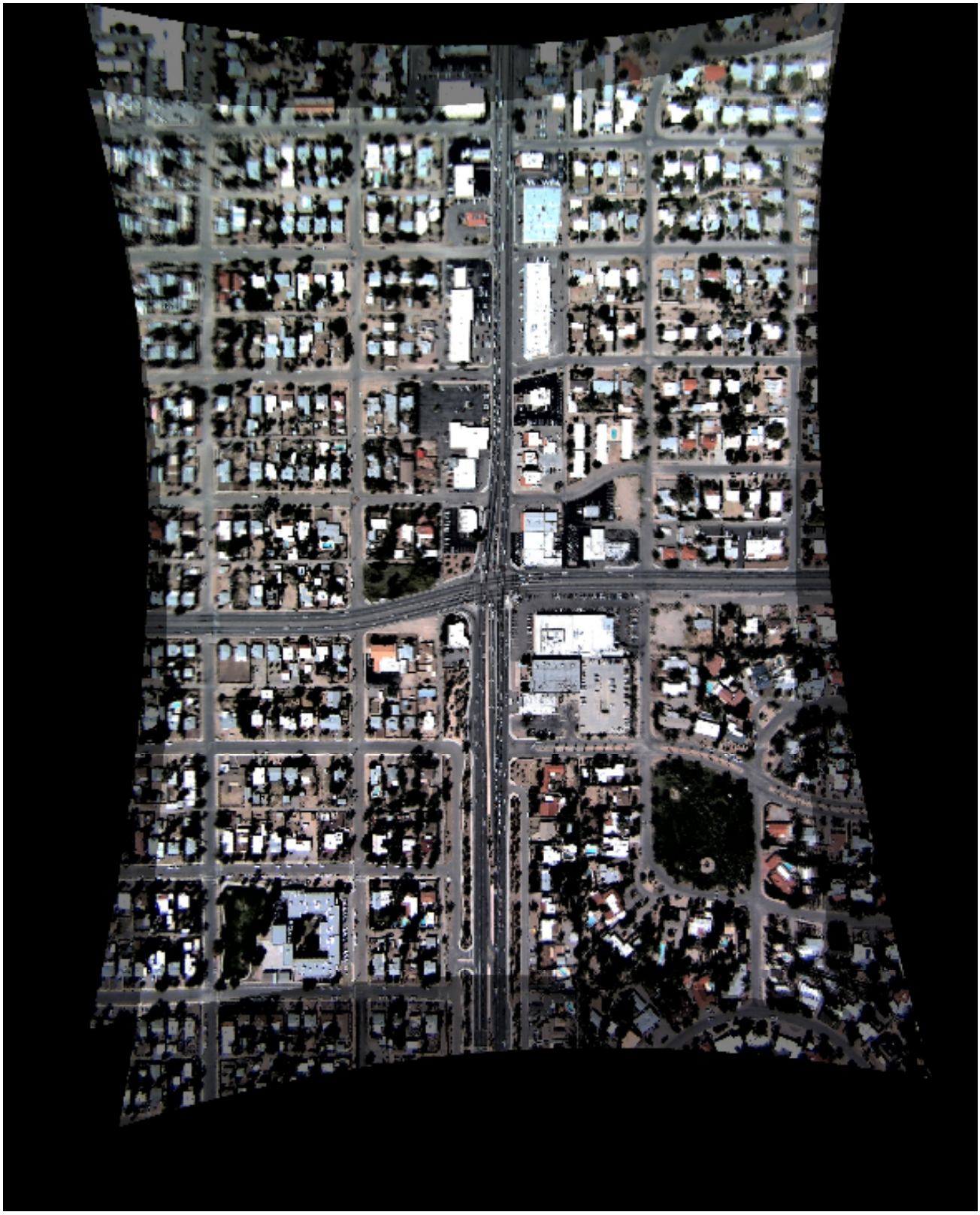

Figure 6. Intrasensor registered input frame 10 transparently superimposed onto base frame 1.

This can be quantitavely evaluated by determining the amount of movement remaining following the registration. In order to do this, a fixed set of 5 points were visually located in 10 frames spread evenly over the sequence. The points were deliberately chosen to be: a) on the ground, b) easily identifiable, and c) at the corners, and center, of the image. This was done for the initially unregistered (but radially corrected) input sequence, the globally registered sequence following the intrasensor transformation, and the local displacement field intrasensor sequence. The results are shown in Table 2.

Table 2. Registration results for 5 locations, using manually identified points, for the initial imagery and following the global and local registration.

Stage	Location											
	NW		NE		Center		SW		SE		Average	
	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
Unregistered	86.7	46.0	96.0	60.0	69.0	49.2	125.9	50.2	88.8	46.2	93.3	50.3
Post-Global	3.0	2.0	2.6	2.0	2.5	1.2	2.2	1.0	2.4	2.3	2.6	1.7
Post-Local	1.7	1.0	1.0	0.7	0.9	0.6	1.0	0.7	0.8	0.3	1.1	0.6

The table shows the mean and standard deviations of the points' motion for each of the selected corners and center, with the final column on the right showing the average of these over the locations. It's clear that each of the stages improves the results, with a final best result of registration within about 1 pixel, most of the time.

The most dramatic differences are obviously seen between the initial unregistered results and the global registration. The local registration has significantly improved the results as well, however, providing a better than 50% reduction in the final result, from around 2.5 pixels down close to 1. This difference is significant given the small number of pixels available on the targets, and in the size of the lanes in the roads.

Another way to gauge the effectiveness of the registration is to watch a movie of the registered imagery. Doing so shows that some artifacts remain, and the general effect is unsettling because of the rapid (though small) displacements of roads, buildings, etc... The movie suggests that although the table is generally correct, there are occasional sudden excursions that are larger than would be anticipated. Overall, however, the level of registration should be sufficient for tracking.

7. CONCLUSION

In this paper we have shown that usable aerial data of urban traffic imagery can be obtained with low-cost components from an unstabilized helicopter platform, given sufficient post-processing, which in this case included radial distortion correction and a coarse-to-fine registration algorithm. The algorithms used, including a polyprojective transform and local displacement field correction, demonstrated sufficient flexibility to compensate for the interaction of the sensor's rolling shutter combined with the helicopter motion, providing registration to an orthorectified base image with error magnitudes of approximately one pixel.

ACKNOWLEDGMENTS

The cooperation and encouragement of Drs. Mark Hickman and Pitu Mirchandani of the University of Arizona is very gratefully acknowledged. Thanks also to Dr. David Lowe of the University of British Columbia for making his SIFT code available online. Partial funding for the data collection and reduction for this effort has been supplied by Los Alamos National Laboratory, the National Nuclear Security Administration, and the U.S. Department of Energy.

REFERENCES

1. R. Kumar, H. Sawhney, S. Samarasekera, S. Hsu, H. Tao, Y. Guo, K. Hanna, A. Pope, R. Wildes, D. Hirvonen, M. Hansen, and P. Burt, "Aerial video surveillance and exploitation,," *Proceedings of the IEEE* **89 no. 10**, pp. 1518–1539.

2. A. Angel, M. Hickman, P. Mirchandani, and D. Chandnani, "Methods of traffic data collection, using aerial video,," *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems* , pp. 31–36, 2002.
3. S. Hoogendorn, H. Van Zuylen, M. Schreuder, B. Gorte, and G. Vosselman, "Microscopic traffic data collection by remote sensing,," *Transportation Research Board Annual Meeting CD-ROM* , 2003.
4. J. Heikkilä, "Geometric camera calibration using circular control points," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **22**, pp. 1066–1077, Oct. 2000.
5. H. Beyer, "Accurate calibration of ccd-cameras," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* , pp. 96–101, June 1992.
6. B. Tordoff and D. W. Murray, "The impact of radial distortion on the self-calibration of rotating cameras," *Computer Vision and Image Understanding* **96**, pp. 17–34, Oct. 2004.
7. D. Claus and A. Fitzgibbon, "A rational function lens distortion model for general cameras," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **1**, pp. 213–219, 2005.
8. L. G. Brown, "A survey of image registration techniques," *ACM Comput. Surv.* **24**(4), pp. 325–376, 1992.
9. B. Zitova and J. Flusser, "Image registration methods: a survey," *Image and Vision Computing* **21**, pp. 977–1000, October 2003.
10. A. Zisserman, "Single view and two-view geometry," *Handout, EPSRC Summer School on Computer Vision* , 1998.
11. D. Lowe, "Distinctive image features from scale-invariant keypoints," in *International Journal of Computer Vision*, **20**, pp. 91–110, 2003.
12. M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM* **24**(6), pp. 381–395, 1981.
13. R. Loveland, *Interactive Detection and Classification of Anomalies in Urban Traffic Imagery.* PhD thesis, Oxford University, 2008 - in preparation.
14. J. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal* **7**, pp. 308–313, 1964.
15. W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C*, Cambridge University Press, Cambridge, UK, 1999.