AIRBORNE VIDEO REGISTRATION FOR VISUALIZATION AND PARAMETER ESTIMATION OF TRAFFIC FLOWS

Anand Shastry
Robert Schowengerdt
Electrical and Computer Engineering Department
The University of Arizona
Tucson, AZ 85721
anandcs@ece.arizona.edu
schowengerdt@ece.arizona.edu

ABSTRACT

Techniques for incorporating airborne video into a multimedia GIS for traffic flow visualization are described. The base layer of the GIS consists of mosaiced aerial orthographic images of Tucson. Digitized non-roadway segments serve as a mask to emphasize roadways. A helicopter with a sophisticated on-board attitude measurement system was used to collect video and still images of traffic flow. The video was digitised, and a software feature tracker was used to track fixed features between successive frames of a video sequence and make frame-to-frame correspondences to obtain control points for registration. Each frame is then registered to the previous frame using a polynomial distortion model to obtain a modified video sequence. Most of the helicopter motion is thus removed and the modified video sequence appears to be taken from a stationary platform. The feature tracker also tracks cars through the sequence, enabling velocity estimation in an automated fashion. Thus, the entire scheme converts spatio-temporal data into temporal-only data, thereby making vehicle velocity estimation possible.

INTRODUCTION

Multimedia and GIS

Approaches to integrate multimedia with a GIS (Geographical Information System) fall under the following categories:

- *Multimedia in GIS*: Being the favoured approach by GIS analysts, this brings multimedia capabilities to existing GIS packages such as ArcView, MapInfo, GRASS, etc.
- *GIS in multimedia*: The emphasis is on seamless integration of multimedia data types using proprietary multimedia authoring tools such as Macromedia Director, Icon Authorware, Toolbook, etc (Panagopoulou, G., 1994; Fonesca, A., 1994; Schneider, B., 1999).
- *Web-based GIS*: An increasingly popular approach makes use of the ability of the worldwide web to integrate software such as CGI (Common Gateway Interface), Java applets and internet map servers (ArcIMS, MapobjectsIMS) (Soomro, T, 1999).

We have used the first approach here because of ease of implementation. The registered and corrected video is included in the GIS at the relevant locations.

Video for traffic parameter estimation

Traffic management systems rely on sensors to estimate traffic parameters. The currently predominant technology uses magnetic loop detectors buried underneath roads to count vehicles. Video is a promising tool for monitoring traffic since additional parameters such as vehicle classifications, travel time, queue lengths, lane changes, relative velocity, closing distances, vehicle trajectory, etc can be determined. Video camera installation is superior to loop detectors in that it does not require digging up of road/pavements and can be upgraded or removed relatively easily.

Video cameras are usually mounted on traffic lights at junctions, bridges and on tall poles or structures constructed especially for the purpose. These permanent or semi-permanent cameras have a communication link by which the video is transmitted to control stations or research centers.

Most stationary platform-mounted commercial systems suffer from occlusion of vehicles due to moving shadows and other vehicles and fail due to changing light conditions, traffic congestion, and camera motion due to wind.

We explore video cameras attached to an airborne helicopter as a means for overcoming the drawbacks of stationary platform mounted cameras. These flights are conducted on a data collection-need basis and typically last not more than a few hours. The problem of occlusion and shadowing of vehicles is eliminated in vertical (nadir) viewing. It now becomes possible to follow platoon movement on city roads and freeways, where a platoon is defined as a group of vehicles driving in close proximity. Following and monitoring traffic flow at any given location is a capability also afforded by airborne remote sensing.

Video Registration

Registration is the process of establishing correspondences between images, so that the images are in a common reference frame. Knowing the co-ordinate locations of points corresponding to the same physical object in both images, an affine or higher order transformation can be computed to warp the images to the reference. Video adds new dimensions to the registration problem, since there are a very large number of frames even in a short video sequence, depending on the frame rate. For aerial monitoring of ground sites, it is important that the video be related to the 3-D real world co-ordinates inherent in a traditional map or orthographic image.

One approach to video registration relies on recovery of elevation information from the images in the form of a DEM (Digital Elevation Map) and subsequently matching it to a reference elevation map stored in the database (Park, R.H., 2002). This technique avoids the problem of changing lighting conditions due to different viewpoints and changes in cloud or weather conditions. However, it relies very heavily on a-priori information and its accuracy. Conventional approaches make use of metadata or telemetry data, which contain time-sequenced real world co-ordinates, along with the rotational angles of roll, pitch and yaw of the airborne camera, all of which are synchronized with the video frames, to obtain an initial coarse registration. A combination of techniques using telemetry data for initial estimate, coarse matching using visual appearance features and precise alignment using a DEM map is suggested in (Kumar, R., 1998). A multi-step process of projecting the reference, with elevation information to the video imagery, frame-to-frame alignment using an affine mapping and a local-to-global matching scheme for final precise geo-registration is described in (Wildes, R.P., 2001). Bundle block adjustment is used to extend these schemes to highly oblique and zoomed in video imagery (Kumar, R., 2000).

Drawbacks of existing techniques are:

Excessive reliance on sensor data: The availability of telemetry data accompanying the video requires high cost sensors such as inertial navigation systems. The accuracy of such systems is limited, when low cost alternatives are used.

Motion of both subject and platform: Every case considered in the literature review above is an application where either the airborne video of a static scene is taken from a moving platform or stationary cameras monitor movement of objects and humans. Our case is unique in the sense that there is motion of both the platform and the scene being videographed.

Inadequate stabilization: Stabilization only removes the shakiness in handheld video, but does nothing to correct for the roll, pitch and yaw of the helicopter.

Availability of elevation maps: It is assumed in most cases that a high resolution and accurate elevation map is available for every area, which is not always the case. Also, extraction of the elevation grid for every frame can be extremely computation intensive.

Semi-automation: Manual intervention in video registration involving registration of a very large number of frames is not only cumbersome and expensive in terms of time and labor, but can be less accurate than mathematical techniques.

Our approach

A feature tracker is used to automatically track features through the sequence of images in the video. These feature location correspondences are used as control points to compute a polynomial transformation function to warp every frame in the sequence successively to the reference. The reference frame is chosen to be one of the frames in the video segment, based on its location and position in the sequence. The geometric registration is done only in 2-D, assuming a flat scene. Extension to 3-D with a DEM or stereo analysis has not been necessary to date, but could be incorporated.

KLT FEATURE TRACKER

Formulation

The Kanade-Lucas-Tomasi feature tracker (Shi, J., 1994) is used to track features and obtain control points between successive frames of the video. The approach is to model the feature motion as a pure translation and track the features in windows, based on minimization of the sum of squared intensity differences in consecutive frames. A feature is considered good if it can be reliably tracked.

The image motion is modeled as a function I(x, y, t) with the (x, y) pair defining the spatial location of the pixel and the variable t is the temporal locator index, within the sequence. In our case, the symbol I can be thought of as representing the pixel intensity. Thus this function satisfies the following property:

$$I(x, y, t + \tau) = I(x - \varepsilon, y - \eta, \tau) \quad (1)$$

which means that an image taken an instant τ later is considered to be shifted from the original image by $\mathbf{d} = (\varepsilon, \eta)$, called the *displacement* in time τ.

Dropping the time variable, we define $J(\mathbf{x}) = I(x, y, t + \tau)$ and $I(\mathbf{x-d}) = I(x-\varepsilon, y-\eta, t)$. The image model now becomes $J(\mathbf{x}) = I(\mathbf{x-d}) + n(\mathbf{x})$, where n is the noise term. The displacement **d** is chosen to minimize the following error integral over a window W:

$$e = \int_W [I(\mathbf{x} - \mathbf{d}) - J(\mathbf{x})]^2 w d\mathbf{x} \quad (2)$$

w is a spatial weighting function, which can be unity or a Gaussian function, for example, to preferentially weight the centre of the window.

Solving for the displacement vector

For a small displacement vector, the image intensity can be approximated by the truncated Taylor series as $I(\mathbf{x-d}) = I(\mathbf{x}) - g\mathbf{d}$. The residue ε can now be written as

$$e = \int_W [I(\mathbf{x} - \mathbf{d}) - J(\mathbf{x})]^2 w d\mathbf{x} = \int_W [I(\mathbf{x}) - g\mathbf{d} - J(\mathbf{x})]^2 w d\mathbf{x}$$
$$= \int_W (h - g\mathbf{d})^2 w d\mathbf{x} \quad (3)$$

where $h = I(\mathbf{x}) - J(\mathbf{x})$. Differentiating the last term in the previous expression with respect to **d** and equating to zero yields:

$$\int_W (h - g\mathbf{d})\mathbf{g} w dA = 0 \quad (4)$$

Since $(\mathbf{g} \cdot \mathbf{d})\mathbf{g} = (\mathbf{g}\mathbf{g}^T)\mathbf{d}$ and a constant displacement is assumed within each window W, we get

$$\int_W (\mathbf{g}\mathbf{g}^T w dA)\mathbf{d} = \int_W h\mathbf{g} w dA \qquad (5)$$

This scalar equation can be rewritten as

$$G\mathbf{d} = \mathbf{e} \qquad (6)$$

where $G = \int_W \mathbf{g}\mathbf{g}^T w dA$ and $e = \int_W (\mathrm{I} - \mathrm{J})\mathbf{g} w dA$, where h = I – J.

This equation is the basis for tracking. For every pair of frames, G is computed by estimating the gradient and it's second order moments. The vector **e** is then obtained using the difference between the two frames and the gradient **g.** The displacement is now the solution of the equation G**d = e**.

Selection of features to be tracked

The selection of good features is based on the reliable solution of equation (6). The matrix G must satisfy the following two conditions: It must be above the noise level of the image, meaning that the major and minor eigenvalues of G must be large relative to the noise variance. G must also be well conditioned, meaning that the two eigenvalues have to be of the same order of magnitude. If the minor eigenvalue is larger than the noise threshold, it means that G satisfies both the conditions, since the larger eigenvalue cannot be arbitrarily large. The noise threshold λ is chosen somewhere in between the upper and lower bound eigenvalues determined from regions of high texture and low texture, respectively.

The minor eigenvalues for each window are sorted in descending order and the windows with high values are considered the best windows for tracking. The pixel co-ordinate to achieve correspondence is chosen at the center of the window. For this reason, the width and height of the window need to be an odd number of pixels. Overlapping windows can be avoided by deleting all the new features that come within an already selected window.

The tracking window

The choice of window size is crucial for the tracker to function well. A small window is sensitive to noise, since only a few pixels are being tracked, and they might change due to different illuminating conditions caused by shadows or reflection from objects in the image. The tracking is more robust with larger windows, but the computation time can be large, even for a fairly short sequence of 30 seconds, corresponding to 900 frames (at 30 frames/second). The computation time on a Sunblade 2000 (UltraSPARC3 900MHz) is of the order of a few seconds for processing each frame.

A minimum distance of 30 pixels is enforced as a separation between features, so that the features are spread apart and not all collected together in regions with a lot of detail, i.e., regions of high frequency content. The size of cars is between 15 to 20 pixels in width and 5 to 10 pixels in height. A window width of 7 pixels and height of 17 pixels is used for finding and tracking the features in our data. This window size is robust to noise, but not so large that the computation time is too high. Choosing the window size of 7x17 avoids placing the entire window on a car. A smaller window or a window of 17x7 would, for instance, coincide entirely with a car. It is then possible that the feature tracker finds a feature on the car, since the cars have corners and some detail, such as the windshield, visible from an airborne video. The choice of a window so that non-car pixels are included in the window avoids finding features on the car.

REGISTRATION ALGORITHM

The process to register two frames is a two-stage refinement where the control points are obtained from the correspondences identified by the feature tracker.

Steps in the process

1. The feature tracker gives the control points, which are correspondences between consecutive frames I_i and I_{i+1}.
2. Except for the first frame (i.e., i not equal to 1), the points $p1_i$ on the reference image I_i are transformed using the previously computed transformation coefficients $T2_{i-1}$ to yield the set of points $p2_i$.
3. Coefficients $T1_i$ are calculated based on N correspondences between set of points $p2_i$ and $p2_{i+1}$.
4. The error at each of these N control points is found and ranked in ascending order.
5. N/2 control point correspondences which give the least error are picked from the ranked list.
6. These N/2 points are used to re-compute transformation coefficients $T2_i$.
7. The warped image I^T_i is obtained by applying the transformation to the image I_i.

Implementation details

A polynomial distortion model is used to transform the coordinates in the current image to the reference image. The model is global in the sense that a single transformation is applied to the entire image. This is sufficient in our case where the motion of the camera is approximated by a translation between two consecutive frames, which are $1/30^{th}$ of a second apart. The model is given below: (Schowengerdt, R., 1997)

$$x = \sum_{i=0}^{N} \sum_{j=0}^{N-i} a_{ij} x_{ref}^{i} y_{ref}^{i}$$
$$y = \sum_{i=0}^{N} \sum_{j=0}^{N-i} b_{ij} x_{ref}^{i} y_{ref}^{i} \qquad (7)$$

A four-term polynomial, which is an affine transformation, with an additional xy term to compensate for the x-dependent scale in y and y-dependent scale in x, is used here.

$$x = a_{00} + a_{10} x_{ref} + a_{01} y_{ref} + a_{11} x_{ref} y_{ref}$$
$$y = b_{00} + b_{10} x_{ref} + b_{01} y_{ref} + b_{11} x_{ref} y_{ref} \qquad (8)$$

As seen from the above equation, we have four unknowns to be solved for to completely obtain the transformation. Hence, to get an exact solution, four control points are sufficient. A large number of features are tracked since large inter-frame motion due to sudden platform movement, different lighting/viewing conditions and changing field of view cause loss of feature points. About 50 control points are used in the first iteration to compute the transformation. This is subsequently reduced to 25 after the error checking bounds are applied.

REGISTRATION RESULTS

The registration progresses in a cumulative way, registering every frame to the previous frame and thereby registering all the frames in the sequence to the reference. The frame 150 shown below is the reference to which the entire sequence is registered. As can be seen, the earlier frame 50 is registered to align well with locations in the reference. There is residual error in the registration, which we call jitter.

Figure 1. Reference frame 150

Figure 2. Distorted frame 50

Figure 3. Frame 50 registered to frame 150

Jitter Calculation

The calculation of jitter is done by choosing four ground points in the reference image, all of which are visible through the entire sequence. The points here correspond to the easily identifiable corners of buildings. The jitter is computed as the average of the error at these four points, with respect to the reference frame, which happens to be the first frame in the video sequence. Both the x and y axis jitter are found to be within a few pixels. This enables accurate velocity measurement from the video, since the velocity of vehicles (about 30-50 mph inside the city) corresponds to displacement of a few hundred pixels over several seconds. The RMS jitter is 1.58 pixels and 1.64 pixels for the x and y axis, respectively.

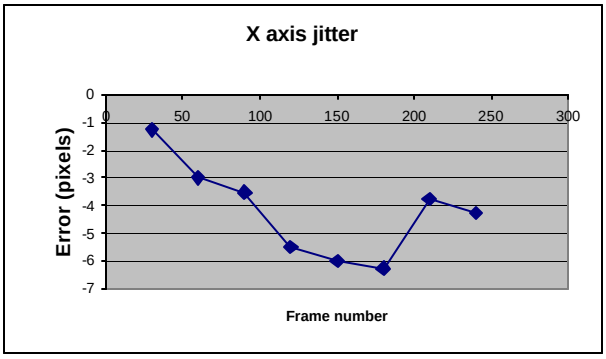

Figure 4. X-axis jitter vs. frame number

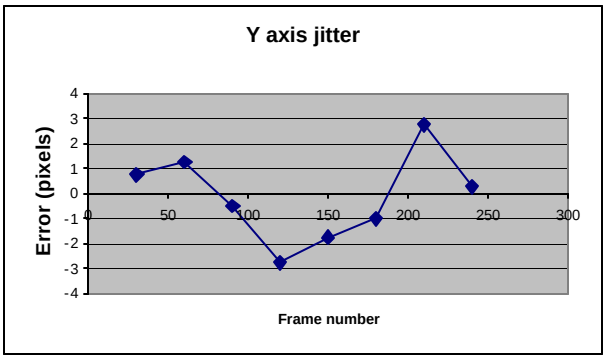

Figure 5. Y-axis jitter vs. frame number

MULTIMEDIA GIS

The GIS has the following layers:

- The base layer consists of six PAG (Pima Association of Governments) orthophotographs at 1-foot resolution, which were mosaiced to cover our 3 x 2 sq. mi. area of interest around the University of Arizona campus.
- The second layer is the roadmap layer formed by manually digitising the non-roadway segments for the area corresponding to the base layer.
- A polygon shapefile layer is the next layer, introduced to link to the multimedia data:
    - Four band multi-spectral ADAR (Airborne data acquisition and registration) images at 0.7m resolution.
    - Recorded and registered airborne video, from the helicopter flights conducted on 05/16/01 and 02/26/02.
    - Real time video from autoscope cameras positioned along major arterial roads in the city, streaming across the web, the implementation of which is in progress.

The GIS is created in ArcView and the multimedia objects reside on the users hard disk and are opened using standard Windows file associations. There are icons in the GIS, indicating locations with available multimedia content. Clicking on these brings up a menu to interface with the multimedia data.

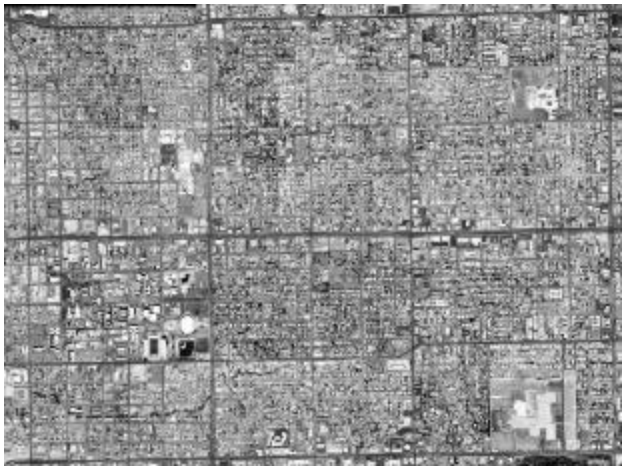

Figure 6. Mosaiced base-layer

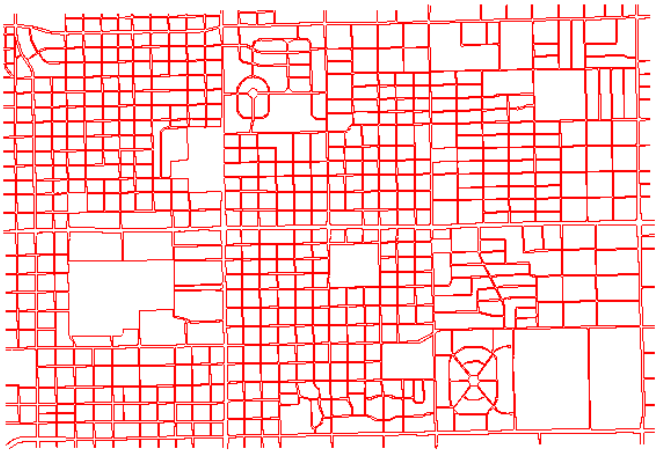

Figure 7. Digitized roadway map

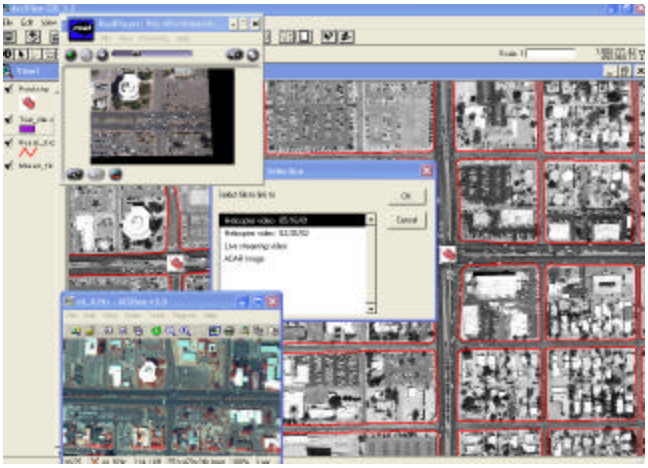

Figure 8. Snapshot of the multimedia GIS

CONCLUSIONS

An automated registration scheme capable of correction of airborne video has been described. Such a system also enables visualization of traffic flow as seen from a stationary platform. A multimedia GIS incorporating the registered video along with other relevant geographical data, including imagery, is created. This combines the functionality of a traditional GIS with the power of multimedia objects.

ACKNOWLEDGEMENT

This work is a collaboration between the Digital Image Analysis Laboratory (DIAL) and the Advanced Traffic and Logistics Algorithms and Systems (ATLAS) Laboratory (Pitu Mirchandani, Director) at the University of Arizona.

This research was supported by the U.S. Department of Transportation Other Transaction Agreement DTRS56-00-T0004 on Remote Sensing of Transportation Flows to the ATLAS Laboratory.

REFERENCES

Fonesca, A., C. Gouveia, S. C. Antonio, F. Francisco. Environmental impact assessment using multimedia GIS. in Proceedings of the Fifth European Conference and Exhibition on Geographical Information Systems, EGIS/MARI `94. 1994. Paris.

Panagopoulou, G., S. Sirmakesis, A. K. Tsakalidis, ATHENA: Integrating GIS and Multimedia Technology; The Design of a Tourist Information System for the County of Attica. in Proceedings of the Fifth European Conference on Geographical Information Systems (EGIS/MARI '94) , EGIS Foundation. 1994. Paris.

Kumar, R., H. S. Sahwney, J. C. Asmuth, A. Pope, S. Hsu. Registration of video to geo-referenced imagery. in Proceedings of the International Conference on Pattern Recognition. 1998. Brisbane, Australia.

Kumar, R., S. Samarasekera., S. Hsu, K. Hanna. Registration of highly-oblique and zoomed in aerial video to reference imagery. in Proceedings of the 15th International Conference on Pattern Recognition. 2000. Barcelona, Spain.

Park, R. H., D. G. Sim, Localization based on DEM matching using multiple aerial image pairs. *IEEE Transactions on Image Processing*, 2002. **11**(1), p. 52-55.

Schneider, B. Integration of analytical GIS-functions in multimedia atlas information systems. In Proceedings of the International Cartographic Conference. 1999. Ottawa, Canada.

Schowengerdt, R., Remote Sensing - *Models and methods for image processing*. 1997: Academic Press.

Shi, J. and T. Carlo, Good features to track. Proceedings of IEEE Computer Vision and Pattern Recognition, 1994: p. 593-600

Soomro, T., T. Zheng, and Y. Pan. Html and multimedia web gis. in Proceedings of the Third International Conference on Computataional Intelligence and Multimedia Applications. 1999. New Delhi, India.

Wildes, R. P., D. J. Hirvonen., S. C. Hsu, R. Kumar, W. B. Lehman, B. Matei, W. Y. Zhao. Video Georegistration: Algorithm and Quantitative Evaluation. in Proceedings of the IEEE International Conference on Computer Vision. 2001, Vancouver, Canada.