Multi-sensor fusion and enhancement for object detection
NNL04AA51G

Summary of Research
submitted to
NASA Langley Research Center
Hampton, Virginia 23681

Principal Investigator
Zia-ur Rahman
Research Associate Professor

Department of Applied Science
College of William & Mary
PO Box 8795
Williamsburg, VA 23187-8795
ATTN: Grants & Research Administration

February 15, 2005

This is the final report for NASA cooperative agreement NNL04AA51G and covers the period from 17 May, 2004 to 02 July, 2004, the complete performance period.

1 Description of Research

This was a quick 6-week effort to investigate the ability to detect changes along the flight path of an unmanned airborne vehicle (UAV) over time. Video was acquired by the UAV during several passes over the same terrain. Concurrently, GPS data and UAV attitude data were also acquired. The purpose of the research was to use information from all of these sources to detect if any change had occurred in the terrain encompassed by the flight path.

The following three issues needed to be addressed:

1. **Changes in lighting conditions**: Because the images are acquired over a period of hours or a few days, the lighting condition under which the images are acquired are bound to differ. A comparison of the same scene over two different flights may, thus, produce false alarms solely due to the changes in the strength and orientation of shadows.

2. **Co-registration and rectification of data**: Because the images are acquired from an airborne mobile platform, even with the availability of the GPS coordinates, data from different excursions would be different because of slight changes in flight path due to wind direction and speed, and due to attitude changes in the aircraft. The data would have to be rectified and registered before any changes in terrain can be detected.

3. **Possible presence of haze and dust in the FOV**: Haze and dust make the task of automatic extraction of detail more difficult. The impact of haze is to blur the image, thus making it very difficult to resolve fine details. So image-processing techniques that can mitigate the effects of these conditions need to be applied to the data before performing change detection.

2 Proposed solution

Two of the three problems stated above—images taken under hazy conditions and changing lighting condition—can be resolved, or, at least mitigated, by applying the Multi-Scale Retinex (MSR) algorithm.[1, 2, 3]. The MSR is a general-purpose, non-linear image enhancement algorithm that provides simultaneous dynamic range compression, lightness constancy, and sharpening. The dynamic range compression property allows the retinex to enhance details that are otherwise obscured by shadow or smoke/haze/dust, Figure 1. Figure 2 also shows the application of the MSR to images taken under very poor visibility conditions other than haze and fog. This example shows a nighttime image where the information in the original is very hard to interpret. The MSR processed image provides a 5-10 times boost in the visibility of features. Both of these Figures show how the MSR can be used to overcome the problems that arise from the third issue listed in Section 1.

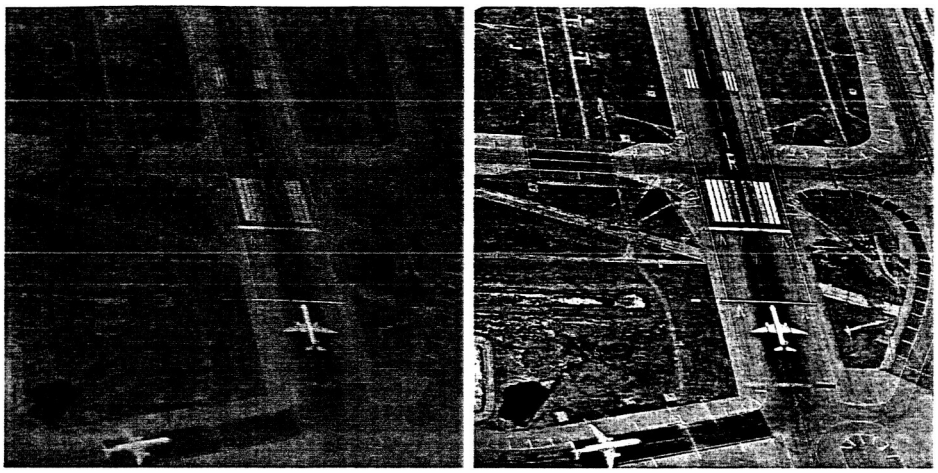

Figure 1: Original image (left) and the MSR processed image (right). The haziness in the original is almost completely removed in the processed at the slight cost of flattening some features.

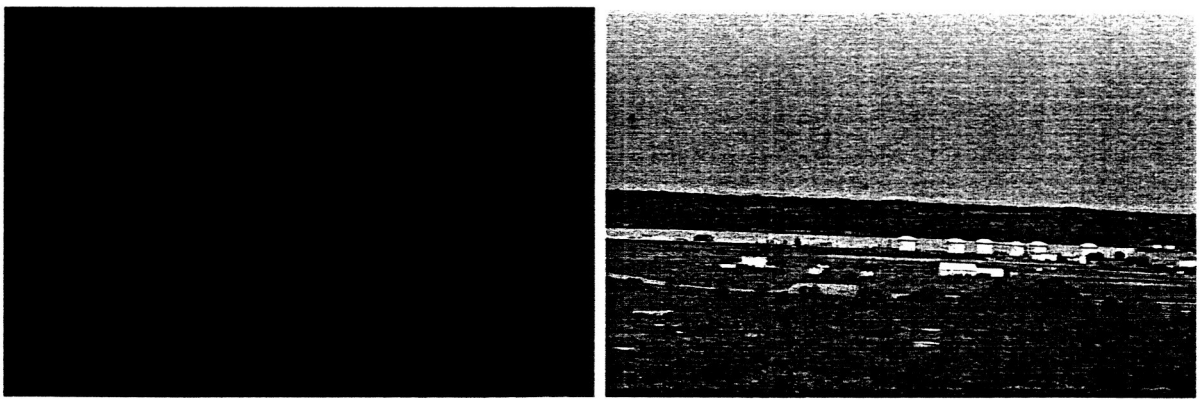

Figure 2: Original image (left) and the MSR processed image (right). The impact of low-light levels during the acquisition process is considerably ameliorated by the MSR process.

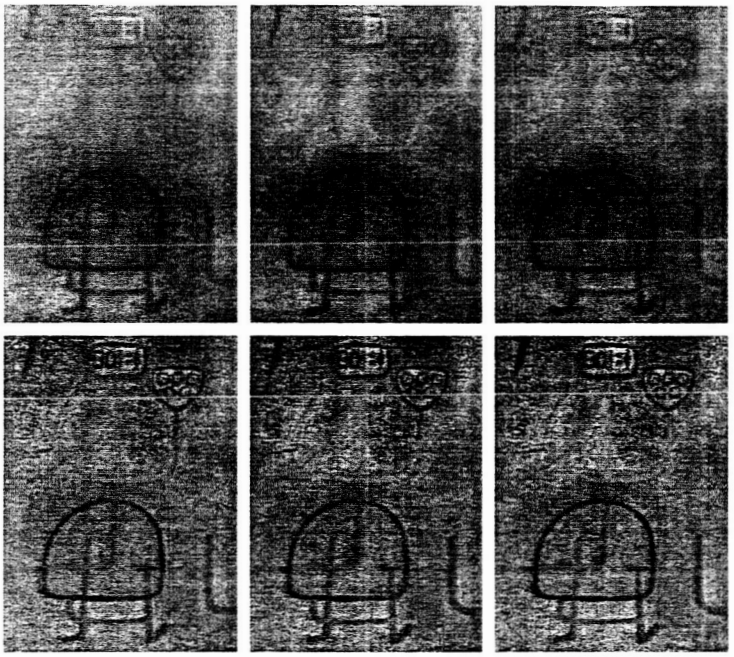

Figure 3: Original images taken under three different lighting conditions(top) and the MSR processed images (bottom row). The impact of the illuminant is removed from the processed images by the MSR process.

The lightness constancy property of the MSR allows it to (almost) completely remove the impact of lighting on the processed image. Figure 3 shows an example of this behavior, where a painting has been imaged under three different lighting conditions, in simulation. Though the three acquired images look substantially different, the MSR output for each is remarkably the same. This suggests that the MSR processed data is (almost) independent of the lighting source. Thus, if images of the same terrain were taken on different days and at different times, the effects that can be attributed to lighting changes can be minimized with this process. This addresses problem 1 listed in Section 1.

The MSR can thus be used to handle both the problem of changing lighting conditions, and hazy/dusty atmospheric conditions. That still leaves the significant problem of co-registration and rectification of data acquired over different time periods and altitudes. The only real requirement that we had placed on proposing a solution to this problem was that the video data be GPS indexed. In other words, we need a frame of reference for each video frame so that we can roughly align the fly-overs over the same terrain. Once this has been done, we can use the recorded attitude information to perform fine-grain alignment. We had previous experience of aligning data from multi-sensors and, in conjunction with Glenn Hines of NASA Langley Research Center, had developed software to perform this task. This software was an outgrowth of some recent work under the aegis of the Aviation Safety Program (AvSP). That research was concerned with efficient enhancement and fusion of multi-sensor data. Since the data was acquired by 3 different sensors, one of the first task was to perform co-registration and rectification of data in order to remove any

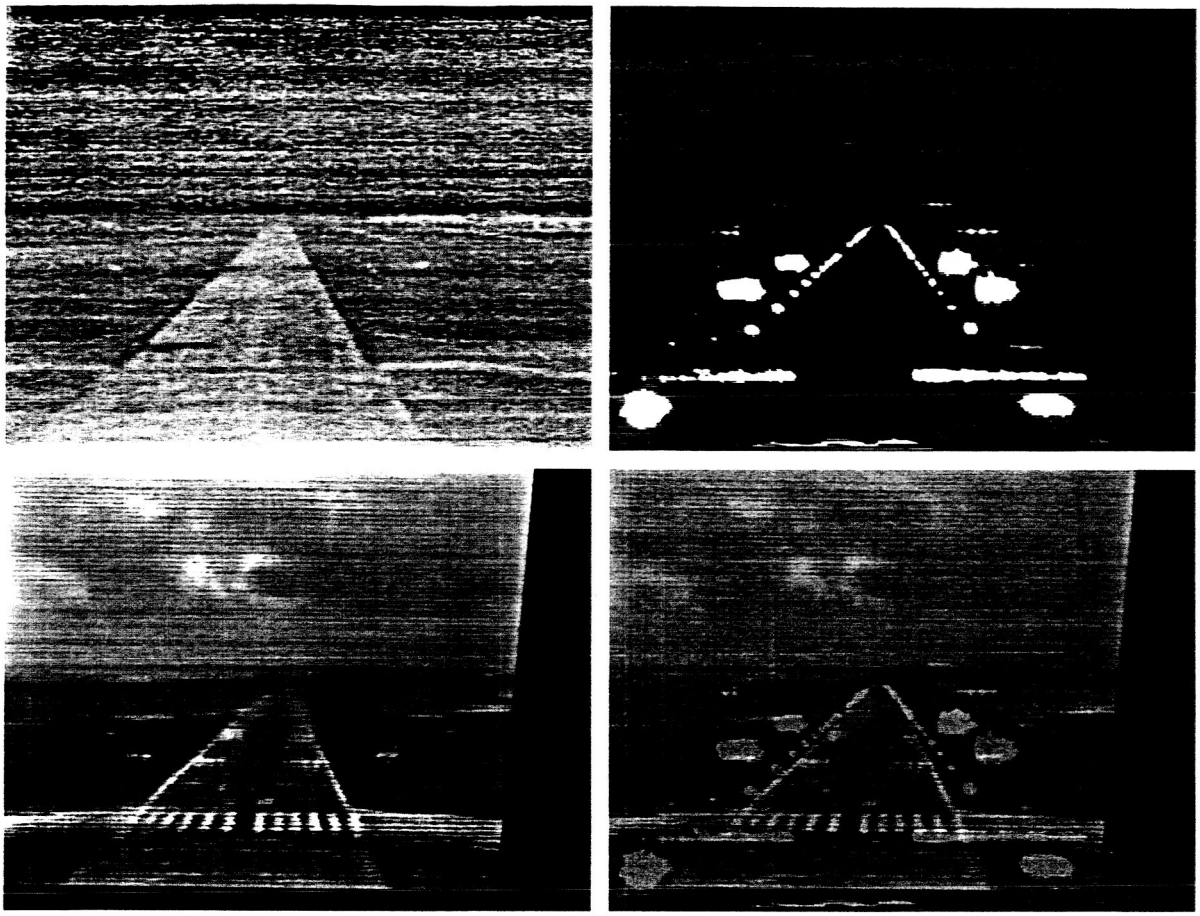

Figure 4: LWIR (top-left), SWIR (top-right), RGB (bottom-left), and fused (bottom-right). The fused image synergistically combines the information present in the LWIR, SWIR, and RGB data streams into a single representation. These data streams have been enhanced prior to fusion.

anomalies that could have crept into the data due to sensor misalignment[4]. Figure 4 shows an example of data where this algorithm has been used to correct for the sensor misalignment. Also shown is the fused result that contains more information than any of the three sensor channels do individually.

Figure 4 (bottom-right) deserves some additional explanation. As a part of the AvSP program, we performed data fusion between three diverse imaging sources: long-wave infrared (LWIR), short-wave infrared (SWIR) and visible (RGB)[5]. We converted the RGB image to grayscale before processing. The main thrust of performing fusion is that the different sensors can provide complementary information under differing visibility conditions. For instance, the LWIR and SWIR images can contain information that is not visible in the RGB image under very poor lighting conditions or under conditions where there is smoke. Conversely, the RGB image can provide more information on a bright, sunny day than can either LWIR or SWIR. Fusing the data streams ensures that the best information is represented in the final image regardless of whether it were a better

lighting situation for one or the other type of sensor. Since we already have experience in data fusion via the AvSP, if multi-sensor data had been available for the problem under consideration, we could have provided fused data as well. As Figure 4 (bottom-right) shows, the fused image contains significant features from all three imaging sources compactly in a single display.

Once the rectification and enhancement have been performed on the data, the task of detecting what has changed between images acquired over different time periods still remains. Several different approaches can be brought to bear on this problem which has now become simpler because of the pre-processed data. We list two simple, yet robust, techniques here.

1. **Simple frame differences**: Frame differencing, i.e., the concept of subtracting two frames from each other to seek the changes between the two frames is well established in image-processing research. If the only changes in the scene are due to the presence of objects, then this method will provide the necessary detection. Our pre-processing is geared toward ensuring that the changes in scenes are not due to misalignment of frames, or due to lighting changes or hazy/dusty atmospheric conditions. So this offers a simple, yet solid, way of detecting changes in the scene.

2. **Edge enhancement and thresholding pre-processing**: The images can also be edge-enhanced and thresholded to remove objects that are below a certain size and hence not of interest. Once this has been applied to the images, frame-differencing can be used to detect differences. This further reduces the impact of changing imaging conditions on the final image.

3 Issues

From early experiments, it was evident that the MSR processing was able to substantially reduce, if not eliminate, the impact of lighting changes on the imagery. The resulting differences in shadow strengths were small enough so as to not trigger false alarms. However the rectification and co-registration turned out to be bigger issues than anticipated. There were several reasons for this and we will address them below.

3.1 GPS referenced data

As we noted above, one of the few requirements we placed on finding a viable solution to the change-detection problem was the availability of concurrently acquired video, GPS and attitude data. In actuality, this was not the case. Table 1 shows the rates at which the data was actually acquired for the three different information streams.

The typical airspeed of the UAV was 60knots which is roughly 70 miles per hour, or 102 feet per second (fps). Having the GPS update rate of 1Hz, we can align the video data to within 100 feet accuracy which is too coarse a resolution for the kind of changes that we were attempting to detect.

Data Type	Video	GPS	Attitude
Data Rate	30Hz	1Hz	5Hz

Table 1: Data acquisition rates for video, GPS, and attitude information.

Additionally the attitude data update rate is 5Hz which means that the UAV has traveled 20 feet in between updates. Under calm weather conditions, it is possible that there were no changes in attitude, i.e. yaw, pitch or roll, during those 20 feet, or 0.2seconds. In practice, however, the coarse temporal resolution of the updates makes it near to impossible to automatically align the data over different flight passes. So perfect alignment using the data streams is only possible if the update rates of the data streams are (a) at the same temporal resolution, and (b) in synchrony with each other. We will return to this latter issue below.

We were able to achieve some rough grained alignment though. Using the coarse GPS update rate, we targeted only those video frames for which GPS data was available. Using this, we could find common points between two flight paths that lay with a pre-defined radius of each other. After detecting a sequence of such points, we could with some confidence state that the terrain was common to the two flight paths that were being examined. We then used the slightly finer grain attitude information to correct for yaw, pitch, roll and elevation changes to rectify the data from the two frames to a common grid. A simple differencing at this point would allow us to highlight the changes in the terrain between two flights.

3.2 Camera optics and UAV speed

As we pointed out above, the UAV's typical speed was approximately 100 fps. There were two different cameras that were used with the UAV to acquire video data. Though these cameras have the same spatial resolution in terms of the size of the CCD and the number of pixels in each dimension, they differ greatly—with accompanying problems—from each other in their FOVs. The table below shows the respective FOVs. Using simple trigonometry, we can compute the footprint

Camera	**FOV (H)**	**FOV (V)**
1	20°	17°
2	58°	48°

of the FOV on the ground. For an altitude of h, the horizontal, x, and vertical, v, dimensions of the footprint can be computed by

$$x = 2h\tan(FOV(H)) \quad (1)$$

$$y = 2h\tan(FOV(V)) \quad (2)$$

Results obtained by using Equations 1 and 2 are shown in the table below:

There were two different kinds of problems that arose from this conjunction of camera FOV and UAV speed.

Camera	FOV (H)	FOV (V)	Altitude: 200ft		Altitude: 400ft	
			Horizontal	Vertical	Horizontal	Vertical
1	20°	17°	70'	60'	141'	120'
2	58°	48°	221'	173'	443'	356'

1. In order to process the video data, we need to be able to access individual frames. Each frame is formed from two interlaced fields. With the UAV flying at 100 fps, and a video field update rate of 60Hz—30Hz frame rate, 2 fields per frame—the UAV moves 1.67′ between acquisition of each field. Since each frame has 640 × 480 (columns × rows), at an altitude of 200′, this translates into a motion of roughly 15 × 13 pixels for Camera 1. This is a huge amount of movement between fields so the frames have to be deinterlaced before processing. Since the fields only have full resolution in one direction, deinterlacing introduces interpolation, and hence blurring, into the data. This problem is not as severe at an altitude of 400′ but is still significant. Typically, the UAV cruises at an altitude of 400′ but there are some constraints on that that are discussed below.

2. Camera 2 does not have the same motion problems as Camera 1 because of the larger FOV. The movement in terms of pixels for Camera 2 at 200′ is roughly 5 pixels. However, the very large FOV of camera 2 had its accompanying problems: barrel distortion. This lens introduces a non-linear distortion at the edges of the image. This additional source of misalignment, because of its non-linear nature, makes it almost impossible to align data from two different flight passes.

3.3 Camera Optics and spatial resolution

There is a minimum requirement on the size of changes that need to be detected. The expectation is to be able to detect if an object roughly $1' \times 1.5'$ has been placed along the flight path in between flights. The instantaneous FOV (IFOV) of the camera is a measure of angular resolution. Let us assume that in order to see an object that is 1.5′ long, its projection on the image needs to be at least 10 pixels long. The IFOV for the cameras is shown in the table below: Using Equations 1 and

Assume the cameras are $640(H) \times 480(V)$ pixels		
Camera	IFOV (H)	IFOV (V)
1	0.03125°/pixel	0.03542°/pixel
2	0.09063°/pixel	0.10000°/pixel

2, each pixel covers $(5.45h \times 6.18h) \times 10^{-4}$ feet for Camera 1. In order for the camera to be able to have a 1.5′ feature be 10 pixels long, the corresponding altitude would be about 170′. As we have seen above, an altitude of 200′ causes significant deinterlacing problems. So our two requirements appear to be in conflict: the UAV needs to be able to fly at a lower altitude in order to resolve

small object but needs to fly at a higher altitude in order to acquire data that is (a) not blurry and (b) spatially contiguous.

We recommend using a lens with a eider FOV than the one currently being used on the UAV.

3.4 Synchronization

As we stated above, the data from the GPS, attitude information from the navigation unit, and the video stream is acquired and recorded at different update rates. To compound the problem, it is difficult to synchronize the data from the navigation unit, which is in text format, with the video data, which is binary. As we have also shown above, an alignment error of a single frame can have significant impact on how the GPS and video data correlate with each other. With this in mind, we suggested that the video data acquisition and GPS data acquisition start concurrently. However, this requirement was not fulfilled because the navigation unit starts recording when it acquired a GPS lock and the acquisition of video data cannot be linked to that event. We then suggested a series of "ON/OFF" events to synchronize the two data streams. This consisted of turning off recording for the navigation unit and powering OFF and ON the video recorder three times. The series of three OFF/ON events in the navigation data, and three BLANK series of frames would be sufficient to align the data. However, since the video module takes time to initialize, this procedure was also not entirely successful. Additionally, several seconds need to be allowed between these events for the video unit to stabilize.

In order to better align the video and navigation data, we recommend that instead of powering the video unit ON and OFF, an electronic shutter be placed on the lens. This shutter can then be used to mark the start of video data.

4 Experiments

Despite all of the problems that we have outlined above, we did have some success in performing the task that we set out to do. We used the following procedure:

1. Initialize the video data stream with the recorded navigation data by using the series of ON/OFF events described above.
2. Use the 1Hz GPS information to find potential points of alignment. These would be points where the FOVs from the two different flights have sufficient overlap, i.e., $> 50\%$.
3. Find the video frames that correspond to those points of alignment.
4. Since attitude information is available at a 5Hz update rate, we can use an additional 4 frames from the data stream and rectify them for roll, pitch and yaw.

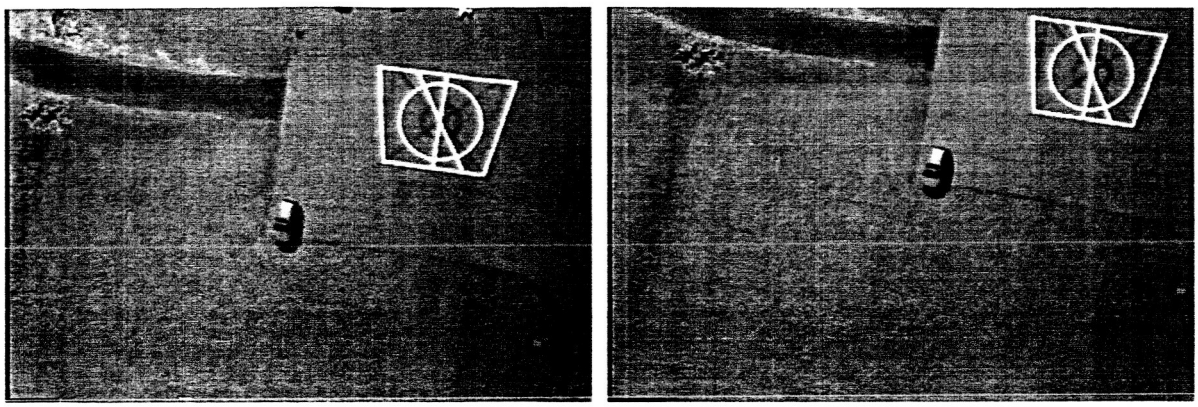

Figure 5: Frames from two different sorties that have substantial overlap. These frames were automatically selected using GPS information.

5. These frames can then be registered to find differences.

With the data that we had and the attending problems, lack of synchronization being the major one, we were unable to automate this process. However, we were able to locate frames from different data streams that had sufficient overlap and were able to adjust for attitude changes. We then uses manual registration to obtain the final results. Figure 5 shows two initial video frames from two different flight events selected automatically by using the GPS data. The frame from the second video stream (Figure 5 right) was then manually registered with the the frame from the first video stream (Figure 5 left). Figure 6 shows the video frame from the second stream (left) and its corrected version (right). And, finally, Figure 7 shows the original frame from the first video stream (left), the corrected frame from the second video stream (middle), and their difference (right). There were no changes detected in these data. For this experiment, since there was very little temporal delay or lighting variation between the two flights, we chose not to use the MSR for lightness constancy.

5 Conclusion

This was a short, 6-week attempt to solve a problem that turned out to be more complex than originally anticipated. The complexity arose from the following factors:

1. The three different data stream, GPS, navigation, and video had (significantly) different update rates.

2. The three data stream were not synchronized. Additionally, attempts to synchronize the data were not always successful.

3. There were video drop-outs in the acquired data, whenever the UAV got too far. Additionally there was line jitter making the video useless for detecting small changes.

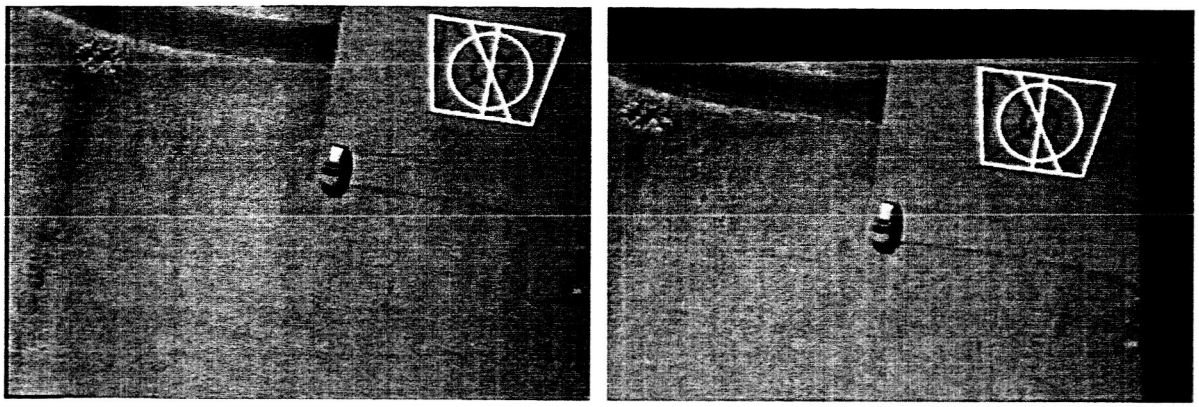

Figure 6: The frame from the second video stream was corrected to match the orientation of the frame from the first video stream. The original is shown on the left and the corrected image on the right.

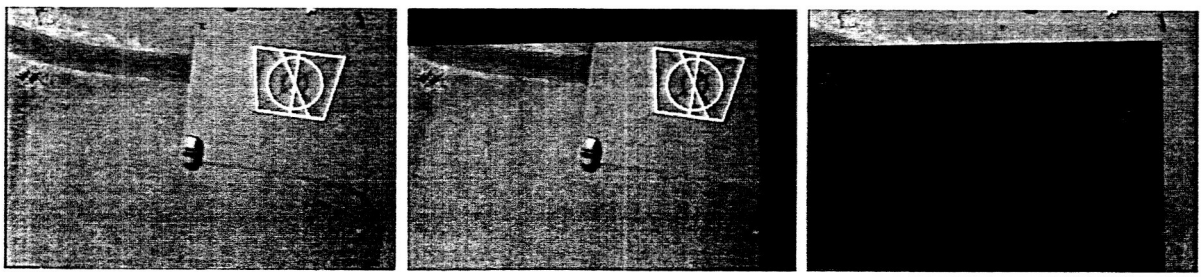

Figure 7: The frame from the first video stream (left), the corrected frame from the second video stream (middle), and their difference (right). No changes were detected.

4. The relationship between the altitude of the UAV, FOV of the sensors, and UAV speed was not carefully analyzed. This resulted in a mismatch between the requirements to resolve small features and data update rates.

We were successful in using the GPS information and the navigation data to obtain corresponding frames from different data streams. However, the lack of synchronization made it difficult to use the navigation information such as altitude changes, roll, pitch, and yaw to correct the data between two flights. Manual registration shows that object detection can be performed given synchronized data streams with same temporal resolution.

We should also note here that the GPS and navigation information were only available *after* the completion of the flight. Hence any real-time object detection scheme would entail transmitting the navigation and GPS information back in addition to the video data.

We make the following recommendations:

1. Use a GPS unit with an update rate higher than 1Hz. There are several units on the market with update rate > 20Hz.

2. If possible, use a navigation unit that updates information at a higher rate. Ideally, a unit that can transmit information during flight would be used.

3. Camera optics need to be closely matched with the dimensions of the object that needs to be detected from a particular altitude.

4. The UAV speed needs to be carefully matched to the FOV of the sensor, and the dimensions of the object that needs to be detected.

5. A synchronization scheme between all the data streams needs to be developed. We recommend using an electronic shutter on the camera so that synchronization blanks can be inserted in the video stream. Powering the camera ON and OFF did not provide satisfactory performance because the initialization time—i.e., time it takes the CCD to stabilize—is unknown.

We still feel that this is a doable task, given enough time and resources. We were able to get decent results manually despite issues surrounding synchronization, etc. addressed above. For fully automatic operation, better synchronized data at a higher update rate is essential.

References

[1] Z. Rahman, D. J. Jobson, and G. A. Woodell, "Retinex processing for automatic image enhancement," *Journal of Electronic Imaging* **13**(1), pp. 100–110, 2004.

[2] D. J. Jobson, Z. Rahman, and G. A. Woodell, "Properties and performance of a center/surround retinex," *IEEE Trans. on Image Processing* **6**, pp. 451–462, March 1997.

[3] D. J. Jobson, Z. Rahman, and G. A. Woodell, "A multi-scale Retinex for bridging the gap between color images and the human observation of scenes," *IEEE Transactions on Image Processing: Special Issue on Color Processing* **6**, pp. 965–976, July 1997.

[4] G. D. Hines, Z. Rahman, D. J. Jobson, and G. A. Woodell, "Multisensor image registration for an enhanced vision system," in *Visual Information Processing XII*, Z. Rahman, R. A. Schowengerdt, and S. E. Reichenbach, eds., pp. 231–241, Proc. SPIE 5108, 2003.

[5] Z. Rahman, D. J. Jobson, G. A. Woodell, and G. D. Hines, "Multi-sensor fusion and enhancement using the retinex image enhancement algorithm," in *Visual Information Processing XI*, Z. Rahman, R. Schowengerdt, and S. E. Reichenbach, eds., pp. 36–44, Proc. SPIE 4736, 2002.