Investigating Limitations of SURF Approach for Thermal Imaging Analysis and Mapping

Dan McMillen

Abstract: Wildlife conservation involves an understanding of a specific animal, its environment and the interaction within a local ecosystem. Unmanned Aerial Vehicles (UAVs) present cost effective, non-intrusive solution for detecting animals over large areas and the use thermal imaging cameras offer the ability detect animals that would otherwise be concealed to visible light cameras. This report examines some of limitations on using SURF for the development of large maps using multiple stills images extracted from the thermal imaging video camera which contain wildlife (eg. Koala in them).

Keywords: Thermal Imagery, UAV, SURF

Notation

UAV – Unmanned Aerial Vehicle

GPS – Global Positioning System

SURF – Speeded Up Robust Features

ARCAA - Australian Research Centre for Aerospace Automation

GPIO – General Purpose Input / Output

CPU – Central Processing Unit

SOC – System On a Chip

RAM –Random Access Memory

1. Introduction

Wildlife conservation involves an understanding of a specific animal, its environment and the interaction within a local ecosystem. One important component is animal population and its geographic distribution and movement. The ability to collect accurate location data of multiple animals over a large area allows for accurate population estimates and geographic distribution. Unmanned Aerial Vehicle design, path planning for remote sensing is an active field of research. UAVs present cost effective, non-intrusive solution for remote sensing detection of animals over large areas [1, 2, 3, 4, 5, 6, 7, 8, 9]. For detection of a large animal one method takes advantage natural thermal emissions that exist in the long-wavelength infrared spectrum. The use thermal imaging cameras offer the ability detect animals that would otherwise be concealed to visible light cameras [10]. Specific location data of individual animals provides further advantages with the ability to specifically map know locations for further land based studies. Also multiple passes over a set area at set time intervals can provide more accurate movement estimates, territory and animal interaction information. The Global Positioning System provides extremely accurate location data.

The Koala Rescue research project at QUT investigates the use of thermal equipped UAS for identifying Koala across a region of interest. A part of the data pre-processing requires the collected data to be converted into a single stitched image, providing an overview of the object interest. The project goal is to convert a video aerial survey into a single stitched image and automate the process of image stitching in a MATLAB environment.

2. Literature Review

This project requires major development of software components to provide a platform for ground based software for processing. Research into the currently available literature is therefore critical to provide understanding of the current methods used and their effectiveness to produce similar results. This literature review aims to provide the best methods that can be implemented or explored in the development of this project.

We evaluated Pix4Dmapper [11]; a Software specialising in 3D mapping from UAV images with advanced functionality, extensive support and flexibility in hardware used. Agisoft PhotoScan [12] which is an alternate software specialising in 3D mapping from UAV images to Pix4Dmapper and similar functionality claims as Pix4Dmapper. Adaptive Filtering for Subject Detection techniques [13] was evaluated; this is a filtering technique for removing layers of unwanted data with considerations for subject parameters. This is to enhance subject/s and filtering method for background noise reduction. We also evaluated image Saturation and Daytime Data Acquisition [14] as a technique that demonstrates thermal imaging limitations and has techniques for optimal testing and data sets.

We also conducted research in Geo-referencing of Digital Imagery [15] and a methodology involved in geo-referencing and the application of ground control points (GCPs). We also considered Image Stitching and Digital Mosaics [15] and some of the methodology for image stitching. Wolf et al [15] highlights error created from relief and angle displacement between images and illustrates seam distortion and methods for correction and explains use of natural seams for optimal image stitching. The issue of Tilted and Oblique Images was also studied [15] who presented a discussion on the optimal coordinate system and the computational advantages of the omega-phi-kappa over tilt-swing-azimuth systems as well as the Concept of differential rectification and advantages over rectified images.

3. Design and Development

The thermal imaging video is assumed to have been processed and a folder which contains the individual image frames taken from the video is available. Details of the user input will be aided by screen grabs from the MATLAB programming environment.

The design and development consisted of the following phases.

1. Importing images into MATLAB environment and detect Speeded Up Robust Features (SURF).
2. Orientate and combine 2 images by comparing SURF features.
3. Extend to a large dataset of 20+ images and automate process.
4. Extend to full dataset.

To import the images, it is assumed the folder structure used. From a raw video a set of images is created and placed into a specific folder called '2-tif_metadata_output'. Importing the images into the MATLAB environment ensures that the program is minimalistic in code, easy to read and scalable. The MATLAB function *imageSet* is used. This function does not import the images directly into the ram but rather creates an easy reference that MATLAB can use to work on images more efficiently.

Once the images are imported/referenced the detection of features within the images is done. For the program to overlap, and ultimately stitch together an image, the features within a single image are analysed using an inbuilt function of MATLAB. This function does require the image to be grayscale, or one-dimensional matrix, to perform its analysis.

This is done by importing a single image and converting it to greyscale, then the greyscale image is passed through the feature detection function. The output of this is a matrix of points within the image that features are detected. A second function is used to extract the features at the previously extracted points.

Using this same method on two overlapping images results in extracted features appearing in both sets of data with each image having a different point for the image. These features are then compared to find where in the images the same features are present. Then the second image is orientated to the aspect of the first to ensure they overlap correctly before they are combined into a new image.

Extending the data sets to include more images follows the same process and is automated within MATLAB by increasing the number of images within the target folder.

Importing the images is done with a simple set of instructions:

```
% Load images.
imageDir = fullfile('2-tif_metadata_output');
imageScene = imageSet(imageDir);

% This reads the first image from the image set.
% Only images from dataset during flight to be used, all others should be
% deleted manually.
I = read(imageScene, 1);
```

This function makes it easy to reference images in the folder as shown on the last line of the above code box. To read/import any image in this set the read function is used which takes two parameters, the set and the image number. It should be noted that the image numbers are fixed to specific images which by default are sorted by name. This is useful in debugging by allowing an easy to find reference for an image should an error occur.

Image conversion, detection and extraction of the features within an image and their reference points:

```
% Initialize features for first image
grayImage = rgb2gray(I);
points = detectSURFFeatures(grayImage);
[features1, points1] = extractFeatures(grayImage, points);
```

First converting the image to grayscale using the function ‘rgb2gray’ with the input to the function being the variable which the image was read into, in this case ‘I’. Next ‘detectSURFFeatures’ produces a matrix of the points within the image features were detected. With these points the features can be extracted which provide the details necessary for the comparison with other images.

Prepare the images for overlapping:

```
indexPairs = matchFeatures(features2, features1);
matchedPoints = points2(indexPairs(:,1), :);
matchedPointsPrev = points1(indexPairs(:,2), :);
tforms = estimateGeometricTransform(matchedPoints, matchedPointsPrev);
```

The function ‘matchFeatures’ is used to find the features that exist within both images and create an array of their points on each image. At this stage the algorithm knows that for a given feature that exists in both images, its location in within each image. With these known points that need to be overlapped all that remains are manipulating the second images aspect and to do this the MATLAB function ‘estimateGeometricTransform’.

5. Results

Import of images into MATLAB environment is done by creating an image variable and utilising the MATLAB functions to import the image. For the SURF to work the image must be a 1-dimensional matrix so the image is converted to greyscale. Results for importing the images were successful, allowing for quick easy access to any images within the set directory. The conversion to greyscale utilises the optimised MATLAB function which allows the program to execute this command both quickly and easily.

Figure 1: Image 1 import with greyscale conversion

Once the image has been imported and converted to greyscale the function to ‘detectSURFFeatures’ is used to find features in the image. When viewing the features against the image the size of the circle represents the scale of the feature with the corresponding output points being the centre of the feature. This is shown in figure 2.

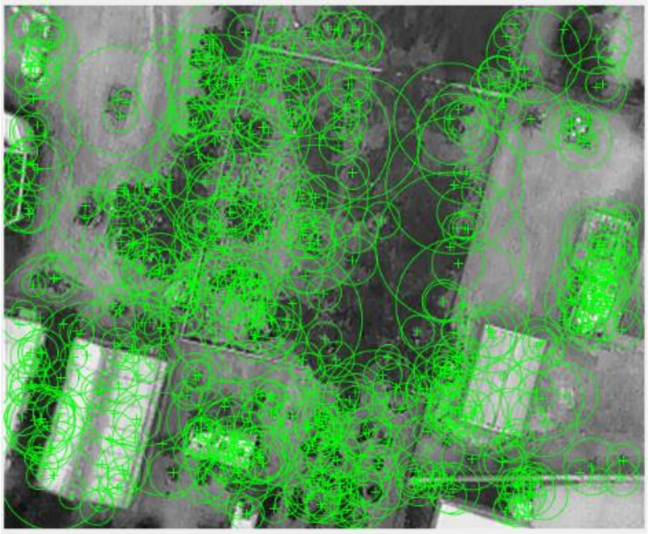

Figure 2: Image 1 SURF features detected with scale of features represented by size of circle

When the video was converted to images there were approximately 12 frames/images per second. For the results to show the full effect and ensure a visual inspection can be done the images selected will be 4 seconds apart.

Repeating the same process to create a second image variable and hence its own set of feature points. With both images having a significant number of features not present in the other image, reducing that number to only those that are in both images is essential. This uses a function that inspects the matrix of features for each image and compiles a matrix with the points that are in both images.

In figures 3 and 4 a visual test can be done to ensure that the SURF function is extracting the same features by inspecting the location and scale of features in each image. The reduction in features stands out as a possible problem with too few overlapping feature the program will not be able to orientate the images to the correct aspect. It is also noticed that the number of trees used as features may present additional problems. While growth can be disregarded as it would not be noticeable between consecutive images. Tree similarities and movement due to winds however can result in mismatched features or fail to match. Mismatching would completely distort the aspect of one image when overlapping the other, and in images where minimal features exist other that trees, a failure to match may result in no matches and hence cause the algorithm to fail at that point.

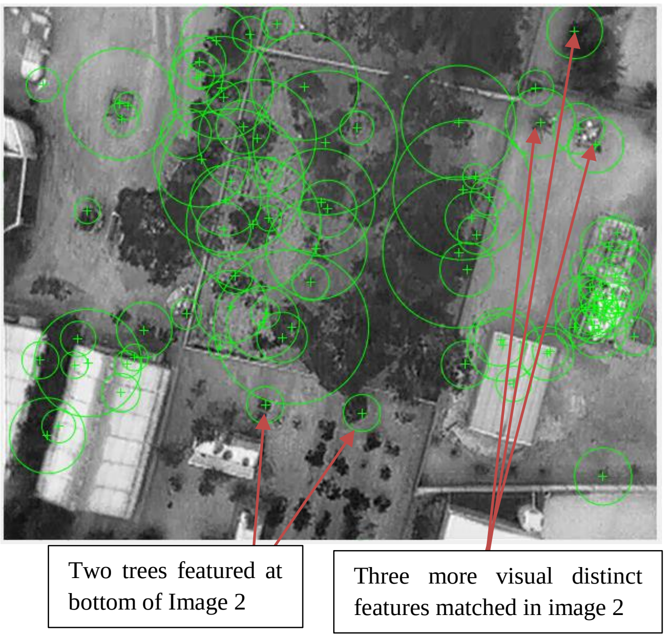

Figure 3: Image 1 Showing only features also in image 2

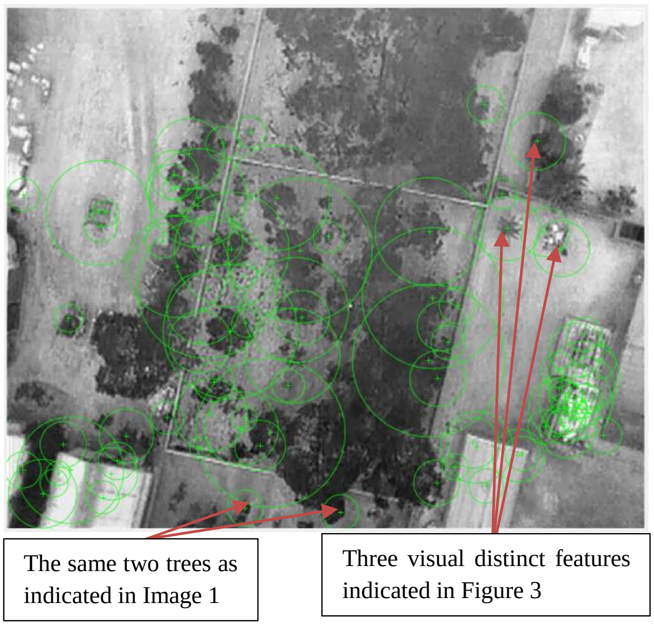

Figure 4: Image 2 Showing only features also in image 1

Now that the two images have features that match and the exact location within the image of these points is know the two images can now be combined. To do this the images must undergo further analysis to find the areas of the images that will be overlapping. MATLAB provides a function ‘estimateGeometricTransform’ that determines any changes needed to the orientation.

Once this is complete a new image is created that will hold the relevant parts of the two images and then the images are imported into this new blank image.

Results below in Figure 5 show the program is able to successful join the two test images with a high level of accuracy.

Figure 5: Images in figures 3 & 4 combined

Extending the data set to include a significantly larger area to analyse is done by following the same process while looping for each image. The algorithm is able to stitch together a larger set of images however a major issue is evident. This problem is multiple images are overlapping incorrectly as seen at the top of Figure 6. The possible reasons for this are incorrect feature detection or possible incorrect geometric transforms.

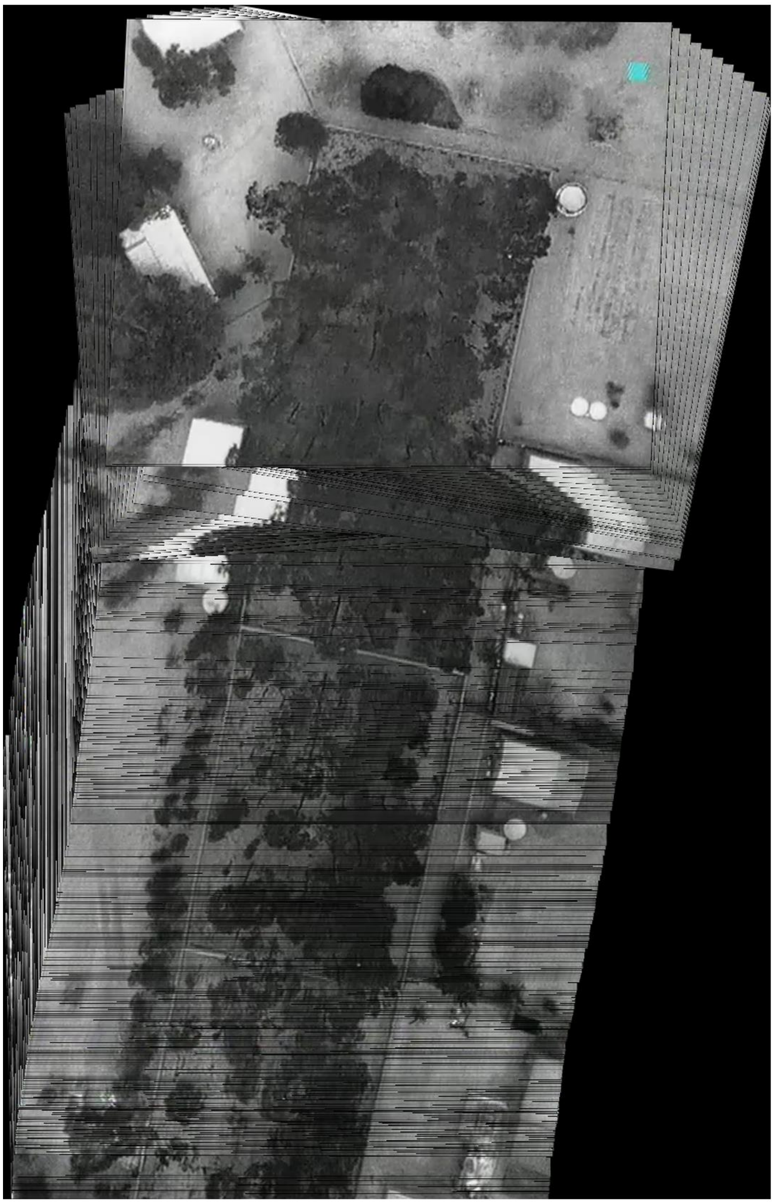

Figure 6: 50 image data set combined, showing clear signs of distortion

Expanding the data set to include all images produces the results similar to those in Figure 6. Investigation into the results of figure 7 show that in addition to the incorrect overlapping; there is also added discontinuation of images being experienced.

Figure 7: Full data set showing incorrect overlapping between the last 11 images.

6. Conclusion

The use of thermal equipment as part of the Koala Rescue research project provides a clear benefit for animal identification via UAV. Unmanned Aerial Vehicles present cost effective, non-intrusive solution for detecting animals over large areas. Additionally, the use of thermal imaging cameras provides the ability to detect animals that would otherwise be concealed to visible light cameras. Converting the thermal imaging video into a single stitched image has delivered significant challenges. As a result, the current working version requires some manual input and data preparation. Primarily discarding images with static, glare, different lighting conditions and those at angles to the ground.

References

[1] L. F. Gonzalez, G. Montes, E. Puig, S. Johnson, K. Mengersen and K. J. Gaston, "Unmanned Aerial Vehicles (UAVs) and artificial intelligence revolutionizing wildlife monitoring and conservation," *Sensors*, vol. 97, no. 16, pp. 1-18, 2016.

[2] D.-S. Lee, L. F. Gonzalez, K. Srinivas, D. Auld and K. C. Wong, "Multi-Objective and Multidisciplinary Design and Optimisation of Blended Wing Body UAV via Advanced Evolutionary Algorithms," in *In 2nd Australasian*

Unmanned Air Vehicles Conference, Melbourne, Australia, 2007.

[3] D.-S. Lee, J. Periaux and L. F. Gonzalez, “UAS mission path planning system (MPPS) using hybrid-game coupled to multi-objective optimiser,” *Journal of Dynamic Systems, Measurement, and Control,* vol. 4, no. 132, pp. 1-10, 2009.

[4] L. F. Gonzalez, “Robust evolutionary methods for multi-objective and multidisciplinary design optimization in aeronautics,” Ph.D dissertation, School Aerospace, Mechanic. Mechatronics Engineering, University of Sydney, Sydney, Australia, 2005.

[5] J. A. Malaver Rojas, N. Motta, L. F. Gonzalez, P. Corke and A. Depari, “Towards the development of a gas sensor system for monitoring pollutant gases in the low troposphere using small unmanned aerial vehicles.,” Sydney University, In Smith, Ryan N. (Ed.) Workshop on Robotics for Environmental Monitoring , Sydney, NSW, Australia, 2012.

[6] A. Yol, B. Delabarre, A. Dame, J.-E. Dartois and E. Marchand, “Vision-based absolute localization for unmanned aerial vehicles,” in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, 2014.

[7] D. Jia and J. Vagners, “Parallel Evolutionary Algorithms for UAV Path Planning,” in *Proceeding of the AIAA First Intelligent Systems Technical Conference*, Chicago, IL, USA, 2004.

[8] R. Austin, Unmanned aircraft systems: UAVs design, development and deployment, Chichester, Reston, Va, USA: American Institute of Aeronautics and Astronautics, Wiley, John Wiley & Sons Inc, Wiley-Blackwell, 2010.

[9] M. Margarita, Á. B. Jose, A. Pelayo, V. Joaquín and J. N. Juan, “Unmanned Aircraft Systems complement biologging in spatial ecology studies,” *Ecology and Evolution,* vol. 5, no. 21, pp. 4808-4818, 2015.

[10] H. Tang and Z.-L. Li, Quantitative Remote Sensing in Thermal Infrared [electronic resource] : Theory and Applications, Springer, 2013.

[11] Pix4D, “products,” 27 03 2015. [Online]. Available: https://pix4d.com/products/.

[12] Agisoft, “n/a,” 27 03 2015. [Online]. Available: http://www.agisoft.com/.

[13] D. Li, J. Shan and J. Gong, Geospatial technology for Earth observation, New York: Springer, 2009.

[14] C. Kuenzer and S. Dech, Thermal Infrared Remote Sensing [electronic resource] : Sensors, Methods, Applications, Springer, 2013.

[15] P. Wolf, B. DeWitt and B. Wilkinson, Elements of Photogrammetry with Application in GIS, McGraw-Hill Education, 2014.