ID11- SIMPLIFIED CREATION OF PHOTO-MOSAICS FROM AERIAL IMAGES

FERNANDO MARTÍN-RODRÍGUEZ[1], ORENTINO MOJÓN-OJEA[1]

1 atlanTTic research center for Telecommunication Technologies, University of Vigo. Campus Lagoas Marcosende S/N, 36310 Vigo, Spain
Ph: +34-986-812151, +34-986-812161, fmartin@tsc.uvigo.es

Abstract
This communication is about a simplified method to build image mosaics from aerial images. Getting very exact, orthonormally projected mosaics (orthomosaics) from aerial imagery is nowadays a standard technique performed with the so called "Photogrammetry" [1,7] applications.
Nevertheless, photogrammetry applications are normally expensive proprietary software (despite some open-source options of less quality and difficult to use), creating orthomosaics is normally a slow process. In this paper, we try to develop an easy method able to create a simplified mosaic useful for some applications (mainly computer vision applications where very exact geometry measurements are not necessary). Furthermore, our method is much faster than classic photogrammetry as it is based on algorithms used to synthesize panoramic images [2]: control points are automatically derived using SURF [3,4] algorithm and then used to compute geometric transformations [5] so that warped images coincide in the overlapping part. Images are warped and overlapped to form a whole mosaic.

Keywords
Drone, orthomosaic, beach, SURF algorithm.

IMAGE CAPTURE
We have built our own capture system based on a "Raspberry PI" minicomputer [9]. This platform supports the connection of one or more cameras which can be equipped with filters to select different frequency bands. Housing was built in 3D printer. Images will be captured periodically, with a small application (python+OpenCV) [10,11]. With proper relationship between firing rate and drone speed, we can set the desired overlap between photographs.

Fig 1. Payload mounted on a DJI (Phantom IV) quadcopter.
Housing can bear one or two cameras.

ORTHOMOSAIC CREATION
Method is inspired by methods used for synthesis of panoramic images from individual images. This type of algorithms finds the similarities between consecutive images to derive geometric transformations capable of matching the overlapping part. Doing so, in principle, it is possible to obtain mosaics for straight flight sections only. Proposed method uses SURF algorithm (Speeded Up RobustFeatures) to find common points between consecutive images, see figure 2.

II. ORTHOMOSAIC CREATION
Method is inspired by methods used for synthesis of panoramic images from individual images. This type of algorithms finds the similarities between consecutive images to derive geometric transformations capable of matching the overlapping part. Doing so, in principle, it is possible to obtain mosaics for straight flight sections only. Proposed method uses SURF algorithm (Speeded Up RobustFeatures) to find common points between consecutive images, see figure 2.

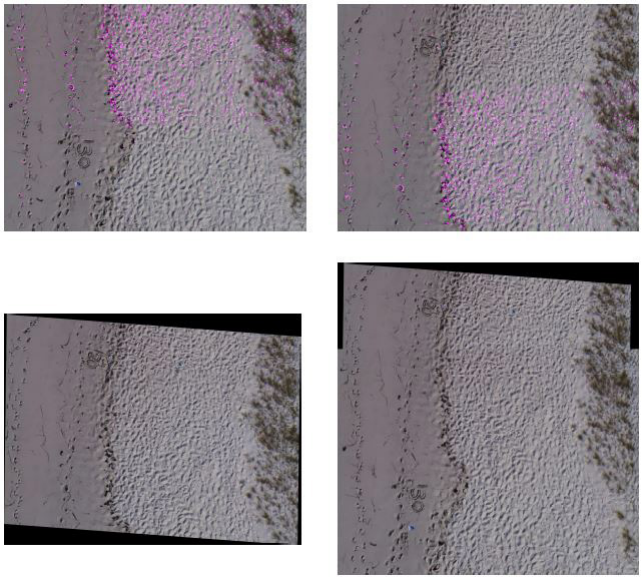

Fig2. Above: individual images with their control points.
Below: second image after applying the transformation and mosaic of the two images.

II. ORTHOMOSAIC CREATION
Method is inspired by methods used for synthesis of panoramic images from individual images. This type of algorithms finds the similarities between consecutive images to derive geometric transformations capable of matching the overlapping part. Doing so, in principle, it is possible to obtain mosaics for straight flight sections only. Proposed method uses SURF algorithm (Speeded Up RobustFeatures) to find common points between consecutive images, see figure 2.

A. Color Equalization.
When the images correspond to shots in automatic or semi-automatic mode (if camera optimizes some parameter in each shot), we can have non-uniform lighting and/or color conditions that will affect the result. Therefore, it is desirable to be able to standardize the color characteristics of the images and we have designed a preprocessing to perform this task. Process is based on computing the average color histogram of all images and transforming them to fit such average [14].

B. Execution times
Times are important in this work since the real objective is to convert hours into minutes. Testing an implementation in matlab [15] on an Intel computer (I7-6700HQ 2.60 GHz, 8Gb RAM, Windows 10) we have the following results, see Table I.

TABLE I
EXECUTION TIMES

Series:	Nº images.	Time (s)	Time/image (s)
1	13	341	26
2	18	568	32
3	27	1789	66

See that time/image is not constant. This can be justified because system operates with bigger images at each processing stage. Color equalization uses about 4 seconds per image. These results would be improved using a more efficient (compiled) programming language like C++.

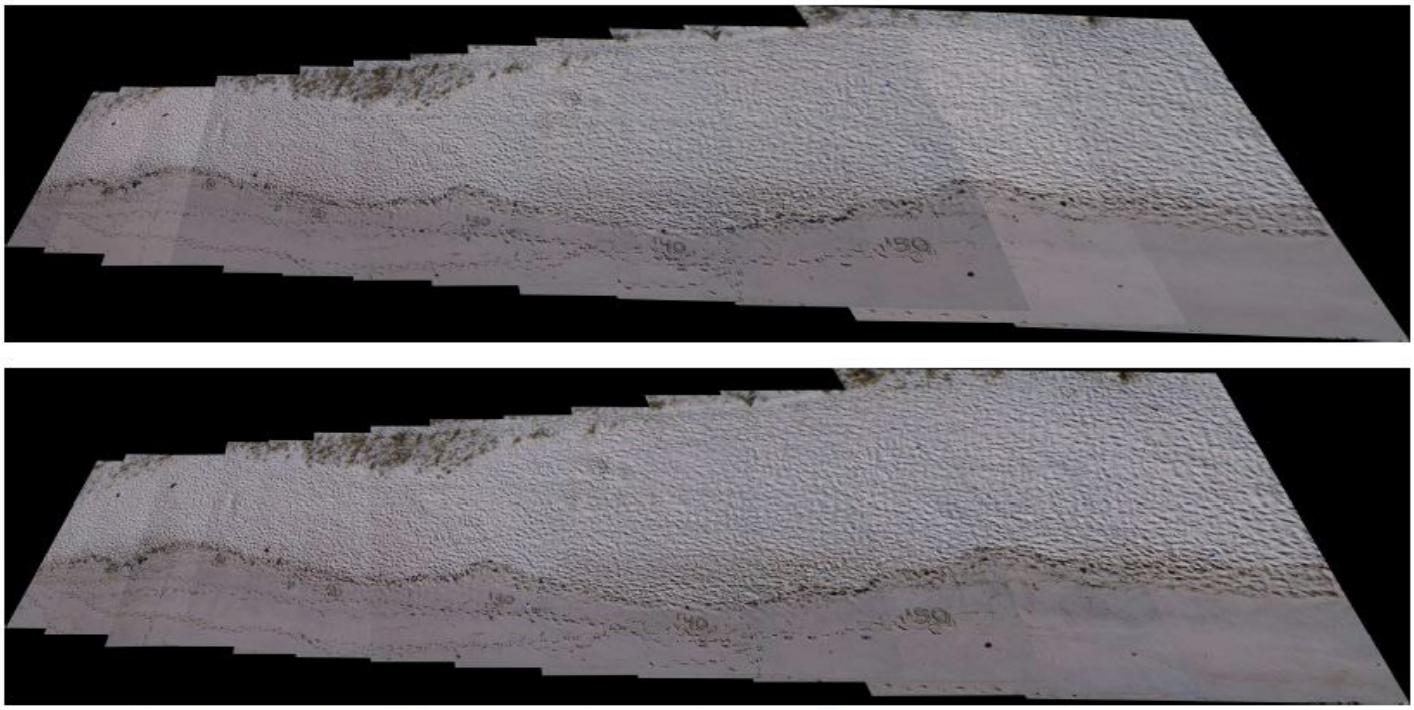

Fig3. Above: without color equalization. Below: with color equalization. 13 images, ~15m flight height.

CONCLUSIONS

A simple and economical (but versatile) capture system has been developed. We have also developed software capable of merging quickly an unlimited number of images, provided that they belong to an approximately rectilinear flight segment. As possible future lines, we can point out trying new transformations and merging several parallel rectilinear sections.

ACKNOWLEDGEMENTS

Authors wish to thank all personnel in atlanTTic who collaborated in this work. We also thank the "Parque Nacional de las Islas Atlánticas de Galicia" for allowing our flights in the beaches of the Cíes islands. We also thank European Commission for funding first stages of this work through the action "EASME/EMFF/2016/1.2.1.4" and spanish ministry for "Transición Ecológica y el Reto Demográfico" for continuing funding through program "Pleamar 2019" (project BEWATS).

REFERENCES

P.R Wolf , B.A. Dewitt, B.E. Wilkinson, Elements of Photogrammetry with Application in GIS, Cuarta edición, McGraw-Hill Education, 2014.
[2] K. Seongdo et al, "Generating panorama image by synthesizing multiple homography", 19th IEEE International Conference on Image Processing, 2012.
[3] H. Bay, T. Tuytelaars, L. Van Gool, "Speeded Up Robust Features" (SURF), ETH Zurich, Katholieke Universiteit Leuven.
[4] S.R. Panchal, S.K. Shah, P M Panchal, "A Comparison of SIFT and SURF", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 1, Issue 2, April 2013
[5]http://alumni.media.mit.edu/~maov/classes/comp_photo_vision08f/lect/08_image_warps.pdf
[6] F. Martín-Rodríguez et al, "LitterDrone: monitorización de basura marina empleando drones y análisis de imagen" Proceedings de URSI-2019, Granada, septiembre 2018.
[7] https://www.photomodeler.com/
[8] https://www.micasense.com/es/rededge-mx
[9] https://www.raspberrypi.org/
[10] https://www.python.org/
[11] https://opencv.org/
[12] L. Gottesfeld Brown, "A survey of image registration techniques", Journal ACM Computing Surveys (CSUR), Volume 24 Issue 4, Dec. 1992, Pages 325-376, ACM New York, NY, USA.
[12] V. Muto Foresi, Curso de Métodos Numéricos, Universidad del País Vasco (http://www.ehu.eus/~mepmufov/html/docencia-mnesp.html).
[14] A.K. Jain.Fundamentals of Digital Image Processing, Pearson Educación, 1999.
[15] https://es.mathworks.com/