International Journal of Robotics and Mechatronics

Image Mosaicing Using Cylindrical Mapping

Santosh D.R[†], Hemanth Kumar[§], Bharath.M.K[§], and S.M.Vaitheeswaran[§]
[†]Reva ITM, Bangalore, Karnataka, India
[§]National Aerospace Laboratories, CSIR, Bangalore, Karnataka, India

***Abstract*—**The objective of this paper is to provide extended field of view for aerial surveillance in MAV's. Aerial surveillance on unmanned flying vehicles using a single monocular camera is a trade-off between the level of detail obtained at low altitudes and degree of coverage obtained by flying the vehicle at high altitudes. The effective footprint of the camera is smaller if the aerial vehicle flies at higher altitudes. On the other hand, the region can be quickly searched from a higher altitude. The demand for high detail and wider coverage can be satisfied using Image Mosaicing i.e., the stitching of selected frames of a video by estimating the camera motion between the frames and thereby registering successive frames of the video to arrive at the mosaic.

***Keywords*—**Field of view, cylindrical mapping, image acquisition, SURF, image stitching.

I. INTRODUCTION

COVERAGE area of the camera is usually less as compared to the area under surveillance. In the recent days, to cover a wide area, more cameras are used. This creates a mesh of cameras that is to be controlled. In surveillance systems, many monitors are used to view a wide area where a specific approach of image processing is required to overcome this problem. The surveillance using an aerial vehicle is a crucial work where the UAVs are used to move to the set points for once and need to cover a wide area of space. If a single camera is used, a wide area cannot be covered, and if two cameras are used then it requires more space and hardware for acquiring and transmission. Thus there is a requirement of a method of implementing a method which overcomes the above mentioned problems.

In this paper, a method for image combining is proposed. To solve the wide area coverage using single camera, different techniques has been proposed for video mosaicing. One approach involves image collection using a mini-gimbal, an actuated platform aiming a camera, which allows the MAV in the form of a quad-copter carrying the gimbal to return multiple viewpoints without varying its course. The video data from these sensors may be fused together to yield a higher resolution panoramic video. An approach to improving the quality of panoramic images can be through optics and hardware design.

II. IMAGE ACQUISITION

For any image processing concept, the image acquisition is the initial step. Here, the images are acquired by the single camera using a gimbal setup. For the acquisition of the images, initially the camera is set at a desired position and the image of this position is captured. After capturing the image, the camera is made to rotate by an angle to move to another desired view and the image of that position is captured. The rotation of the camera by a gimbal is a tough task that has to be taken care. The gimbal rotation is done in such a manner that the image acquisition should be in synchronization with the image stitching process. In this paper, the synchronization of the image acquisition with respect to the image stitching is done by controlling the servo of the gimbal with the Arduino microcontroller by sending the required PWM signals. The delay in the image stitching is compensated by providing the delay for the gimbal after acquisition of the images till the completion of stitching process.

III. CYLINDRICAL IMAGE MODEL

Homographies is used to align images, it is done by by first warping the images into cylindrical coordinates and then apply pure translational model to align them. This only works if the images are all taken with a level camera or with a known tilt angle. Assuming the camera is in its canonical position, i.e., its rotation matrix is the identity, and the optic axis is aligned with the z axis, and the y axis is vertical. The 3D ray corresponding to an (X, Y, Z) pixel is therefore $(\hat{x}, \hat{y}, \hat{z})$. This image is mapped onto a cylindrical surface of unit radius. Points on this surface are parameterized by angle θ and height h, with the 3D cylindrical coordinates corresponding to (θ, h) given by

$$\theta = \frac{(x - x_c)}{f} \tag{1}$$

$$h = \frac{(y - y_c)}{f} \tag{2}$$

where x and y are input rectangular image pixels; x_c and y_c are center pixel; f is focal length of camera.

The 3D point (X, Y, Z) is mapped to $(\hat{x}, \hat{y}, \hat{z})$ is given by

$$(\hat{x}, \hat{y}, \hat{z}) = \frac{1}{\sqrt{X^2 + Z^2}}(X, Y, Z) \tag{3}$$

Corresponding author: S.M. Vaitheeswaran (e-mail: smvaith@nal.res.in).
This paper was submitted on January 22, 2014; revised on May 06, 2014; and accepted on October 19, 2014.

From this correspondence, we can compute the formula for the cylindrical coordinates given by

$$(\sin\theta, h, \cos\theta) = (\hat{x}, \hat{y}, \hat{z}) \tag{4}$$

Now convert the image to cylindrical coordinates given by

$$(\tilde{x}, \tilde{y}) = (f\theta, fh) + (\tilde{x}_c, \tilde{y}_c) \tag{5}$$

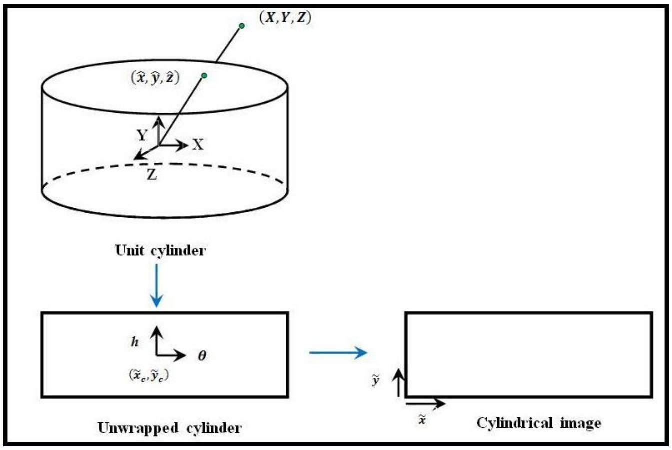

Figure 1 Cylindrical wrapping

For inverse Cylindrical projection, which is to map an image to a rectangular view can be obtained by

$$\hat{x} = \sin\theta \tag{6}$$

$$\hat{y} = h \tag{7}$$

$$\hat{z} = \cos\theta \tag{8}$$

$$x = f\frac{\hat{x}}{\hat{z}} + x_c \tag{9}$$

$$y = f\frac{\hat{y}}{\hat{z}} + y_c \tag{10}$$

IV. FEATURE MATCHING

Feature matching of the images acquired is done by using a Speeded up Robust Feature [SURF] Extraction method. SURF is a local feature recently presented in [1]. It is based on the Hessian matrix for both selecting the location and the scale in image feature description given by

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \tag{11}$$

Given a point X = (x, y) in an image [1]-[3],

$H(x,\sigma)$ is Hessian matrix in $\boldsymbol{X}$ at scale σ. $L_{xx}(x,\sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\sigma)$ with the image at point x, and similarly to $L_{xy}(x,\sigma)$ and $L_{yy}(x,\sigma)$.

An important characteristic of SURF is the fast extraction process and the fast matching speed it permits, mainly achieved by a single step added to the indexing based on the sign of the Laplacian of the interest point [1].

For the extraction of the SURF descriptor [1]-[3], the first step consists of constructing a square region centered on the interest point, and oriented along the selected orientation. The region is split up regularly into smaller 4×4 square sub-regions. For each sub-region, *dx* and *dy* features at 5×5 regularly spaced sample points are computed.

Here, *dx* and *dy* denote the Haar wavelet response in horizontal direction and vertical direction respectively. To increase the robustness towards geometric deformations and localization errors, the responses *dx* and *dy* are first weighted with a Gaussian centered at the interest point.

Then, the wavelet responses *dx* and *dy* are summed up over each sub-region and form a first set of entries to the feature vector. In order to bring in information about the polarity of the intensity changes, the absolute values |*dx*| and |*dy*| are summed in order to obtain information about the polarity of the image intensity changes. Hence, each sub-region has a four dimensional descriptor vector *v* for its underlying intensity structure

$$v = \left(\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|\right) \tag{12}$$

The resulting SURF descriptor vector for all 4×4 sub-regions is of length 64.

V. IMAGE STITCHING

To obtain the wide area, the image stitching is required for combining the images that are acquired. As the images are acquired, the images are subjected to a feature detection process using SURF [1]-[3]. Using these features, the good features are extracted by considering the points which are greater than three times the minimum distance point. These good features, the images are stitched [4]-[6]. In this stitching process, the first image is retained as it is. The second image is altered by considering the features and the portion of second image is removed which has the feature points less than three time the minimum distance point. The remaining part is wrapped using a perspective transformation. Perspective transformation is similar to affine transformation but it is done by using a 3×3 matrix. The simplest way to define this transform is by setting the source image (second image) points to three corners, for example, the upper and lower left together with the upper right of the source image. The mapping from the source to destination image is then entirely defined by specifying destination points, the locations to which these three points will be mapped in that destination image. Once the mapping of these three independent corners (which, in effect, specify a "representative" parallelogram) is established, all the other points can be warped accordingly.

The perspective projection maps points in the three-dimensional physical world onto points in the two-dimensional image plane along a set of projection lines that all meet at a single point called the center of projection [7],[8]. The transformed image is wrapped in such a way that the result image should be as if it is a single image.

VI. EXPERIMENTAL RESULTS

Initially, the images are been experimented by accessing the images and generating the cylindrical projected images in MATLAB simulation environment. After the simulation, the process of real time is executed. In our system, a gimbal setup is kept on a UGV for testing purpose. In this system, a camera is mounted on the gimbal. An Arduino UNO microcontroller is fixed on the vehicle to provide the necessary rotation for the gimbal. **Figure 2** shows the gimbal setup with servo for the rotation of the gimbal.

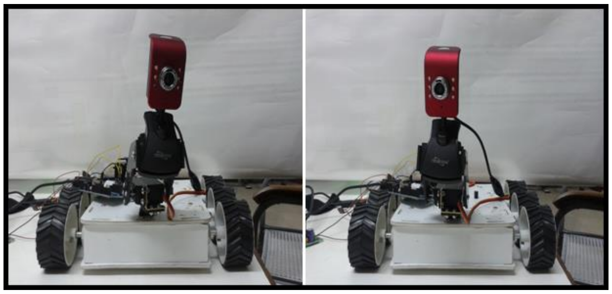

Figure 2 Experimental setup

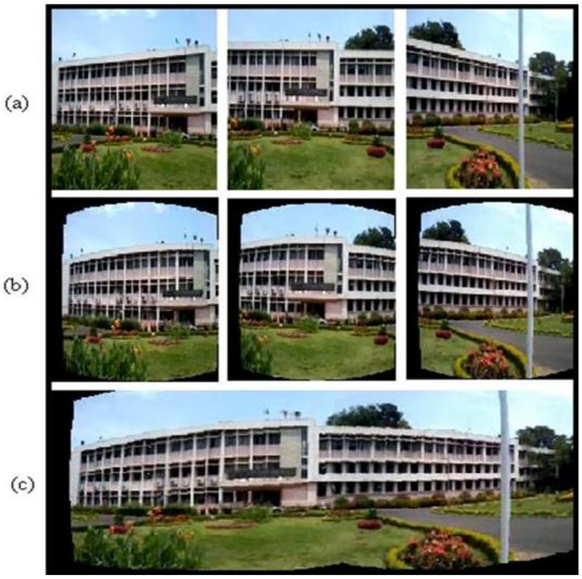

Figure 3 MATLAB simulation results

Figure 4 Real time execution results

As the images are acquired from the camera, the images are processed in OpenCV on Ubuntu platform. **Figure 3** shows the Matlab simulation of the images acquired. Here, three images are captured from a camera and processed offline to generate a panoramic image. **Figure 4** shows the real time executed images. The servo rotates by an angle of 300 of rotation for each frame and two frames, one at a certain angle and another frame with a deviation of 300 is used to obtain a panoramic view in real time.

VII. CONCLUSION AND FUTURE WORK

This paper describes the generation of a panoramic view of a system comprising a gimbaled camera with overlapping fields of view to generate the rectangular as well as cylindrical views on board a quad-copter system in a simulation environment and also experimentally using a ground robot. By using both synthetic and real data, the accuracy in the motion estimation process is verified.

Experiments reveal that cylindrical mapping is much faster when compared to rectangular mapping, as few images are only required for cylindrical mapping compared to rectangular. This reduces computation load and increases the performance of the system in real time.

The future work that can be developed is increasing the speed of the execution process by reducing the delay and implementing for further increase in the field of view. This has a wide application in the field of military and commercial areas.

ACKNOWLEDGMENT

The authors SMV, MKB and HE, NAL gratefully acknowledge the funding received from AR&DB, GoI under the NPMICAV program to carry out the above work. The work is carried out at National Aerospace Laboratories, CSIR, Bengaluru, Karnataka, India.

REFERENCES

[1] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features”. In The 9th European Conference on Computer Vision, 2006.

[2] Zhao, Jin, Sichao Zhu, and Xinming Huang, “Real-Time Traffic Sign Detection Using SURF Features on FPGA.”

[3] Zhou, Zhiping, and Haibin Yang, “A Fast Video Forensics Method Based on Face Detection”, 2013.

[4] Botterill, Tom, Steven Mills, and Richard Green, “Real-time aerial image mosaicking”, Image and Vision Computing New Zealand (IVCNZ) 2010, 25th International Conference of. IEEE, 2010.

[5] Steedly, Drew, Chris Pal, and Richard Szeliski, “Efficiently registering video into panoramic mosaics”, Computer Vision, 2005, ICCV 2005, Tenth IEEE International Conference on. Vol. 2. IEEE, 2005.

[6] Ning, Yufang, Ren Chen, and Pengfei Xu, “Wide baseline image mosaicing by integrating MSER and Hessian-Affine”, Image and Signal Processing (CISP) 2011, 4th International Congress on. Vol. 4. IEEE, 2011.

[7] Peter J. Burt, Edward H. Adelson, “A Multi resolution Spline with Application to Image Mosaics”. ACM Trans. Graph. (TOG 1983) 2(4):217-236.

[8] Saravanan, P., C. K. Narayanan, P. V. S. S. Prakash, and GV Prabhakara Rao, “Techniques for Video Mosaicing” In WEC (5), pp. 286-289. 2005.