IJCTA, 8(3), 2015, pp. 933-940

DSP Based Velocity Sensor for Planetary Lander

Tanisha Bhatia, Jayanta Laha, Arindam Mal, Subhalakshmi .K*

Abstract: In Planetary landing missions, it is important to navigate the lander accurately to have successful and safe landing. Precise navigation of Planetary Lander is possible using Vision camera and Image processing techniques. Image processing involves large computational capability and fast processing which can be accomplished using high speed Digital Signal Processors (DSP). This paper presents a Vision sensor for computing the relative horizontal velocity of the lander using phase correlation techniques. The performance is demonstrated using a DSP kit, Actel FPGA and Active Pixel Sensor detector. It is tested in laboratory environment using scene simulator.

***Keywords**:* DSP; FPGA; APS; Phase correlation; Fourier shift; LRO

1. INTRODUCTION

Two aspects of spacecraft navigation are: knowledge and prediction of spacecraft's position and velocity. When a spacecraft leave its orbit and enters the descent phase it not only has vertical velocity but also horizontal velocity. Terrain relative navigation can be used for accurate landing of the spacecraft. It is essential to have precise estimation of the position and velocity during the various phases of landing like rough breaking phase, precision breaking phase, retargeting phase, hovering phase, final decent phase and so on. Precise spacecraft navigation results in safe landing thus avoiding craters, boulders and other hazards. Vision based navigation involves the execution of high end image processing algorithm in real time. In this paper, the design and implementation of a vision based in-plane relative horizontal velocity sensor is described. The results of the tests conducted in lab are also presented.

Different algorithms are used for finding out the relative shift. Feature based velocity estimation is one of the conventional methods where velocity is estimated by matching features of subsequent images [2]. Same features of the subsequent frame of images are correlated to evaluate the position [3]. A distinct approach is the utilization of Phase information of images [1]. This paper describes Phase correlation method i.e., correlation in phase domain which is independent of feature extraction .The peak in correlation function is likely to be detected more accurately using phase correlation method than the classical cross correlation[2]. Phase correlation gives an acute peak at the registration point while cross correlation results in broad peaks and main peak is often not centered at the right point. The phase correlation method gives better accuracy with a stable output compared to other methods(feature based) in presence of noise, illumination variation etc. So it is suitable to use such system for interplanetary landing.

The organization of this paper is as follows: Section II gives the overall overview of the sensor. Section III describes the Phase correlation method used for finding the pixel displacement. Section IV gives the hardware details of the system. The simulation results are given in section V and section VI contains the conclusion of the paper.

Scientists at Laboratory for Electro-Optics Systems, Indian Space Research organization, Bangalore -560058, India

* e-mail: varun.aduru@gmail.com

2. SYSTEM OVERVIEW

The sensor comprises of the collecting optics, the detector and the processing unit [Fig.1.b]. It will be mounted on Lander in such a way that it views the planetary terrain vertically and captures images continuously. The sensor can operate up to maximum velocity of 200m/s at 7km. Fig.1.a. shows the sensor, the lander trajectory and the images of the terrain. Two images will be taken at fixed time interval and processing of these images is done to calculate the pixel shift / displacement in x & y axes. Knowing the altitude, the average velocity of the lander is derived from this as pixel shift per unit time. Details of sensor are given in table 1.

This information is used by the Navigation and Guidance control system (NGC). The camera interface to NGC is via MIL-STD-1553B.

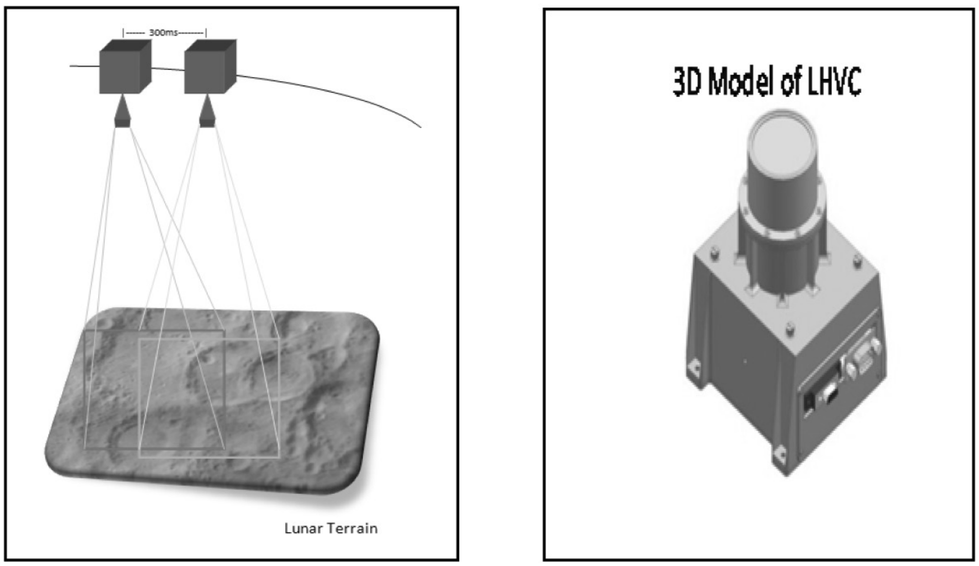

Figure 1a: Sensor capturing images of the terrain 1.b 3D model of Camera

3. PHASE CORRELATION METHOD

Image acquisition of the planet by the camera starts at the altitude of 7 km till descent. Images are acquired at fixed time intervals and processed.

Table 1
Details of the Sensor

Sl. No.	*Parameter*	*Value*
1.	$Velocity_{max}$	200m/s
2.	Operating range	7km to 100m
3.	Update rate	1Hz
4.	Mass	800gm
5.	Power	2W(avg)
6.	Volume	110x110x150
7.	Detector(1024X1024)	Star 1000
8.	Pixel size	15u X 15u
9.	Focal length	90mm
10.	F number	6
11.	Field of view	10^0

Here, phase correlation method is used to compare two consecutive images for calculating the pixel shift in x and y direction.

Using a high end Digital Signal Processor, the images are processed and will give shift in two directions. The NGC uses altimeter data and computes velocity as given in eq. 1.

$$\Delta V = (\text{Pixel resolution at Z height} * \text{pixel shift})/\Delta T \quad (1)$$

Where,

Z = Height given by Altimeter
ΔT = Time interval between two images acquisition.
Pixel shift = Shift given by the sensor.

3.1 PHASE CORRELATION

Phase correlation is one of the image registration methods, in which Fourier shift property is the underlying principle. It uses frequency-domain approach to estimate the pixel shift between two consecutive images. Consider i_A and i_B are the two images captured from the camera. Both images are converted to frequency domain using 2D discrete Fourier transform. I_A and I_B are the FFT results of image i_A and i_B correspondingly. The complex conjugate of I_B is multiplied by I_A element wise and normalization is carried out to give R as mentioned in eq. 2.

$$R = \frac{I_A \circ I_B^*}{|I_A \circ I_B^*|} \quad (2)$$

Where, o is the Hadamard product (entry-wise product). By applying the inverse FFT, normalized cross correlation is obtained.

$$r = \mathcal{F}^{-1}(R) \quad (3)$$

The location of the peak in r will be determined using eq. 4.

$$(\Delta x, \Delta y) = argmax_{(x,y)}\{r\} \quad (4)$$

Fig. 2 shows the flowchart for the overall phase correlation technique.

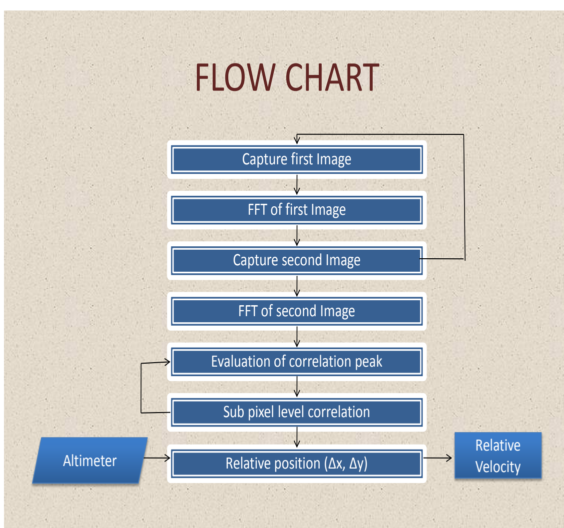

Figure 2: Flow Chart of Phase Correlation technique

4. HARDWARE IMPLEMENTATION

The algorithm is implemented using a high speed Digital Signal Processor TMS320C6748P from Texas Instruments. This is used for the development model. For imaging, a Rad-hard Active Pixel Sensor STAR1000 [Onsemi] is used. The captured images are saved in an off-chip SRAM for further processing. A Rad-hard 2 million gate FPGA [Actel RTAX 2000] generates the APS drive signals and performs all logic. The block diagram of the hardware implementation of the system is shown in Fig. 3.

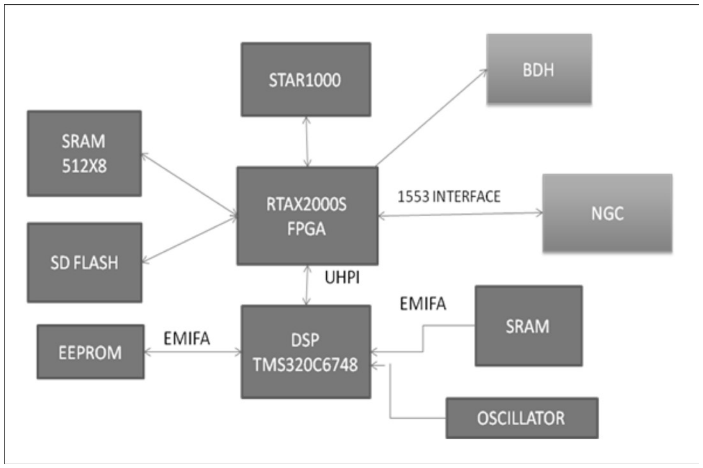

Figure 3: Block diagram of the Hardware Implementation

4.1 APS detector

The detector used is a CMOS active pixel image sensor which has 1024 by 1024 pixels on 15 µm pitch. It has an on-chip 10 bit ADC which provides direct digital output to the FPGA.

4.2 ACTEL FPGA

Actel FPGA RTAX 2000 is interfaced with the DSP. With the on-chip ADC, the pixel data of the APS is read by FPGA. Thus the digital form of representation of the scene is sent to the DSP frame by frame for processing.

Finite state machines and sequential blocks are used to generate APS drive signals from FPGA. A 24 MHz crystal provides the basic clock which is converted to 12MHz and 3MHz inside FPGA. 3 MHz clock is derived for driving and 12 MHz for operation of state machine. For the APS, line address read, line address reset and pixel address are selected onto address bus. In the address bus of the detector, loading and latching of pixel address and the clock necessary for A/D conversion are enabled by sequential block.

The detector gives one frame of image as 1024X1024 but in order to achieve higher update rate, windowing is done with 128X128 array. FPGA sends only small window of image to DSP for processing. The sensor also consists of an additional feature of storing images in SD card while landing which can be retrieved post landing.

4.3 Digital Signal Processor Design

Some of the features of the DSP TI- TMS320C6748 are listed below.

- 250 MHz; 1500 MFLOPS
  - 256 KB of L2 SRAM and 32 KB of L1 program and data RAM

- 16-bit HPI for connecting to hosts
- Two I^2 C, two SPIs, 16-bit EMIF

The algorithm is written in C language and implemented using DSP. An XDS emulator is used to fuse the C programming done in Code Composer Studio (CCS) into the hardware kit.

The clock to DSP is supplied by 24MHz crystal which is converted to 250 MHz frequency using the internal PLL of the DSP. By setting the PLL configuration register, different clock frequencies can be generated. DSP is connected with FPGA, EEPROM and SRAM and all the operations have to be done at different clock frequencies. EEPROM frequency is set at 10 MHz, SRAM at 40 MHz and 24 MHz clock for FPGA using PLL whereas the internal operations of the DSP core takes place at 250MHz. Two interfaces are used for DSP connection viz. EMIFA and HPI. Through Asynchronous External Memory Interface (EMIFA), program code stored in EEPROM is transferred to DSP at the boot time. Asynchronous EMIF is also used to interface with the SRAM.SRAM is used for both data and program memory storage.

At power on condition, DSP boots from EEPROM and it copies the instructions from EEPROM to SRAM. Later, all the instructions are carried out from SRAM. External memory is used instead of the internal memory of DSP as internal memory is vulnerable to upsets in space. The same SRAM is used for storing the image data. DSP and FPGA are connected by 32 bit bus thus achieving a faster transfer rate as FPGA can simultaneously send 4 pixel information each being 8 bit.

In CCS, the linker file is generated which is used for the allocation of memory. Using Host Port Interface (HPI), processor receives 8-bit image data from the FPGA. FPGA acts as master and DSP as a slave, data is sent to DSP in burst mode.

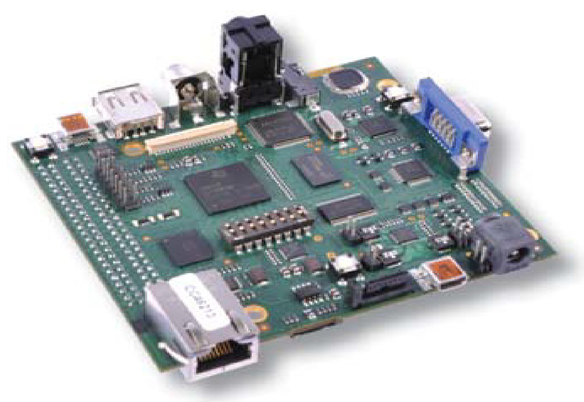

Figure 4: DSP kit used for evaluation

5. REAL TIME SIMULATIONS AND TEST RESULTS

For algorithm verification, first simulation is done in MATLAB. The input image is taken from the Lunar Reconnaissance Orbiter (LRO) images available from public LRO camera website and Chandrayaan-I images. It is used as the first image and the second image is created by giving a shift in the first image using MATLAB tools. The result obtained matches with the shift given manually, thus proving the algorithm. A STAR100 camera is interfaced to the DSP development kit containing TMS320C6748 DSP (equivalent version for space grade DSP). The real time operation is checked by simulation carried out in lab. A video is generated from the LRO images and this scene is captured by the detector. The sensor processes the images and output is displayed on console. PC connected with DSP using a USB acts as a console in our system. Fig. 5 shows the two images which are used for carrying out simulation in MATLAB.

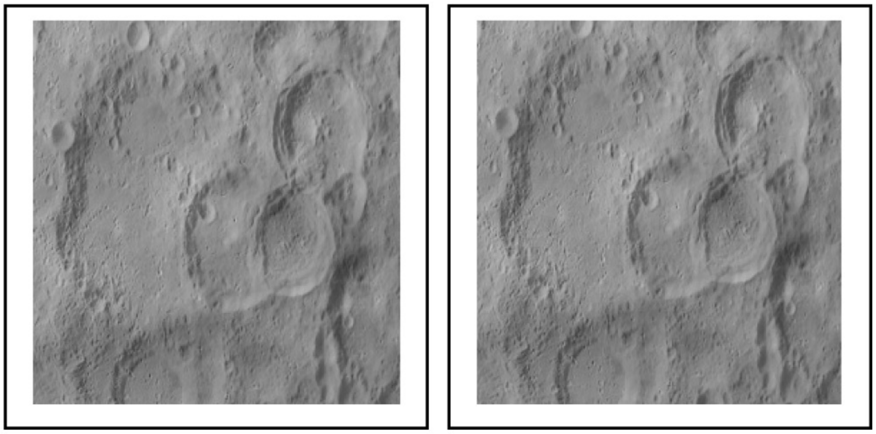

Figure 5: Images for MATLAB Simulation

Fig. 6 shows the two images captured by the sensor in real time in lab from the scene simulator. The shift in both the images can be observed by seeing the circled feature's displacement. The result obtained from the sensor matches with the shift seen by observing the images. Fig. 7 gives the FFT results of the captured images. The correlation peak obtained is encircled in Fig.8.

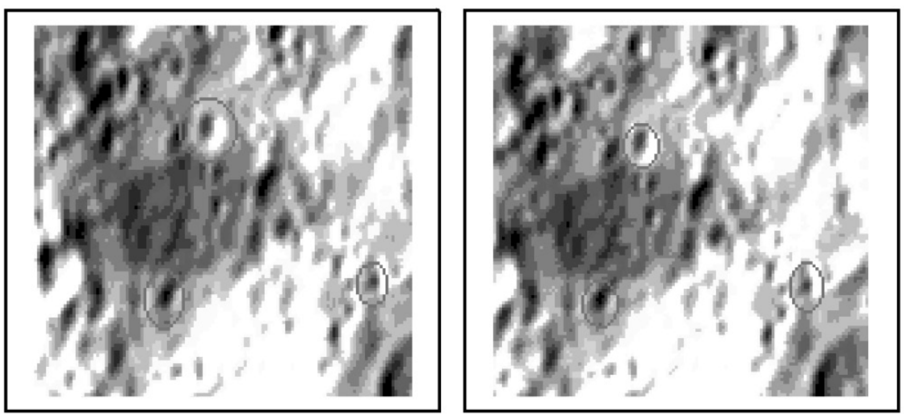

Figure 6: Images captured by sensor in real time

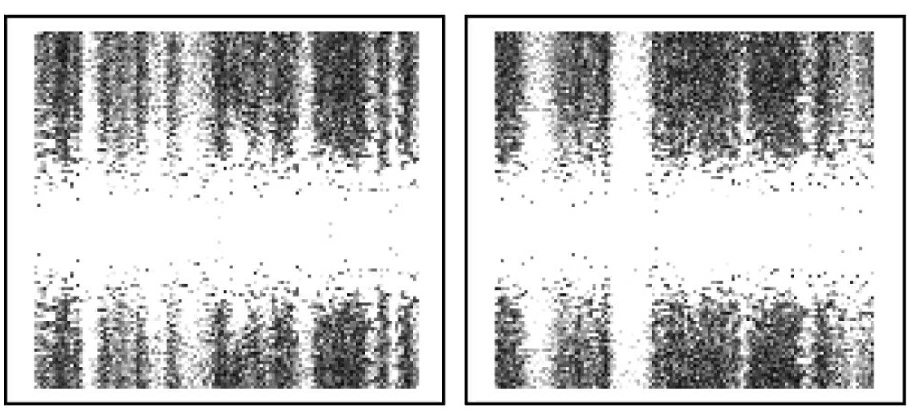

Figure 7: FFT of consecutive images

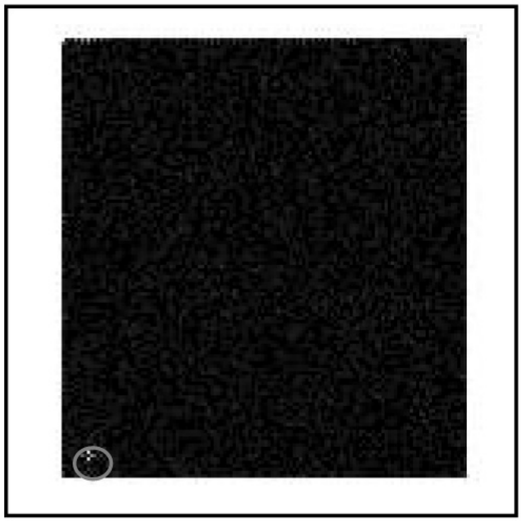

Figure 8: Correlation peak

The experimental result shows that the entire displacement estimation is done in 730ms. Time consumed for each step is as follows: ~200ms for FFT of first image, ~200ms for FFT of second image and ~330 ms for computation of correlation peak. The sensor will have an update rate of 1Hz.

6. CONCLUSION

This paper describes the theory, design and implementation of a vision sensor for computing the horizontal velocity of the Lander. The implemented algorithm of phase correlation is better than other methods in conditions with variable illumination and noise. The simulations are carried out in MATLAB and CCS and experiment is done using the developed model in lab.

Future work will involve improving the update rate of the sensor by carrying out FFT operations in Hardware. Image processing techniques like noise removal, filtering can be applied for very high speed landing spacecrafts.

ACKNOWLEDGEMENT

The authors express their thanks to Dr.G.N.Rao, Director, Laboratory for Electro-Optics Systems (LEOS)-ISRO for the valuable technical suggestions provided by him in this research work.. The authors thank Sri. A.S.Laxmi Prasad, Deputy Director, SDA, LEOS also, for his continuous encouragement and guidance.

References

[1] Hassan Foroosh (Shekarforoush), Josiane B. Zerubia, "Extension of Phase Correlation to Subpixel Registration", *IEEE Transactions on Image Processing*,Vol.11, No.3, March 2002.

[2] Weill JIAO, Yaling FANG, Guojin HE. "An integrated feature based method for sub-pixel image matching". *Center for Earth Observation and Digital Earth*, Chinese Academy of Sciences, CHINA.

[3] Arindam Mal et. al. "Vision based navigation for safe landing in interplanetary missions", *LCPM-10*, June 2013, USA.

[4] M.A. Garratt, A.J. Lambert, T. Guillemette, "FPGA Implementation of an Optic Flow Sensor using the Image Interpolation Algorithm", Australasian Conference on Robotics and Automation (ACRA), December 2-4, 2009, Sydney, Australia.

[5] Harris, C., Stephens, M., 1988. A combined corner and edge detector. In: fourth Alvey Vision Conference, Manchester, UK, pp.147-151.

[6] Andrew Johnson, Reg Willson, Yang Cheng, Jay Goguen, Chris Leger, Miguel Sanmartin and Larry Matthies, Jet Propulsion Laboratory, California Institute of Technology, CA, USA “Design Through Operation of an Image-Based Velocity Estimation System for Mars Landing”, *International Journal of Computer Vision*, 2007.

[7] Peter Beukelman, Laurens Bierens, “A Reconfigurable Architecture for Real-Time Rotation and Scale Invariant Template Matching Algorithms on Streaming Images”, *Signal Processing Sensor Fusion and Target Recognition, Proc. of SPIE* Vol.5809.

[8] Larry Matthies, Mark Maimone, Andrew Johnson, Yang Cheng, Reg Wilson, Carlos Villalpando,Steve Goldberg, and Andrews Huertas, Jet Propulsion Laboratory, California Institute of Technology, USA, “Computer Vision on Mars”, *International Journal of Computer Vision*, 2007.

[9] Zitova, B., Flusser, J., “Image registration methods:a survey”, *Image and vision computing*, 21, pp. 977-1000, 2003.

[10] Misganu Debella-Gilo, Andreas Kääb. “Sub-pixel precision image matching for measuring surface displacements on mass movements using normalized cross-correlation“. Institute of eco sciences, University of Oslo, Norway.