Range Determination Algorithm Performed on Mars Exploration Rover Stereo Images

Eric Wasatonic, Sedig Agili and Aldo Morales
Electrical Engineering Program
Penn State Harrisburg
777 West Harrisburg Pike
Middletown, PA 17057

Abstract

The electrical engineering students at Penn State Harrisburg have an ample opportunity to pursue interests in electrical and electronic circuits, including digital circuits and VLSI and its fabrication, microprocessors and their applications, electromagnetics, communications, control systems, digital signal/image processing and computer vision. They also have opportunity to demonstrate their knowledge through hands-on course projects and laboratory experiences, in the above fields. In this paper, an example of an image processing application project is developed, in the context of an image-processing course. This paper presents an algorithm that uses stereo images, obtained from two cameras mounted on the Mars Exploration Rovers, to determine the range of distant objects in the images by using correlation and triangulation. The initial value obtained by the algorithm was not accurate because it did not take into account the fact that the range of an object beyond the camera's focal point is non-linear in appearance, and to the non-linearity of the camera lens, thus the range obtained was subjected to correction to compensate for these two factors. The results obtained by the algorithm are comparable to those obtained by NASA.

I. Introduction

Penn State Harrisburg offers Bachelor of Science in EE, BS EET, and Master in Electrical Engineering degrees. The BS EE program provides an opportunity for students to pursue interests in electrical and electronic circuits, including digital circuits and VLSI and its fabrication, microprocessors and their applications, electromagnetics, communications, control systems, digital signal/image processing and computer vision. The BSEET program provides similar experience however, its strengths include: an applied, hands-on approach and extensive laboratory experience. Through a senior capstone design project, both curricula emphasize written as well as verbal communication and a teamwork approach among students to attain a common goal. Students in the DSP, communication, senior project and other courses have ample opportunities to demonstrate their acquired knowledge through course projects. In addition, the EE and EET programs at PSU Harrisburg are ABET accredited and one of the ABET requirements is to have program outcomes. Indeed, one of the electrical engineering program outcomes[1] states that *"A graduate from the Electrical Engineering program at Penn State Harrisburg will demonstrate the ability to identify, formulate, and solve engineering problems.*

- *A student will be able to classify problem types (i.e. analysis vs. design) and select appropriate solution methods.*
- *A student will be able to integrate solution methods to a problem that may include several engineering problems.*"

In this context, the identification, formulation, solution, and implementation of the project presented in this paper, was accomplished by an undergraduate student using MATLAB and knowledge built from several EE courses. This project details an algorithm that uses stereo images, obtained from two cameras mounted on the Mars Exploration Rovers, to determine the range of distant objects in the image by using correlation and triangulation.

The two very successful Mars Exploration Rovers (MER) named Spirit and Opportunity were sending data of the Mars environment back to Earth. Much of this data came in the form of images, whether they were infrared, visible, or x-ray. These images were readily available on NASA's website[2]. In addition to sending images to Earth, each rover analyzed its own images from a pair of cameras to help it navigate around obstacles[3,4]. The reason for using a pair of cameras is to determine the range of distant objects just as a person would estimate such a range simply by using his or her binocular vision.

In this paper, an algorithm is presented to analyze stereo images sent from the Spirit rover. The aim of the algorithm is to determine the range of various rocks and hills in the distance, and compare the results obtained with those from NASA. The objects analyzed here are portions of images sent from Spirit's panoramic cameras on April 2004[2]. Each camera has a 16^{o} full field of view projected onto a 1024×1024 pixel CCD array[2]. They are mounted 1.5 meters above ground level, separated horizontally by 0.3 meter, and each has a 1^{o} toe-in, which causes them both to focus on a point 8.6 meters away. There were three main parts of this algorithm: correlation between the target (object) and the original (base) images, pre-range determination, and the calculation of the correction coefficient. The purpose of the correlation algorithm is to determine exactly where the target (object) is located in each stereo image (left and right images). This location occurs at the point in the correlation image with the greatest positive magnitude. The algorithm then calculates an uncorrected range for the object relative to each camera and averages those two ranges together. However, this range is inaccurate because it does not take into account the fact that the range of an object is non-linear in appearance and also due to the non-linearity of the cameras' lenses. This range is corrected by multiplying it with a correcting coefficient. Results obtained from this stereo range algorithm successfully compares with NASA results. This paper is organized as follows, in +section II, the basics of stereo image is reviewed. The algorithm and results are presented and discussed in sections III and IV, respectively. Conclusions are given in section V.

II. Review of Stereo Imaging

Stereo (binocular) vision uses two cameras (color or gray scale) to obtain left and right images[4-9]. In these two images, the coordinates of associated pixels with a target object are compared and triangulated to determine the range and depth of the specified object in the images, just as human vision does. The basic geometry[5,9] used for obtaining range or depth information of the target object, from a stereo imaging system, is depicted in Figure 1, where L and R corresponds to the left and right camera locations, respectively, and the left camera coordinates are chosen as the 3D world reference.

There are different variations[5-7,9] from this model, but the simplest case arises when the optical axes of two cameras are parallel as shown in Figure 1, where f is the focal length of both cameras, and the baseline (that is the line connecting the two lens centers) is perpendicular to the optical axes. Once the relative position and orientation of the two cameras are known[2], then the coordinates (X,Y,Z) of a point P in the stereo images can be reconstructed from the perspective projection of P on the cameras' image planes. In this geometric model, let b be the distance between the two lens centers, XZ be the plane where the optical axes lie, and XY plane be parallel to the image plane of both cameras. The baseline is on the X axis, and the origin O of the (X,Y,Z) world reference system is the lens center of the left camera. Using this setting the equations of the stereo triangulation are:

$$Z = (b \times f) / (x_1 - x_2)$$
$$X = x_1 \times Z / f$$
$$Y = y_1 \times Z / f$$

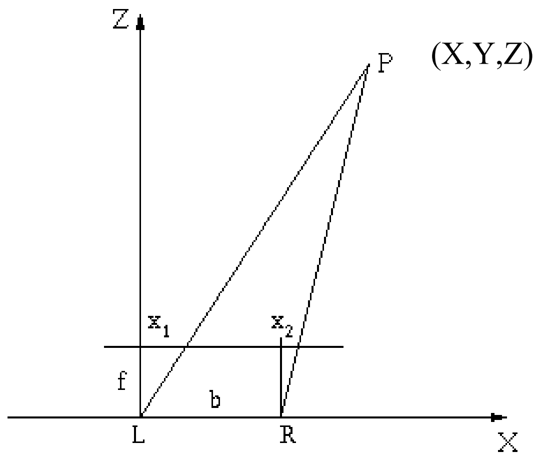

Figure 1: Basic geometry of parallel cameras[5,9]

where (x_1,y_1) and (x_2,y_2) are image coordinates of the left and right cameras, respectively. There are, however, some subtleties when applying these standard stereo vision equations. For instance, in order to calculate the coordinates (X, Y, Z) one has to know all the characteristics of the cameras used such as lens distortion, horizontal scanning and sampling frequencies. However, based on simple geometry, image correlation, and curve fitting a range determination algorithm was developed, with comparable results to those obtained by NASA.

III. Range Determination Algorithm

The geometry used in the derivation of the distance from the cameras to the object is shown in Figure 2. In the figure, the two cameras are 0.3 meters apart with a 1° toe-in (1° displacement from the vertical line), therefore 89° (1.5533 radians) from the horizontal axis. Each field of view is 16° (0.2793 radians) wide with DL and DR measured from the center of each view, and considered to be negative on the inside, and positive on the outside. Because of the 1° toe-in, the centers of the two views converge on a focal point 8.5935 meters away, at a total angle of 2° (0.0349 radians). Inside the isosceles triangle is an object being observed. It has a negative pixel displacement of –DL and –DR in the image obtained from the camera. Using an image width of W = 256; the ratio 0.2793 / W is a factor used to convert DL and DR from pixel displacement to angular displacement. In Figure 2, side a (the left pre-range) is facing angle A, angle B is the top angle between the lines pointing to the object. Side a is then taken as the range assuming that the true range (perpendicular to the 0.3 m line) is much greater than 0.3 m. Therefore, the calculation of side a can be determined using the law of sines as follows:

$$a = \frac{0.3 \sin A}{\sin B}$$

The right pre-range is obtained in a similar fashion. The final range is calculated by averaging the left and right pre-ranges. Note that this calculation ignores the fact that the cameras are 1.5 meters above the ground. Nevertheless, these calculations are good approximations and the results obtained here are comparable to those obtained by NASA.

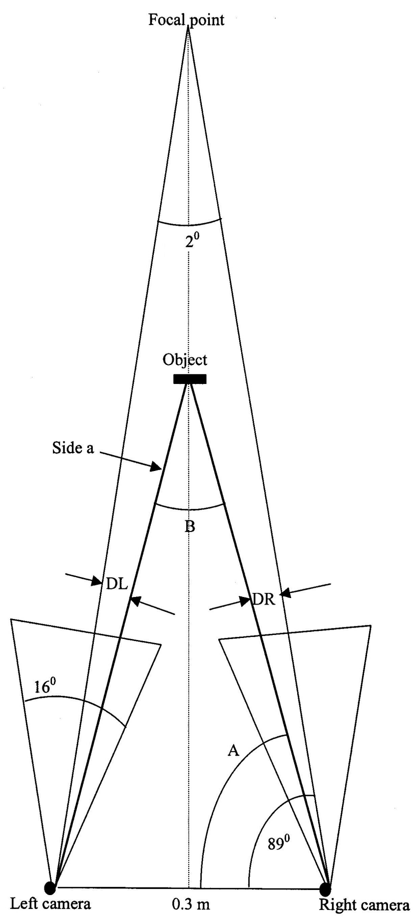

Figure 2: Geometry used to calculate the range

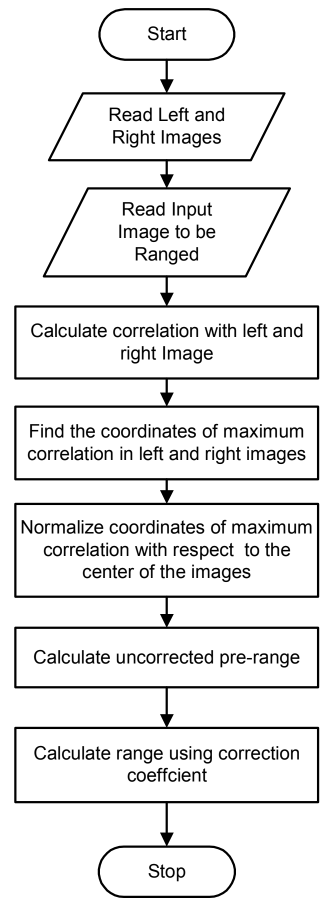

Figure 3: Algorithm flow-chart

The program flowchart for the algorithm is shown in Figure 3 and works as follows: The user inputs the two stereo images, obtained from NASA's rover website[2], and a user-defined target image to be ranged. The size of the original stereo images were 1024×1024 pixels, and they were scaled down to 256×256 to speed up the program. The object image is 19×19 pixels and was taken from the left stereo image. An iterative step in the program takes this 19×19 object image as a sliding window and correlates it with the left image. The result of each correlation is stored in a correlation matrix. This correlation process is then repeated for the right image. The purpose of the correlation algorithm is to determine exactly where the object is located in each stereo image. This location occurs at the point of maximum correlation magnitude in the correlation image. The program then calculates the linear displacements (the DL and DR values) of each maximum correlation point with respect to the center column. Finally, it calculates a range for the object relative to each camera and averages those two ranges together. However, this range is inaccurate because it does not take into account the fact that the range of an object is non-linear in appearance, and also due to the non-linearity of the cameras' lenses. Therefore, this range is corrected by multiplying it with a correcting coefficient

The algorithm was applied to 5 different objects shown in Figure 4, and seems to work well for objects located in front of the focal point, 8.6 meters away. However, range calculation for any object further than that is incorrect. For instance, the algorithm, before correction, calculates the Grissom Hill (obj5, in Figure 4) on the horizon to be only 13.1 meters away. This is not the true ground-range of this object as reported by NASA. This inaccuarte range calculation failed to take into account that lateral distances appear much smaller as they get further away. For example, if a person sees an object move from 1 meter to 10 meters, it will appear 1 / 10 of its original size at 1 meter. Therefore, in order to correct ranges beyond the focal-point distance, the averaged uncorrected pre-range is multiplied by a correction coefficient factor obtained from curve fitting the following equation:

$$y = \frac{-\alpha}{\beta x - c}$$

where x is the pre-range obtained, y is the correction coefficient, α, β, and c are constants obtained from curve fitting and trial-and-error experiments. It was found that for this project $\alpha = 15.82$, $\beta = 3.506$, and $c = 46$. The uncorrected range is corrected by multiplying it with the correction coefficient y. The entire program was developed in MATLAB and is shown in the appendix.

IV. Results

The above algorithm was applied to images sent from Spirit's panoramic cameras on April 2004, available on the Rovers website[2]. Each camera has a 16^{o} full field of view projected onto a 1024×1024 pixel CCD array. They are mounted 1.5 meter above ground level, separated horizontally by 0.3 meter, and each has a 1^{o} toe-in, which causes the cameras to focus on a point 8.6 meter away. Figure 4 shows the five different objects to be range selected from the left camera image.

The results shown in Figures 5 through 9 are organized as follows: the left columns show the original left and resulting left correlation images. The right columns depict the original right and the corresponding right correlation images. The middle image is the target object to be ranged. Figure 5 shows the algorithm results of the first object ranged at about 3.259 meters. Figure 6 shows the results of the second object ranged at 4.0959 meters. The range obtained for the third object is 9.44 meters and it is comparable with the actual known range of approximately 9 meters, obtained from NASA, these results are shown in Figure 7. Object four is ranged at 23.166 meters and the results are shown in Figure 8. The fifth object is ranged at 6977 meters, which is 6.97 % off from the actual known value of 7500 meters.

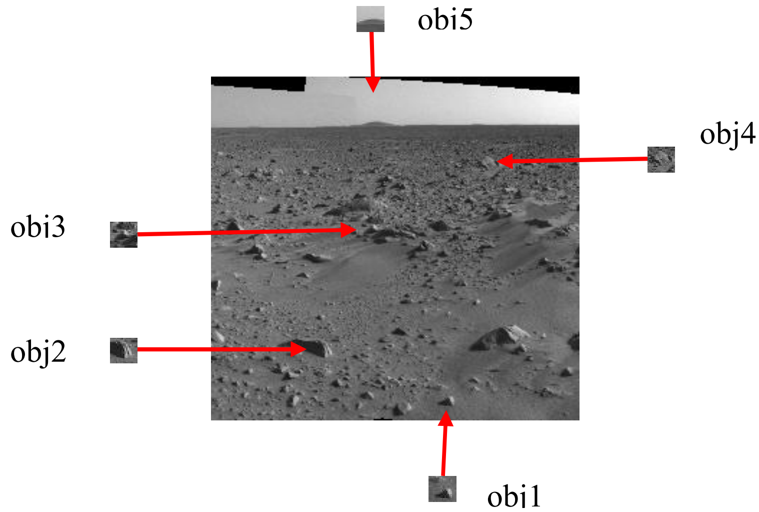

Figure 4: Left camera image and selected objects to be ranged.

V. Conclusions

Students in the EE program at Penn State Harrisburg are required to work on course projects to have a broader and practical experience of the subject taught. This is particularly important in courses such as DSP, image processing and computer vision, communications and electronic design. In this context, the paper presented here, demonstrates an innovative and practical project, developed by an undergraduate student to fulfill a course requirement. This project involved developing an algorithm to range objects using actual data obtained from stereo images sent by MARS Rovers. The initial value obtained by the algorithm was not accurate due to the fact that the range of an object is non-linear in appearance, and to the non-linearity of the cameras, thus the range obtained was subjected to correction to compensate for these non-linearities. The results obtained by the algorithm are comparable to those obtained by NASA.

It takes about ten minutes to run the full program on a Pentium-2 computer if the stereo images are 256×256 and the object image is 19×19. It would probably go much faster if the object image size were decreased. However, with a smaller object image (taken from the left image) comes the risk that the correlation with the right image might yield an incorrect location of the object. The object must be detailed with much contrast so it cannot be mistaken for any similar, but incorrect objects in the right image.

Bibliography

[1] http://www.hbg.psu.edu/hbg/programs/undergrad/eeng.html
[2] http://marsrovers.jpl.nasa.gov/gallery/all/spirit.html
[3]NASA Fact Sheets, "Mars Exploration Rover," http://www.jpl.nasa.gov/facts/mars03rover.pdf
[4] Olson, Clark, Nathiesm Larry, Schoppers, Marcel and Maimone, Mark, "Rover Navegation Stereo Ego-motion," Robotics and Autonomous Systems 43, pp. 215-229, 2003, Elsevier.

[5] Iocchi, Luca, "Stereo Vision: Triangulation" http://www.dis.uniroma1.it/~iocchi/stereo/ triang.html
[6] Shapiro, Linda and Stockman, George, Computer Vision, Prentice Hall, 2001
[7] Wang, Yao, Ostermann Jörn, and Zhang, Ya-Qin, Video Processing and Communications, Prentice Hall, 2002.
[8] Kim, Yeon C. and Aggarwal, J. K. , "Positioning Three-Dimensional Objects Using Stereo Images" IEEE Journal of Robotics and Automation, IEEE Vol. RA-3, NO. 4, August 1987.
[9] Horn, Berthold K.P. , Robot Vision, The MIT Press, McGraw-Hill, 1986.

ERIC WASATONIC
Eric Wasatonic was a undergraduate student in the Electrical Engineering program at Penn State Harrisburg. He obtained his B.S.E.E. degree in the spring 2004.

SEDIG AGILI
Dr. Agili received his BS, MS, and Ph.D. in Electrical and Computer Engineering from Marquette University in 1986, 1989, and 1996, respectively. He has been teaching and conducting research in electronic communications, fiber optic communications and fiber optic sensors at Penn State University at Harrisburg since August 2001.

ALDO MORALES
Dr. Morales received his electronic engineering degree with distinction from the University of Tarapaca, Arica, Chile, and M.S. and Ph.D. degrees in electrical and computer engineering from the State University of New York at Buffalo. His research interests are digital signal and image processing, and computer vision. He is now an Associate Professor of Electrical Engineering at Penn State University at Harrisburg.

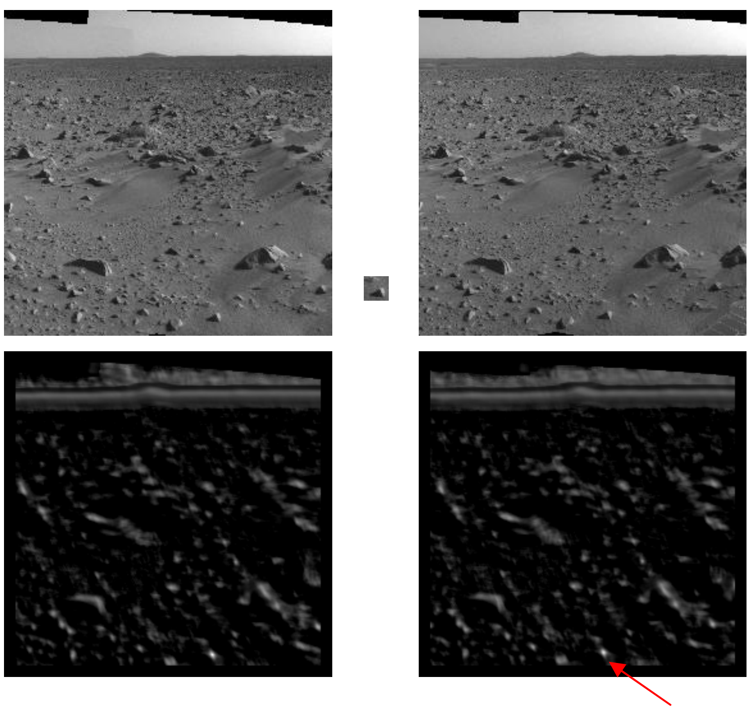

Figure 5: Upper two show left and right camera images, lower two their corresponding correlation with the target image (object 1) shown in the center. Note that the brightest pixel, in the correlation images shows the maximum correlation location.

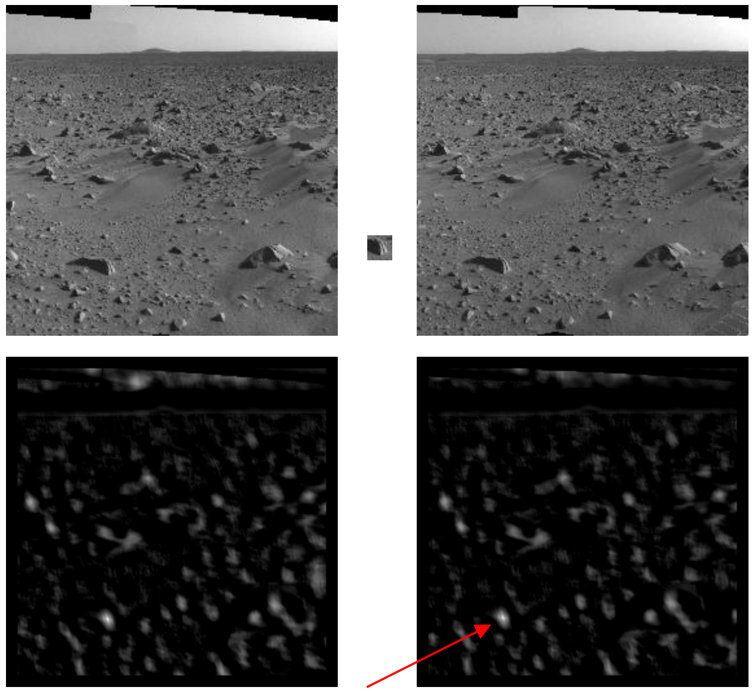

Figure 6: Upper two show left and right camera images, lower two their corresponding correlation with the target image (object 2) shown in the center. Note that the brightest pixel in the correlation images show the maximum correlation location.

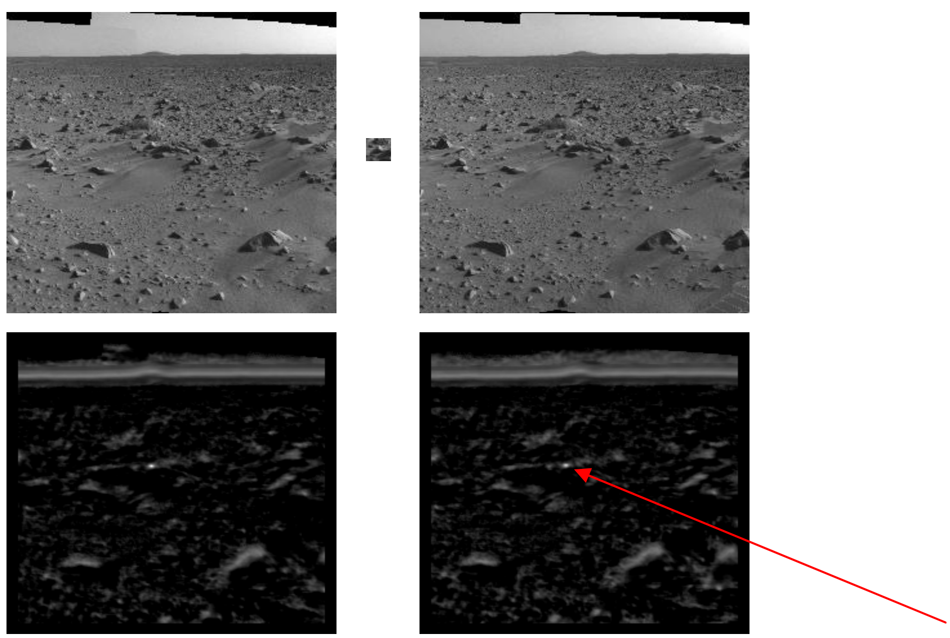

Figure 7: Upper two show left and right camera images, lower two their corresponding correlation with the target image (object 3) shown in the center. Note that the brightest pixel in the correlation images show the maximum correlation location

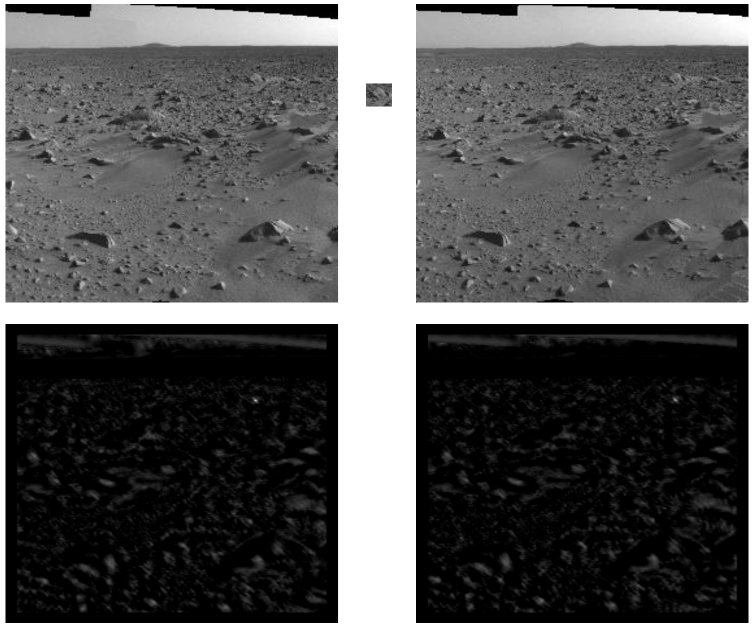

Figure 8: Upper two show left and right camera images, lower two their corresponding correlation with the target image (object 4) shown in the center.

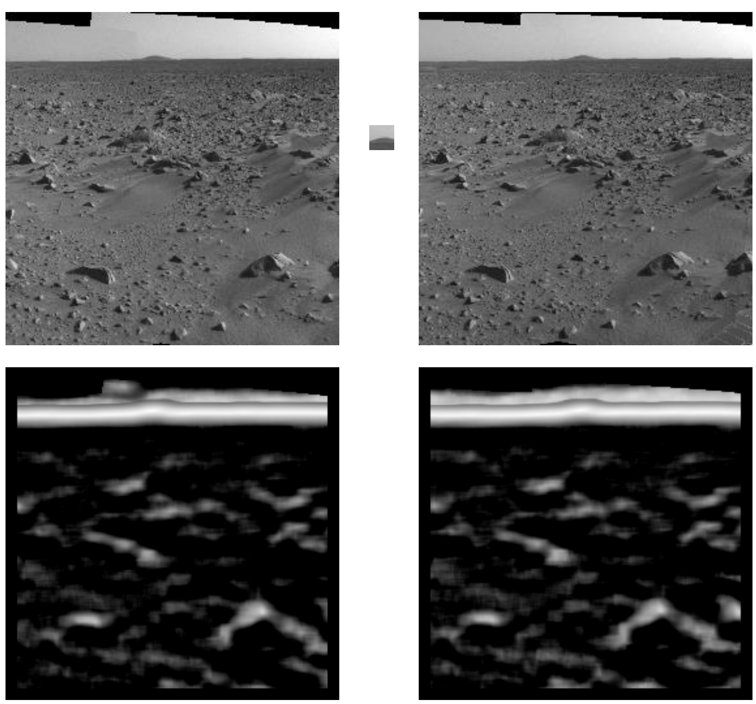

Figure 9: Upper two show left and right camera images, lower two their corresponding correlation with the target image (object 5) shown in the center.

Appendix: MATLAB code

```
function[Range]=stereo_range(imageleft,imageright,objectimage)
%
% Given known parameters of a stereo camera pair, this program reads the two
% stereo images and a
% user-defined object (size = 19*19) within one of those images.  It
% correlates that object with
% each image to determine the location of the object in each image.  Then it
% uses basic triangulation
% to calculate the range from the cameras to the object.
% This program is specifically designed to analyze images from the panoramic
% cameras aboard
% the Mars Exploration Rovers, Spirit and Opportunity.  Each camera has a 16
% degrees (0.2793 radians)
% field of view for each image measuring 1024*1024 pixels (which may be scaled
% down).  The cameras
% are separated horizontally by 0.3 meters and each has a toe-in of 1 degree
% (0.0175 radians).
% Therefore, the focal point of the camera pair lies at a range of 8.5724
% meters.

%-------input left image
close all;
Status=imfinfo(imageleft);     % obtain info of left image
H=Status.Height;
% height of image
W=Status.Width;
% width of image
L=imread(imageleft);           % store left image as L
L=double(L);                   % double L
figure; imshow(uint8(L));      % display L

%-------input right image (same size as left image)
R=imread(imageright);          % store right image as R
R=double(R);                   % double R
figure; imshow(uint8(R));      % display R

%-------input object (size = 19*19) from L
obj=imread(objectimage);       % store right image as obj
obj=double(obj);               % double obj
figure; imshow(uint8(obj));    % display obj

%-------correlate L with sweeping window obj
corrL=zeros(H,W);              % corrL is left correlation image
for i=10:H-9
    for j=10:W-9
        corrL(i,j)=corr2(L(i-9:i+9,j-9:j+9),obj);
    end
end
corrL=double(corrL);                    % double corrL
figure; imshow(uint8(255*corrL));       % display corrL
imwrite(corrL,'corr1L.tif','tif');  % save corrL on disk

%-------correlate R with sweeping window obj
corrR=zeros(H,W);                       % corrR is right correlation image
```

```
for i=10:H-9
    for j=10:W-9
        corrR(i,j)=corr2(R(i-9:i+9,j-9:j+9),obj);
    end
end
corrR=double(corrR);                        % double corrR
figure; imshow(uint8(255*corrR));           % display corrR
imwrite(corrR,'corr1R.tif','tif');          % save corrR on disk

%-------locate brightest points in corrL and corrR
[maxL,maxLcol]=max(max(corrL));
% max_col = column of corr_ matrix containing brightest point
[maxR,maxRcol]=max(max(corrR));

%-------calculate brightest point's displacement from center column
DL=W/2-maxLcol;
% DL > 0 if maxLcol is left of center, DL < 0 if maxLcol is right of center
DR=maxRcol-W/2;
% DR < 0 if maxLcol is left of center, DL > 0 if maxLcol is right of center

%-------use camera specs and law of sines to calculate uncorrected range of
% object
PreRangeL=0.3*sin(1.5533+DR*0.2793/W)/sin(0.0349-(DL+DR)*0.2793/W);
PreRangeR=0.3*sin(1.5533+DL*0.2793/W)/sin(0.0349-(DL+DR)*0.2793/W);
PreRange=(PreRangeL+PreRangeR)/2;
% average of left and right PreRanges

%-------multiply PreRange by a correction coefficient
Range=PreRange*15.82/(46-3.509*PreRange);
```