A STEREO VISION SYSTEM FOR AN AUTONOMOUS VEHICLE

Donald B. Gennery
Computer Science Department
Stanford University
Stanford, California 94305

Abstract

Several techniques for use in a stereo vision system are described. These include a stereo camera model solver, a high resolution stereo correlator for producing accurate matches with accuracy and confidence estimates, a search technique for using the correlator to produce a dense sampling of matched points for a pair of pictures, and a ground surface finder for distinguishing the ground from objects, in the resulting three-dimensional data. Possible ways of using these techniques in an autonomous vehicle designed to explore its environment are discussed. An example is given showing the detection of objects from a stereo pair of pictures.

KEY WORDS: Computer vision, Stereo vision, Matching algorithm, Robots.

Introduction

This paper describes a stereo vision system for use by a computer-controlled vehicle which can move through a cluttered environment, avoid obstacles, navigate to desired locations, and build a description of its environment. One possible application of such a vehicle is in planetary exploration. Our experimental vehicle is described in [4].

As the vehicle moves about, it takes stereo picture pairs from various locations. This could be done with two cameras mounted on the vehicle, but with our present vehicle with one camera, it is done with the vehicle at two locations. Each of these stereo pairs is processed to extract the needed three-dimensional information, and then this information from different pairs can be combined in further processing.

The processing of the stereo pairs is done as follows. First, an interest operator finds small features with high information content in the first picture. Then, a binary search correlator finds the corresponding points in the other picture. (The interest operator and the binary search correlator were both developed by Moravec [4].) Next, a high-resolution correlator is given these matched pairs of points. It tries to improve the accuracy of the match, and it produces an accuracy estimate in the form of a two-by-two covariance matrix, and a probability estimate giving the goodness of the match. The coordinates of these matched points are corrected for camera distortion as described by Moravec [4]. A stereo camera model solver then uses these matched pairs of points to find the five angles that relate the position and orientation of the two camera locations. The accuracy estimates are used by the camera model solver to weight the individual points in the solution and to compute accuracy estimates of the resulting camera model. A dense sampling of points is now matched over the pictures. The known camera model is used to restrict the search for these matches to one dimension, and by first trying matches approximately the same as neighboring points that have already been matched, often no search is needed. In any case, the precise matches are produced by the high-resolution correlator, and its probability estimates are used in guiding the search. After these matched points are corrected for camera distortion, distances to the corresponding points in three-dimensional space are computed, using the known camera model. The accuracy estimates of the matches and of the camera model are propagated into accuracy estimates of the computed distances. The three-dimensional information for all of the matched points is now transformed into a coordinate system approximately aligned with the horizontal surface. (The high-resolution correlator, the stereo camera model solver, and the technique for producing the dense sampling of matches are described later in this paper.)

Information from more than one stereo pair can be combined to produce a more complete mapping of points over the area. A ground surface finder is then used to find the ground for portions of the scene, which may be tilted slightly relative to the assumed horizontal coordinate system. (The ground surface finder is described later in this paper.) Points which lie sufficiently above the ground surface can be assumed to lie on objects. (In the process of finding the ground surface and finding objects, the accuracy and probability estimates are useful.)

Stereo Camera Model Solver

If the image plane coordinates of several pairs of corresponding points in a stereo pair of images have been measured, it is possible in general to use this information to compute the relative position and orientation of the two cameras, except for a distance scale factor. Once this calibration has been performed, the distance to the object point represented by each pair of image points can be computed.

A procedure that performs the above stereo camera model calibration by means of a least-squares adjustment has been written. It includes automatic editing to remove wild points, the use of a two-by-two covariance matrix for each point for weighting purposes, estimation of an additional component of variance by examination of the residuals, and propagation of error estimates into the results.

Consider any point in the three-dimensional scene. Let the coordinates of the image of this point in the Camera 1 film plane be x_1,y_1 and the coordinates of its image in the Camera 2 film plane be x_2,y_2. Image point x_1,y_1 corresponds to a ray in space, which, when projected into the Camera 2 film plane, becomes a line segment. The distance (in the Camera 2 film plane) from image point x_2,y_2 to the nearest point in this line segment is the magnitude of the error in the matching of this point. This error is a function of the angles which define the relative position and orientation of the two cameras. (These angles are the azimuth and elevation of the position of Camera 2 relative to the position of Camera 1, and the pan, tilt, and roll of Camera 2 relative to the orientation of Camera 1.) The camera calibration is done by adjusting these angles to minimize the weighted sum of the squares of these errors for all of the points that are used. Since the problem is nonlinear, the procedure uses partial derivatives to approximate the problem by the general linear hypothesis model of statistics, and iterates

to achieve the exact solution.

The automatic editing is done as follows. First, a weighted least-squares solution as described above is done using all of the points. Then the point which has the largest ratio of residual to standard deviation of the residual is found. This point is tentatively rejected, and the solution is recomputed without this point. If this point now disagrees with the new solution by more than three standard deviations, it is permanently rejected, and the entire process repeats. Otherwise, the point is reinstated, and the process terminates. However, if an F test comparing the computed and given values of the additional variance of observations shows the solution that includes the point to be bad, the point in question is rejected in any event.

A more complete description of the camera model solver can be found in [1].

High-Resolution Correlator

Consider the following problem. A pair of stereo pictures is available. For a given point in Picture 1, it is desired to find the corresponding point in Picture 2. It will be assumed here that a higher-level process has found a tentative approximate matching point in Picture 2, and that there is an area surrounding this point, called the search window, in which the correct matching point can be assumed to lie. A certain area surrounding the given point in Picture 1, called the match window, will be used to match against corresponding areas in Picture 2, with their centers displaced by various amounts within the search window in order to obtain the best match.

Thus when the matching process (correlator) is given a point in one picture of a stereo pair and an approximate matching point in the other picture, it produces an improved estimate of the matching point, suppressing the noise as much as possible based on the statistics of the noise. It also produces an estimate of the accuracy of the match in the form of the variances and covariance of the x and y coordinates of the matching point in the second picture, and an estimate of the probability that the match is consistent with the statistics of the noise in the pictures, rather than being an erroneous match. This probability will be useful in guiding a higher-level search needed to produce a dense sampling of matched points.

Let $A_1(x,y)$ represent the brightness values in Picture 1, $A_2(x,y)$ represent the brightness values in Picture 2, x_1,y_1 represent the point in Picture 1 that we desire to match, x_2,y_2 represent the center of the search window in Picture 2, w_m represent the width of the match window (assumed to be square), and w_s represent the width of the search window (assumed to be square), where x and y take on only integer values.

The following assumptions are made. A_1 and A_2 consist of the same true brightness values displaced by an unknown amount in x and y, with normally distributed random errors added. The errors are uncorrelated with each other, both within a picture (autocorrelation) and between pictures (cross correlation), and the errors are uncorrelated with the true brightness values. (The asuumptions concerning errors hold fairly accurately for the usual noise content of pictures. The assumption concerning the true brightness values will be relaxed slightly below to allow bightness bias and contrast changes. However, another type of change is perspectve distortion, which can be important with large match windows, but it will not be discussed here.)

We temporarily assume that the variance of the errors is known for every point in each picture.

We now wish to find the matching point x_m,y_m which will produce the best match of $A_2(x+x_m-x_1,y+y_m-y_1)$ to $A_1(x,y)$ in some sense. Traditionally the match which maximized the correlation coefficient between A_1 and A_2 has been used [2]. Indeed, this is a reasonable thing to do if one of two functions has no noise. However, here both functions have noise. This fact introduces fluctuations in the cross-correlation function which may cause its peak to differ from the expected value. *Ad hoc* smoothing techniques could be used to reduce this effect, but an optimum solution can be derived from the assumed statistics of the noise.

Let ϵ represent the w_m^2 - vector of the differences $A_2(x+x_m-x_1,y+y_m-y_1)$ - $A_1(x,y)$ over the w_m by w_m match window, for a given trial value of x_m,y_m, and let x_c,y_c represent the true (unknown) value of x_m,y_m. Let P represent a probability and p represent a probability density with respect to the vector ϵ. Then by Bayes' theorem

$$P(x_m,y_m=x_c,y_c \mid \epsilon) = \frac{P(x_m,y_m=x_c,y_c)\, p(\epsilon \mid x_m,y_m=x_c,y_c)}{\Sigma\, P(x_m,y_m=x_c,y_c)\, p(\epsilon \mid x_m,y_m=x_c,y_c)}$$

If we assume that the *a priori* probability $P(x_m,y_m=x_c,y_c)$ is constant over the search window and is zero elsewhere, this reduces to

$$P(x_m,y_m=x_c,y_c \mid \epsilon) = k\, p(\epsilon \mid x_m,y_m=x_c,y_c)$$

where k is any constant of proportionality. Since ϵ consists of uncorrelated normally distributed random variables,

$$p(\epsilon \mid x_m,y_m=x_c,y_c) = k \prod \exp\left(-\frac{.5\,\epsilon_i^2}{\sigma_1^2+\sigma_2^2}\right)$$

$$= k \exp\left(-\frac{.5\,\Sigma\,\epsilon_i^2}{\sigma_1^2+\sigma_2^2}\right)$$

$$= k\, w$$

where

$$w = \exp\left(-.5\,\Sigma\,\frac{\epsilon_i^2}{\sigma_1^2+\sigma_2^2}\right)$$

and where ϵ_i denotes the components of ϵ, σ_2 and σ_2 are the standard deviations of A_1 and A_2, and the product and sum are taken over the match window. (Very often, the the variances σ_1^2 and σ_2^2 can be considered to be constant. In this case, the summation can be reduced to the sum of the squares of the differences over the march window, with the sum of the two variances factored out.) Thus,

$$P(x_m,y_m=x_c,y_c \mid \epsilon) = k\, w$$

So far, the derivation is quite usual. If we simply wanted to maximize P (for the maximum likelihood solution), we would minimize the above sum (that is, use a weighted least-squares solution). However, because of the fluctations in w caused by the presence of noise in both images, the peak of P in general differs from the center of the distribution of P in a random way due to the random nature of the errors.

Therefore, we define the optimum estimate of the matching position to be the mathematical expectation of x_m,y_m according to the above probability distribution. Thus, letting (x_0,y_0) represent this optimum estimate, we have

$$x_0 = \frac{\Sigma\, w\, x_m}{\Sigma\, w}$$

$$y_0 = \frac{\Sigma\, w\, y_m}{\Sigma\, w}$$

where the sums are taken over the search window. The variances and covariance of x_0 and y_0 are given by the second moments of the distribution around the expected values:

$$\sigma_x^2 = \frac{\Sigma\, w\, x_m^2}{\Sigma\, w} - x_0^2$$

$$\sigma_y^2 = \frac{\Sigma\, w\, y_m^2}{\Sigma\, w} - y_0^2$$

$$\sigma_{xy} = \frac{\Sigma\, w\, x_m\, y_m}{\Sigma\, w} - x_0\, y_0$$

The covariance matrix of x_0 and y_0 consists of σ_x^2 and σ_y^2 on the main diagonal and σ_{xy} on both sides off the diagonal.

It might appear that the above analysis is not correct because of the fact that certain combinations of errors at each point of each picture are possible for more than one match position, and the probability of these combinations is split up among these match positions. However, this fact does not influence the results, as can be seen from the following reasoning. The possible errors at each point of each picture form a multidimensional space. When a particular match position is chosen, a lower-dimensioned subspace of this space is selected, in order to be consistent with the measured brightness values. When another match is chosen, a different subspace is selected. These two subspaces in general intersect, if at all, in a subspace of an even lower number of dimensions. Thus the hypervolume (in the higher subspace) of this lower subspace is zero. Therefore, the fact that the two subspaces intersect does not change the computed probabilities.

Now suppose that the standard deviations σ_1 and σ_2 are not known. It is possible to estimate them (actually, the sum of their squares, which is what is needed in the equation for w) from the data if it is assumed that they are constant, that is, the noise does not vary across the pictures. Let v equal the constant value of $\sigma_1^2 + \sigma_2^2$. Then $\epsilon \cdot \epsilon / w_m^2$ (the mean square value of the components of ϵ) is an estimate for v, where $\cdot$ denotes the vector dot product. However, this value is different for each possible match position x_m, y_m. The method used to obtain the best value for v is to average all of these values for v, weighted by the probability for each match position $p(x_m, y_m = x_c, y_c \mid \epsilon) = w$. Thus a preliminary variance estimate is computed by

$$v' = \frac{\Sigma\, w\, \epsilon \cdot \epsilon}{w_m^2\, \Sigma\, w}$$

where the sums are taken over the search window. However, this averaging process introduces a bias because of the statistical tendency for the smaller values to have the greater weights. It can be shown that this effect causes the estimate of variance to be too small by a ratio that can be anywhere from .5 to 1. Therefore, an empirically determined approximate correction factor is applied to the variance estimate as follows:

$$v = \frac{v'}{1 - 0.5\,(1 - \frac{u}{v})^{0.3}}$$

where u is the minimum value of $\epsilon \cdot \epsilon / w_m^2$ over the search window. Since the computation of w requires the value of $\sigma_1^2 + \sigma_2^2$ (=v), the above process is iterative.

An estimate of an upper limit to the variance is also computed from the high-frequency content of the pictures. First,

$$U = \frac{[A(x-1,y) + A(x+1,y) + A(x,y-1) + A(x,y+1) - 4\,A(x,y)]^2}{20}$$

Then U is averaged over an appropriate local window and the results for the two pictures are added together to form the estimate of the upper limit of v.

The overall variance estimate used in the above equations is obtained by an appropriate weighted combination of the *a priori* given value, the derived value, and the computed upper limit.

The probability of a correct match is computed by comparing the derived variance to the *a priori* variance and the upper limit (high-frequency variance) by means of F-tests.

Because of the finite window size, the computed covariance matrix will be an under-estimate. An approximate correction for this effect is made by computing the eigenvalues and eigenvectors of the covariance matrix, applying a correction to the eigenvalues, and then reconstructing the covariance matrix from the eigenvalues and eigenvectors.

The above computations assume that the shift between the two pictures is always an integer number of pixels. In cases where the correlation peak is broad, the smoothing process inherent in the moment computation for x_0, y_0, σ_x^2, σ_y^2, and σ_{xy} cause a reasonable interpolation to be performed if the correct answer lies between pixels. However, when the correlation peak is sharp, this will not happen, and the answer will tend towards the nearest pixel to the correct best match. This is not particularly serious insofar as it affects the position estimate, but it can have a serious effect on the probability estimate. This is because the ϵ vector should be much smaller at the correctly interpolated point than it is at the nearest pixel, because of the sharp peak. Therefore, the probability may come out much too small, indicating a bad match, whereas the match is really good but lies between pixels. To overcome this deficiency, linear interpolation adjustments are made to the variance and probability, and the covariance matrix is augmented to allow for interpolation error.

Since there may be changes in brightness and contrast between the two pictures of the stereo pair, the correlator can adjust a bias and scale factor relating the brightness values in the two pictures. This requires modifying the mathematics given above. Instead of actually using the sum of squares of differences $\Sigma\, \epsilon_i^2$, in the above equations, the moment about the principle axis of the function relating the two sets of brightness values is used. However, the sum of the squares of the differences is still the main ingredient in this computation. Included in this computation are *a priori* weights on the given values of brightness bias and scale factor (contrast). Thus the bias and scale factor can be constrained according to the amount of knowledge about them from other sources, if any.

As stated above, when the variance is assumed to be constant, a major portion of the computation is the sum of squares of differences $\Sigma\, \epsilon_i^2$. This are computed by a very efficiently coded method developed by Moravec [4]. Its inner loop (each term of the summation) requires about one microsecond on the PDP KL10.

Searching for Stereo Matches

Once the stereo camera model is known, the search for matching points in the two pictures is greatly constrained. A point in Picture 1 corresponds to a ray in space, which, when

projected into Picture 2, becomes a line segment terminating at the point corresponding to an infinite distance along the ray. Furthermore, by first trying a match with approximately the same stereo disparity as neighboring points that already have been matched, the search can be eliminated for many points. One criterion for deciding when to accept this tentative match is the probability value returned by the high-resolution correlator. Also, when a search is made, the likeliest correct match is indicated by the highest probability value.

The method used here is similar in some ways to matching techniques used by others (for example, Quam [5] and Hannah [2]). However, there is no region growing in the sense of Hannah, since the equivalent operations are left until later in the processing. Instead, the stereo disparities are allowed to vary in an arbitrary way over the picture, subject to some local constraints discussed later. Furthermore, the acceptance of matches is guided by the probability values. Also, even in areas of low information content, the noise suppression ability of the high-resolution correlator often allows useful results to be obtained. If the content is too low, the correlator indicates this fact by producing very large values for the standard deviations of the two position coordinates. When this happens, the searching can be inhibited to save computer time, but even if this is not done, the results are still as good as the standard deviations indicate. (Actually, the correct test to indicate no useful information is to see if both eigenvalues of the covariance matrix are large. Both standard deviations might be large, but if only one eigenvalue is large, an accurate distance can still be computed for this point unless the corresponding eigenvector is almost parallel to the projected line segment.)

The method currently used is approximately as follows:

1. Divide Picture 1 into square windows, denoted here as "areas", the center of each of which is considered to be a point to be matched to the center of a similar area in Picture 2 in the following steps. (These areas normally would be equal in size to the match window of the high-resolution correlator.)

2. Select a set of starting areas. (Currently a column near the edge of the Picture is used, but this will soon be changed to the points which were produced by the interest operator and binary-search correlator and were not rejected by the camera model solver.)

3. Try areas adjacent (including diagonally adjacent) to areas already tried, where possible working in the direction of the projected line segments in Picture 2 towards the infinity points.

4. If there are at least two already matched areas adjacent to the area in question and the disparities of all adjacent matched areas agree within a tolerance, apply the high-resolution correlator with the search window centered on the position corresponding to the average disparity of these neighbors. Otherwise, go to 6.

5. If the probability returned by the correlator in step 4 is greater than 0.1, accept this match and go to 8.

6. Starting at the infinity point, search along the projected line segment in Picture 2, applying the search window of the high-resolution correlator at points with a spacing of half of the search window width, but not at previously matched areas.

7. Of those matches found in step 6, select the one for which the correlator returned the highest probability. If this probability is greater than 0.1 and at least one neighboring area (including these tentative matches) agrees in disparity and has a probability greater than 0.01, or vice versa, accept this match. Otherwise, of those matches found in step 6 with probability greater then 0.1, if any, accept the one whose disparity agrees most closely with its neighbors, if within the tolerance.

8. When the current group of areas being tried is exhausted, go to 3. If there are no areas left, finish.

Some improvements can be made to this algorithm in the future. For example, another pass can be made over the data to clean things up, utilizing the fact that most areas have more matched neighbors than they did when things were progressing in a basically one-directional manner. Another possibility is to change step 7 in the following way. The best match from those found in step 6 would not be selected immediately. Instead, all of the potential matches with sufficiently high probability would be saved until the entire picture had been processed. Then a cooperative algorithm similar to that discussed by Marr and Poggio [3] could be used to choose the best matches. This should produce more reliable matches, but with a large increase in computation time.

Ground Surface Finder

Once the three-dimensional positions of a large number of points in an outdoor scene have been determined, it is desired to determine which points are on the ground and which are on objects above the ground. By taking a sufficiently small portion of the scene the ground can be approximated by a simple surface whose equation can be determined, and the points which lie above this surface by more than an appropriate tolerance can be assumed to be on objects above the ground.

Such a procedure has been written, which assumes in general that the ground surface is a two-dimensional second degree polynomial. However, weights can be given to *a priori* values of the polynomial coefficients, to incorporate any existing knowledge about the ground surface into the solution. For example, the second degree terms can be weighted out of the solution altogether, so that the ground surface reduces to a plane.

To determine a ground surface from a given set of data, a set of criteria which define what is meant by a good ground surface is needed. These include the number of points within tolerance of the surface (the more the better), the number of points which lie beyond tolerance below the surface (the fewer the better, since these would be due to errors such as mismatched points in a stereo pair), and the closeness of the surface coefficients to the *a priori* values. Note that the number of points above the surface does not matter (other than that it detracts from the number within the surface), because many points can be on objects above the ground. A score for any tentative solution is computed based on these criteria, and the solution with the highest score is assumed to be correct, although a solution with a lower score can be selected by a higher level procedure using more global criteria. The scoring function currently used is

$$V = \frac{N - m}{N + n - 2m} - \frac{B^2}{b^2 + B\,b} - \sum_i \left(\frac{c_i - c_i'}{3\sigma_i}\right)^2$$

where N is the number of points within tolerance of the surface (these points were used to determine the surface by a least-squares fit), n is the *a priori* expected number of points in the surface, B is the number of points below the surface by more than the tolerance, b is the *a priori* approximate maximum number of points below the surface, the c_i are the coefficients of the fitted surface, c_i' are their *a priori* values, σ_i are the standard deviations of these *a priori* values, and m is the number of these coefficients which were adjusted.

Finding the best solution (according to the scoring function) out of all of the possible solutions is a search problem. What is needed is a method which will be likely to find the correct solution without requiring huge amounts of computer time. The method used uses some heuristics to lead the search

to the desired solution. Its main points can be described briefly as follows.

First, a least-squares solution is done using all of the points. This fit is saved for refinement leading to one tentative solution. Then all points within tolerance of this fit or too low, but not less than half of the points used in this fit, are selected, and another least-squares fit is done on these points and saved. This process repeats until there are too few points left. (This portion of the algorithm drives downward to find the low surfaces, even though there may a large amount of clutter above them.)

The refinement of each of the above fits is done as follows. The standard deviation of the points used in the fit about the fitted surface is computed. Then all points within one standard deviation (or within the original tolerance) of the surface are used in a new least-squares fit. This process continues until it stabilizes, in which case the score of the result is computed, or until there are too few points in the solution. (This portion of the algorithm rejects erroneous points and some clutter, in order to find well-defined surfaces.)

Results

Figure 1 shows a stereo pair of photographs taken from positions approximately 1.8 feet apart in a parking lot. Each digitized picture is 270 pixels wide and 240 pixels high.

Figure 2 shows the points found in the left picture by the interest operator and the corresponding points (using the same arbitrary symbols) matched in the right picture by the binary search correlator. The points encircled were rejected because of low probability (<0.1) estimated by the high-resolution correlator. The points in squares were rejected by the editing in the camera model solver. The remaining points then determined the camera model solution.

Figure 3 shows the matches produced by the searching algorithm, constrained by this camera model, using the high-resolution correlator (with eight-by-eight windows). Notice that the algorithm made several incorrect matches, particularly in the left foreground. This is a result of the fact that there was very little contrast in the texture on the pavement, resulting in a low signal-to-noise ratio. Nevertheless, there are a sufficient number of correct matches so that the later stages of processing are not bothered by these errors.

The left side of Figure 4 shows the component of distance (in feet) parallel to the optical axis, computed from the matches for all points that the algorithm matched, superimposed on the left picture. (Single characters are use for these plots, with 0 through 9, A through B, and a through b representing 0 through 61. The number sign represents values from 62 to 100, and the infinity symbol represents everything greater than 100. The right side of Figure 4 shows the relative standard deviation computed for these points, to the nearest foot. (The relative standard deviation indicates only those errors which tend to differ for nearby points. The total standard deviation is larger and indicates the absolute accuracy of the distances.)

Figure 5 shows the results of transforming the three-dimensional information represented by the distances into an approximately horizontal coordinate system. The Camera 1 position is at the bottom center of the figure, looking towards the top. Plotted are the heights above the reference plane in feet, using the single characters described above.

The portion of the data between ranges of 10 feet and 50 feet was given to the ground surface finder. The heights of these points above the resulting plane are shown in Figure 6.

Figure 7 is the same as Figure 6, except that it shows only those points with a height of at least two feet. The points above this threshold are on the two vehicles in the pictures, the light poles, and some shrubbery near the light poles, with a few error points.

Future Plans

It is planned to use the stereo system in a system for operating an experimental exploring vehicle. Stereo pairs will be taken from various locations and their results will be combined. Points which lie sufficiently above the ground will be clustered into individual objects, and simple size and shape information will be computed for each object. A data structure containing a catalogue of objects, with their locations, sizes, and shapes, and properties of the ground, will be built up as the vehicle moves through its environment. By comparing this information to older portions of the data structure, the vehicle can determine if it is in a previously seen area.

There are several opportunties for the previously described components of the system to cooperate and to pass information back and forth. For example, the high-resolution correlator has several parameters used to give it *a priori* information on the noise level in the data and changes in brightness and contrast between the two pictures. It also produces *a posteriori* estimates of these quantities. These results from early applications of the correlator (for example with the points used to obtain the camera model and not rejected by the camera model solver) can given to the correlator in later applications. Also, when the ground surface finder is given points from a certain portion of the scene, it can be given *a priori* values and weights for the surface, which can be obtained from ground surface solutions for adjacent areas. Furthermore, if its apparently best solution does not agree with those of adjacent areas, an alternate solution from the ground surface finder can be used which agrees better with neighboring areas, even though it may not be as good according to local criteria. All of these features may be implemented in the future.

Other possibilities include an automatic segmenter to produce regions of complicated terrain for the ground surface finder to work on, and the addition of *a priori* knowledge about the environment, including models of objects expected.

Acknowledgements

The research described herein was performed at the Stanford Artificial Intelligence Laboratory under NASA contract NASW-2916 and the Advanced Research Projects Agency Order 2494 contract MDA903-76-C-0206.

References

1. Gennery, D. B., "Stereo-Camera Calibration," Stanford Artificial Intelligence Laboratory Memo in preparation.

2. Hannah, M. J., "Computer Matching of Areas in Stereo Images," Stanford Artificial Intelligence Laboratory Memo AIM-239, Stanford University, July 1974.

3. Marr, D. & Poggio, T., "Cooperative Computation of Stereo Disparity," *Science*, Vol. 194, pp. 283-287, October 15, 1976.

4. Moravec, H. P., "Towards Automatic Visual Obstacle Avoidance," in these Proceedings.

5. Quam, L. H., "Computer Comparison of Pictures," Stanford Artificial Intelligence Laboratory Memo 144, Stanford University, 1968.

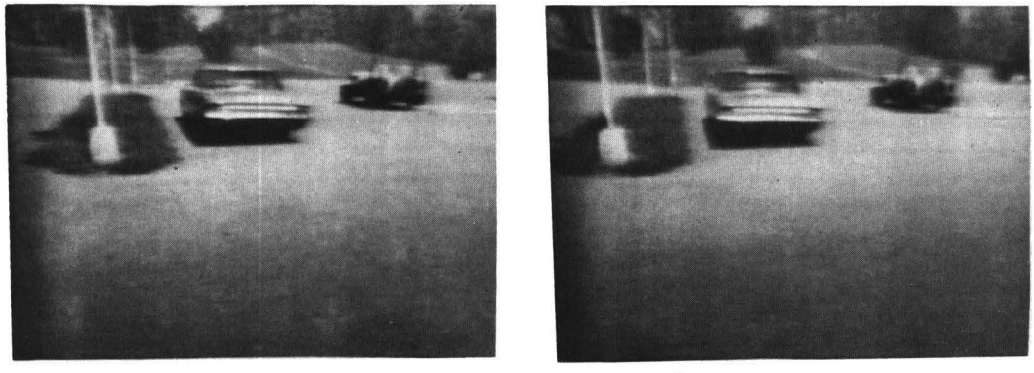

Figure 1. Stereo pair of pictures.

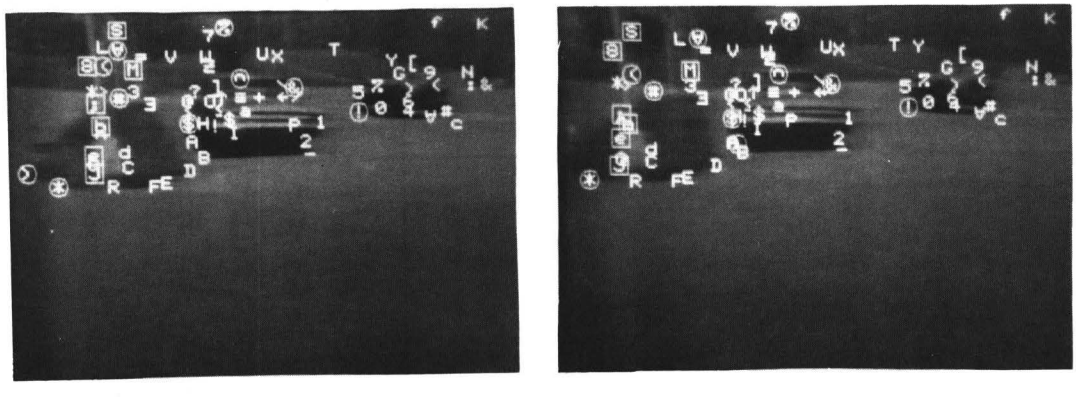

Figure 2. Matching points found by binary-search correlator, rejected by high-resolution correlator (circled), and rejected by camera model (boxed).

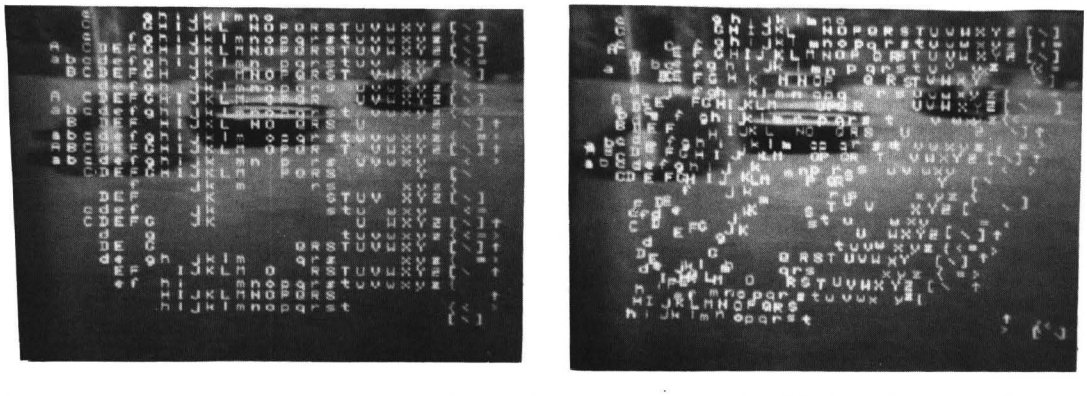

Figure 3. Matching points found by search algorithm using camera model and high-resolution correlator.

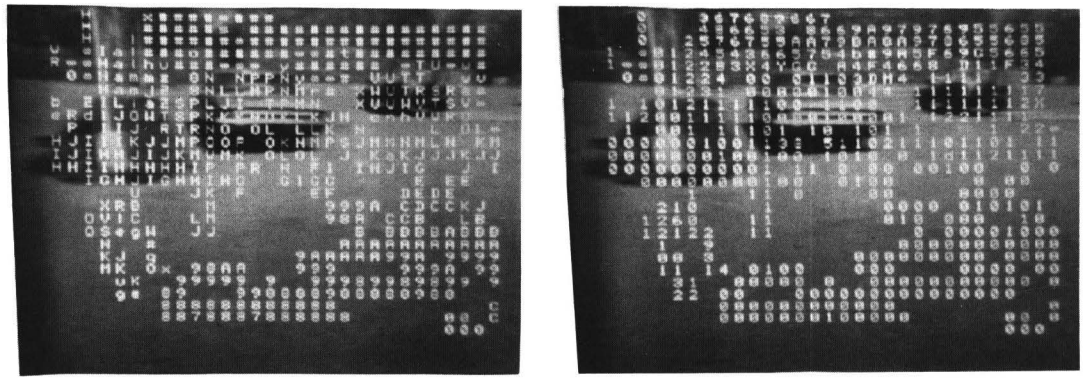

Figure 4. Distances (left) and relative standard deviations (right) to points in left picture of Figure 1. The symbols are defined in the text.

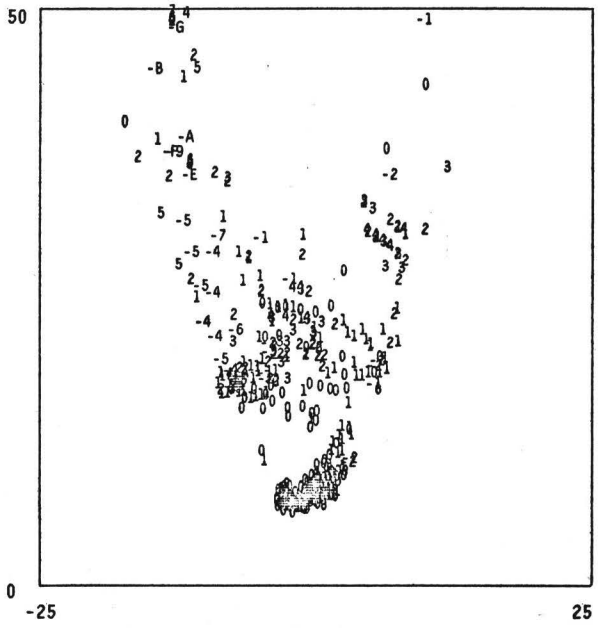

Figure 5. Heights above reference plane.

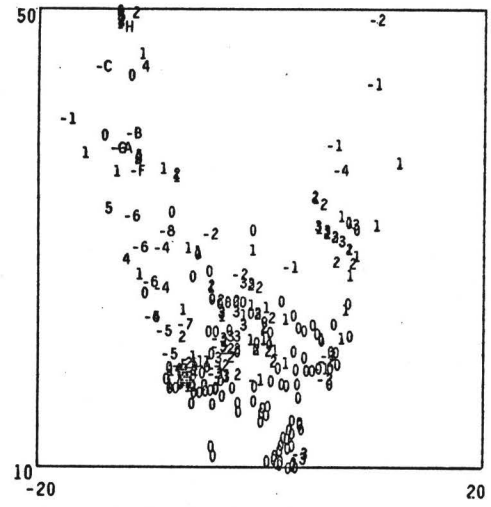

Figure 6. Heights above fitted ground plane.

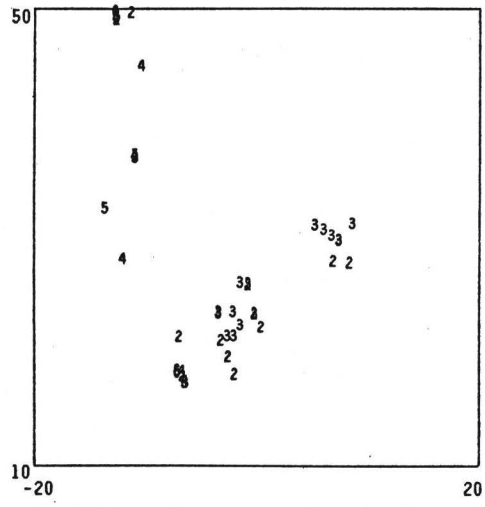

Figure 7. Heights at least two feet above fitted ground plane.