TOWARDS AUTOMATIC VISUAL OBSTACLE AVOIDANCE

Hans P. Morevec, Stanford University This report describes ongoing research on a working system which drives a vehicle through cluttered environments under computer control, guided by images perceived through an onboard tv camera. The emphasis is on reliable and fast low level visual techniques which determine the existence and location of objects in the world, but do not identify them. Included are an interest operator for choosing distinctive regions in images, a correlator for finding matching regions in similar images, a camera solver which determines camera displacement and distance to objects from stereo information tGennery, D.B., this Proceedings] and an automatic geometric distortion corrector for camera nonlinearities. Many of these use pictures reduced in linear dimension by powers of 2 by summation of pixels. Other operators are a high pass filter. a point noise remover, a contrast normalizes a vertical roll corrector, a picture comparator and an operator for reducing pictures by other than powers of two. Our hardware includes an electric vehicle, called the cart, remotely controlled over a CB radio link by a PDP-KL10. It carries a b/w tv camera whose picture is broadcast over a UHF channel, and occasionally digitized by the computer. It has motors for the wheels, steering and camera pan. Each can be made to run forward or backward. There are potentiometers on the steering and pan which enables them to be commanded to point straight ahead. Budgetary and personnel limitations have resulted in crude mechanical arrangements. The motor speeds are poorly regulated, and video is the only feedback to the computer. Dead reckoning errors are about 30%. Our small resources have been spent gaining software experience before undertaking serious hardware work. In my opinion our major hardware limitation is one shared by all other vision work, and AI in general, namely a critical shortage of raw processing power. For instance it would take about 100,000 efficiently programmed PDP-10's to match the human visual system. Results Early versions of the routines described below were used in a program which drove the vehicle in straight lines or uniform arcs. It acquired and tracked distant features, using their motion from frame to frame to build up a model of vehicle response, and to servo on the desired path. With the cart outdoors on a dirty road, it worked well. Ten runs of about 60 steps were completed. The runs were usually terminated by serious hiccups of the radio control link. Each step took the cart 2 feet forward, and used 30 compute seconds. The radio link has since been much improved. Several runs involving the distortion corrector, camera solver and new versions of the interest operator and correlator have been completed. The new program trys to determine the distance to the features by applying the camera solver after tracking them through several images. The performance is poor. The camera solver results are erratic, seemingly due to the degenerate nature of the solution. Objects lying near the camera axis (most of the scene) provide no depth information. Next We are trying a new approach, replacing the camera pan mechanism with one which provides 21 inches of side to side motion, in three 7 in. steps. This should provide adequate parallax, and also close spacing to make the correlations easy. Since the camera motion parameters will be known the correlation searches become one dimensional, and an absolute scale factor is known. The camera solving is also easy. The idea is to locate nearby features in 30 at each vehicle stop. The vehicle motion can be found from the apparent feature motions between stops. The location of the ground can be deduced from the camera height and orientation. Interest Operator This routine is used to acquire new features in a scene. It selects a relatively uniform scattering, to minimize the probability of missing important obstacles, and chooses distictve areas for unambiguous correlation. This is achieved by returning regions which are local maxima of a directional variance measure. Featureless areas and simple edges (which have no variance in the direction of the edge) are thus avoided. Directional variance is measured over small square overlapping windows of specified size (typ. 4x4 to 8x8). Sums of squares of differences of pixels adjacent in each of four directions (horizontal, vertical and two diagonals) over the window are obtained. The variance of the window is the minimum of these four sums. The operator is applied to a reduced version of the picture, where the specified window size shrinks to 2 or 3 pixels. Noise sensitivity is reduced and speed increased. Partly hand coded, the routine takes 75 ms for a 260x240 image, with 8x8 windows.

Binary Search Correlator Given a feature in one picture, the correlator attempts to find the matching region in another image. It takes the position in the first picture, a rectangular search area (often the whole image) in the second picture, and a feature window size n. The search uses a coarse to fine strategy, which begins in reduced versions of the pictures. The order of reduction is chosen to shrink the smaller dimension of the search rectangle to between n and 2n pixels. An n by n window in the shrunken source image, centered on the desired feature, is considered. It covers about 25% of this tiny version of the picture. A correlation coefficient is calculated for each possible placement of this window on the search area. For a search area exactly 2n by 2n, there are $(n+1)^2$ positions. The one with the highest coefficient becomes the search area for the next level of refinement. This is repeated with pictures reduced one step less, i.e. linearly twice as large. An n by n window is again centered around the location of the feature, and is searched for in the best matching window from the previous search, which expands to 2n by 2n at the new reduction. This goes on in successively larger versions of the pictures until an n by n window is matched in the unreduced images. There are about log2(w/n) searches in all, where w is the smaller dimension of the search rectangle in the unreduced picture. This approach has advantages over a simple pass of a correlation coefficient. It needs only 1/150 the number of pixel comparisons to find an 8x8 window in a 256x256 picture (smaller advantage for smaller searches). The simple method comparisons are without context, and a match may be found in totally unrelated parts of the image. In our technique coarse structure guides the higher resolution comparisons, and further speedup is possible because smaller windows work. The searches at coarse levels rarely fail, possibly because noise and distortions are reduced by reduction. The correlation measure used, designed to have limited contrast sensitivity, was obtained by multiplying the normalized correlation coefficient by twice the cosine of the angle with the line a-b. It is: $2\Sigma ab/(\Sigma a^2+\Sigma b^2)$. Normalized correlation is the sum of the pairwise products of a and b divided by the geometric mean of the sum of their squares. The new measure, referred to as pseudo-normalized correlation, is the sum of the products divided by the arithmetic mean of the sums of the squares. By in-line coding the source window and using a table of squares the bulk of the correlation is done in 3 instructions per pixel comparison. An 8x8 window is found in a 260x280 area in 75 ms. The error rate is 10% on interest operator selected features. Typical image pairs are taken two feet apart with a 60 degree lens. Scale Changes As the vehicle moves the image it sees changes. The major element of this transformation is an enlargement of nearby objects. We have tried correlating across images reduced by different geometric scale factors by generating pictures $2^{2/3}$ as large as each of the binary steps. We obtain effective scale changes of 1, $2^{1/3}$, $2^{2/3}$ and 2 by comparing various combinations of reductions of the first and second images. The results are disappointing. The method often introduces as many new errors as it corrects. Experiments in applying it more selectively are planned. Camera Distortion Correction Electron optics tend to have geometric distortions undesirable when using a camera as a measuring instrument. We have written a camera calibration program which is given an image of a square array of black spots on a white background, and told the array to lens center/spot spacing distance ratio. It computes a polynomial for transforming feature image positions accurately to angle in space. It tolerates a wide range of image sizes (3 to 12 spots across) and illumination and arbitrary rotation. After intense fiddling with a training set of 20 images, it has worked without error on 80 widely differing new images. Our test pattern is a ten foot square painted on a wail, with two inch spots at one foot intervals. The algorithm gets an image of such an array, and finds four major peaks in the magnitude of the fourier transform of a reduced version of it, to find its rotation and spacing. The Interest operator is used to find a starting spot, and a special operator, which does local thresholding and finds centroids and moments of black areas, pinpoints all the spots, guided by the rotation/spacing information. A fourth degree least squares polynomial in two variables relating the actual to the ideal position of the spots is then generated. Acknowledgement : This work was supported in part by contracts and grants from ARPA, NASA and NSF.