results. However, other algorithms could benefit from this approach, such as Blake's active splines (Blake, 1992), which represent a low cost way of tracking dynamically moving objects. Instead of sampling an entire image, dynamic foveation could be used to track only the last known position of the spline.

One future experiment we plan to conduct is to study the evolution of optimal sampling strategies based on a given vision task, using a genetic algorithm (Goldberg, 1988). This should give rise to non-rectangular strategies in certain situations, such as log-polar sampling, and would help illustrate the point that the RCV is not limited to rectangular sampling.

6.2 RCV III

The next generation of the RCV will make use of newer chipsets designed for video digitization, such as the Samsung KS0116 series which is a 3-chip set capable of decoding, digitizing and encoding color NTSC, SVHS, and PAL video. This will allow us to generate processed images in real time and display them on color monitors.

We will also be moving away from the 68332BCC as the main controlling processor to Texas Instruments DSP chips, such as the 320C40, which is capable of up to 50 MFlops, as opposed to the 68332's 2 MIPS. The recent reduction in price of these VLSI chips will keep the cost of the RCV near the current level, while greatly enhancing its capabilities.

7. Conclusion

We have presented a low cost, low complexity real time vision system. The RCV is capable of arbitrarily sampling an image, grabbing only those portions of the visual field that may be of interest in a given situation. This flexibility allows the RCV to be used in a large number of applications, primarily as "quick vision" for prototype mobile robots.

We have demonstrated how reduced sampling results in a computation reduction over fully sampling an image. Also, dynamic foveation provides greater rotation and noise tolerance in our Hough transform line tracking algorithm by keeping the visual field to be analyzed in the peak region for this algorithm's performance.

8. Acknowledgments

We would like to thank Andrew Fagg, Michael McHenry, Gaurav Sukhatme, and Jim Montgomery for advice and discussions during the development of this paper, and in the development of dynamic foveation.

9. References

Ross, B., "A Practical Stereo Vision System," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR 93)*, New York, New York, June 1993.

van der Wal, G., Burt, P, "A VLSI pyramid chip for multiresolution image analysis" *International Journal of Computer Vision*, Sept. 1992, 177-189

Bederson, B.B, *A Miniature Space-Variant Active Vision System*, Ph.D. dissertation, NYU department of Computer Science, 1992.

Burt, P., "Smart Sensing within a pyramid vision machine," *Proceedings of the IEEE*, Aug. 1988, pp. 1007-1015.

Horswill, I., Yamamoto, M., *A $1000 Active Stereo Vision System*, submitted to CVPR 94.

Blake, A. and Yuille, A, ed. Active Vision, MIT Press, Cambridge, Mass., 1992

Fagg, A. H., Lewis, M. A., Montgomery, J. F., Bekey, G. A., "The USC Autonomous Flying Vehicle: an Experiment in Real-Time Behavior-Based Control". *Proceedings of the 1993 IEEE Conference on Intelligent Robots and Systems*, July, Yokohama, Japan, pp. 1173-1180.

MC68332 Users Manual, Motorola Inc., 1990.

Davies, E.R., Machine Vision: Theories, Algorithms, Practicalities, Academic Press, New York, 1990

Fagg, A. H., Lewis, M. A., Iberall, T., Bekey, G., "R2AD: Rapid Robotics Application Development Environment." *Proceedings of the 1991 IEEE Conference on Robotics and Automation*, April 1991

Goldberg, D.E., Genetic Algorithms in search, optimization, and machine learning. Addison-Wesley Pub Co, Reading, Mass., 1988

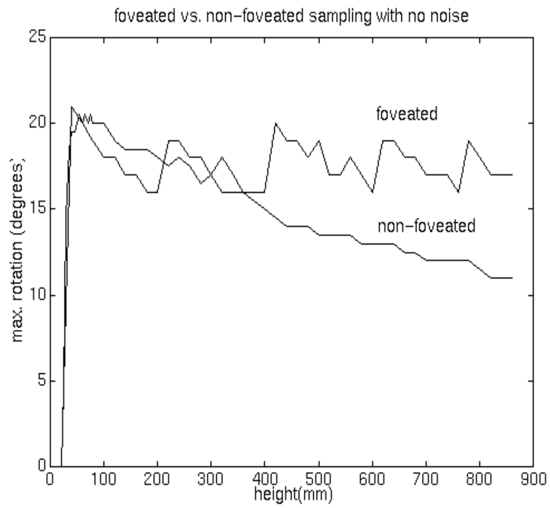

Figure 4. Maximum tolerated rotation of line as a function of height.

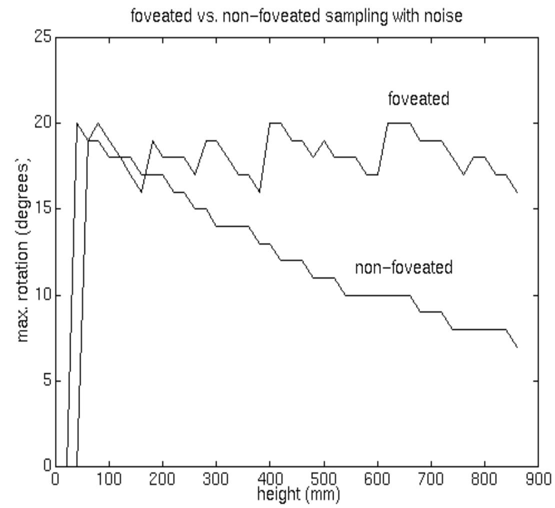

Figure 5. Same as figure 4, but this time with 1/5th of the field covered with noise in the form of bits of paper.

slow down a 68332 enough that it would be unusable in a real time situation.

The improvement in rotation tolerance is shown in Figures 4 and 5. In the non-foveated cases, the maximum allowable rotation decreases as camera height increases. This is because as height increases, the apparent width of the line decreases, therefore as it is rotated, the Hough transform cannot find the peak because it flattens out.

By using dynamic foveation, the relative line width is tacked. When it falls below a given threshold, the system zooms in to the last known position of the line. The line now occupies more of the image, due to the zooming, and this increases the rotation tolerance. Figure 4 shows a comparison of foveated and non-foveated trials. In the foveated case, when the performance starts degrading, the system zooms into the image. The large peaks in the foveated curve show where this happens.

In figure 5, results for noise are presented. The noise used was a section of astro turf, similar to the field at Georgia. The astro turf provided a large number of small reflections when a light was placed by it. Because of this, the non-foveated performance degraded faster than it did without the astro turf.

In the foveated case, however, performance was similar to that when noise was not used. This is because the algorithm stays in the maximum response area. Therefore, by using foveation, the tolerance to noise is increased.

5.1 Comparison to other work

The Cortex-1 system by Bederson(1992) uses special purpose log-polar sensor to generate spatially variant images in real time. Their sensor provides a dense fovea at the center, with exponentially decreasing resolution towards the periphery. This approach would work well in simulating biological approaches to vision.

Gooitzen van der Wal and Peter Burt(1992) have developed an integrated circuit which is capable of building gaussian and laplacian pyramids at about 44 frames per second. Using this, they are able to run multi resolution algorithms in real time.

Ian Horswell(1993) has developed a $1000 vision system which consists of a 68332 directly interfaced to a CCD chip. This provides a 192 x 165 resolution image, which they sub-sample to run stereo tracking algorithms at 7-10 Hz.

While there are many excellent features in all these systems, the prime advantage of the RCV is flexibility. The RCV is capable of log polar sampling, multi resolutional pyramid sampling, and any number of other sampling schemes that can be devised. The disadvantage of the RCV relative to other systems is that this flexibility, combined with the use of a general purpose microcontroller, results in a slower system than what dedicated hardware can deliver.

6. Future Work

6.1. Further applications of dynamic foveation

We also plan to examine other vision algorithms to take advantage of dynamic foveation. We have adapted the simple Hough transform, and it has provided good

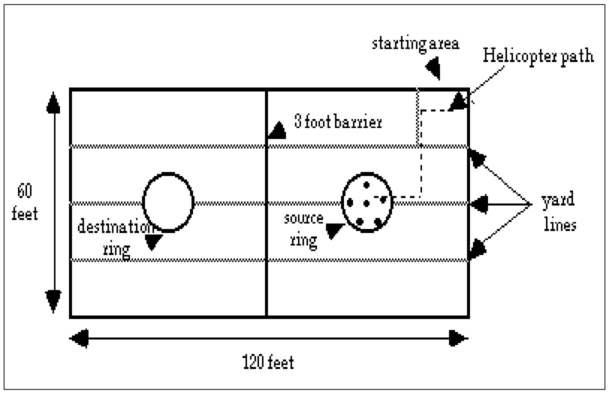

Figure 2. Helicopter contest field and path to source ring.

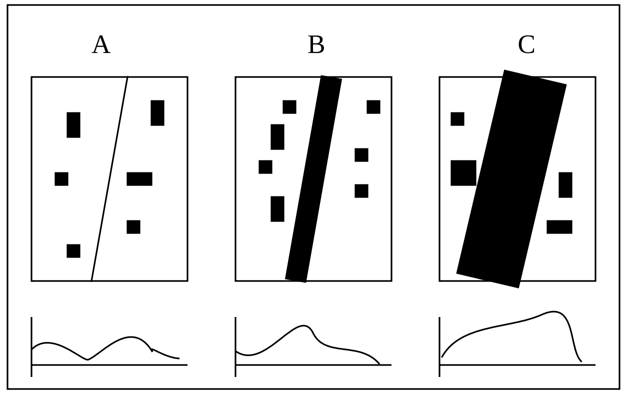

Figure 3. When the line is the size in image B, a clear peak is discernable, as opposed to images A and C.

light enough to be carried on a helicopter with a limited payload capacity.

Our primary strategy has been to track the starting area boundary lines and yard lines on the football field. We assume that we know the width of the lines on the football field, have an estimate of our height, and have control over the yaw axis of the helicopter.

These assumptions allow us to use a simple Hough transform to look for a single line to track (Davies, 1990). In a generalized Hough transform, a line is represented by two parameters, slope and x-intercept. Each point in the image contains a set of lines which can pass through it, and depending on the intensity at that location, votes for its set of lines. The parameterization which receives the most votes is considered to be a line in the image.

To minimize our visual computation costs, we are only actively looking for horizontal and vertical lines. This is possible because of the control we have over the yaw of the helicopter, which reduces the rotational variance of the line. Because we are only looking for vertical and horizontal lines, we can eliminate the slope parameter in our Hough transform. Therefore, we can store all the votes in a vector consisting of a sum of rows and columns and search for a peak in that vector. We find the peak in this vector by finding the center point of a region above a certain threshold. Because this is a very simple peak detection algorithm, there is sensitivity to rotation and noise, both of which are aggravated by the motion of the helicopter.

Again because of the simplicity of this algorithm, there is a peak performance area, where it is capable of tolerating the most rotation and noise. This performance is determined by the width of the line relative to the width of the entire field. If the width is too narrow, the slightest rotation will cause the peak to be too flat to reliably detect. If it is too wide, then it is too hard to discern it from the background, as is shown in Fig. 3.

If the width of the line is known, as it is in the case of the football field, the peak performance point then becomes a function of height of the craft. However, keeping the height of the craft constant at all times is difficult as we sometimes want to be lower or higher, based on proximity to the barrier, the need to have a wide field of view to look for disks, and other factors.

This peak response area was determined with the use of a mock field and a camera and the RCV connected to a PUMA 560 controlled using the RRAD system (Fagg, et al, 1991). This allowed us to control the Puma in Cartesian space and specify height in millimeters and end effector rotation in degrees, giving us a clear view of the peak response area, with and without noise added to the field. We tested cases with and without noise, with the noise being provided by placing the line on astro turf. Astro turf was chosen as noise because it is used on the football field at Georgia Tech where the competition will be held.

5. Results

There are three primary results from our experiments. They are computational time, rotation tolerance, and noise tolerance. The first result is due to our using a sub-sampling strategy, while the latter two are due to dynamic foveation.

The Hough transform we use computes a sum of rows and columns to determine the position of the line. Because of this, its execution time is dependant upon the number of rows and columns that are processed. Given n rows and columns, $O(n^2)$ mathematical operations are performed to find the line.

Our sub-sample of the frame extracts a 32 x 32 array of pixels. A system which ran this algorithm on a full 512 x 512 frame would therefore take 256 times longer (512^2 / 32^2 = 256) to process the image. This would

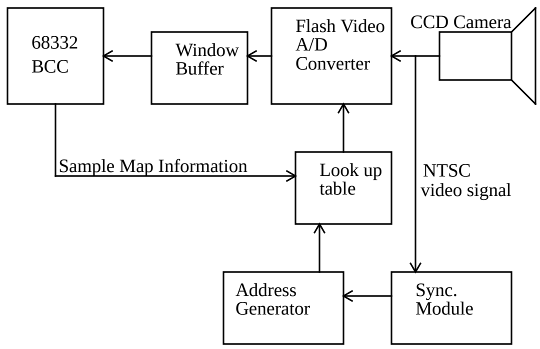

Figure 1. RCV Block diagram.

sparse attention is still paid to the surrounding area, to look for new areas of interest. Therefore, the key to avoiding the loss of information that normally accompanies sub-sampling is to sub-sample the image intelligently.

Our primary demonstration of this will be in the RCV's use in the Association for Unmanned Vehicle Systems autonomous flying vehicle contest, which has been held at the Georgia Institute of Technology for the past four years. The goal of the contest is to build an autonomous flying vehicle which is capable of starting in a pre-determined location on a football field (as shown in Fig. 2) and then move to a source ring which contains randomly placed 6" disks. It must then pick up a disk, transport it over a 3' high barrier, and deposit it in a destination ring. The entire task must be repeated six times in under six minutes.

So far, no one has been able to complete the task. Last year, the USC Autonomous Helicopter project was able to navigate to the source ring using dead-reckoning. Because of weight constraints and the lack of certainty in our knowledge of our position, the craft was unable to pick up a disk. The effort is described in (Fagg et al, 1993).

3. RCV II System Description

The RCV II is the second generation of reduced complexity vision systems built at USC. The first version was a proof of concept model which had many of the same features of the RCV II, but at higher cost and power usage.

Our current version consists of Motorola 68332 BCC and only 15 MSI and LSI chips. Total power consumption is 5 watts. Additionally, we are also able to perform simple operations such as adaptive thresholding and simple Hough transforms at near frame rate and object tracking at half frame rate.

There are four key components to the RCV II, the CPU submodule, analog section, look up table, and window buffer. Each section is independent in operation from the other sections, as shown in Figure 1, with various control signals passing between the modules.

3.1. circuit overview

The Motorola 68332 is a SISD processor based on the 68000 series family, commonly used in the Macintosh. However, as it is a microcontroller, rather than simply a microprocessor, it has a number of features which make it suitable for the control of external hardware. Some of its main limitations are a lack of floating point operators, and a relatively slow clock speed of 16Mhz. (MC68332, 1990)

The analog preprocessor section includes an Analog Devices AD9502 video digitizer. This is a hybrid chip which takes in an NTSC video signal, such as one generated by a camera, TV, or VCR, and outputs 8-bit grayscale data.

This chip is augmented by the use of an LM1881 synchronization module which provides vertical and horizontal sync signals decoupled from the NTSC video signal. The 1881 also provides a signal which indicates when a new frame is beginning.

The most critical section of circuitry is the look up table (LUT). It is here that the sampling strategy is set up. The look up table is a bitmap of the entire video frame. When a pixel is to be sampled, the corresponding bit in the LUT is set to a high state by the CPU. This is done for all the pixels (up to 4096) which need to be sampled.

The system then runs through this map, taking samples where indicated by the map. Again, because there is no constraint on the organization of the map, any sampling strategy is possible, up to the pixel limit of 4K.

The window buffer consists of a bidirectional FIFO which is interfaced to the CPU and the video ADC in the analog section. When the LUT indicates a sample is to be taken, the ADC digitizes that pixel, and places it into the FIFO.

The FIFO has synchronization signals which allow it to tell the processor of its status, such as whether it is empty, or full. Using this, we know when a full frame has arrived.

4. Helicopter Application

We have conducted experiments in using vision to navigate to the source and destination rings. The RCV is a prime candidate for this task, as it is low powered, and

A Reduced Complexity Vision System for Autonomous Helicopter Navigation

Parag H. Batavia[*], M. Anthony Lewis[+], George A. Bekey[*+]

[*]Department of Computer Science
[+]Department of Electrical Engineering
Institute for Robotics and Intelligent Systems
University of Southern California
Los Angeles, CA 90089-0781

Abstract

Many current avenues of vision research involve fully analyzing an image with expensive, high powered computers. This approach has major implications in terms of cost, size, and power consumption. Other methods have involved sub-sampling an image to reduce cost and complexity. This has the disadvantage of information loss. We present a low cost, low powered, reduced complexity vision system capable of intelligently sampling an image to reduce this information loss. The design philosophy and methodology is discussed, along with sample applications. Primarily, we demonstrate how the reduced complexity vision system will be used to aid in navigation of an autonomous flying vehicle. This is quantified by showing how having multiple sampling schemes result in increased robustness and accuracy of our helicopter line tracking algorithm.

1 Introduction

In this paper, we present a low cost, Reduced Complexity Vision system (RCV). This system is capable of arbitrarily sampling an image, with the goal of intelligently reducing the amount of information required to analyze an image.

An average image of 512*512 pixels contains a quarter megabyte of data which needs to be analyzed. Much of this information is redundant, and processing it takes a significant amount of effort that may not be required.

Many current approaches which attempt to analyze full video images at 30 frames per second require either very expensive systems such as Warp machines in the CMU Navlab (Ross, 1993), or expensive, complicated custom hardware (Van der Wal, 1992).

There are a number of previous approaches which sub-sampled images, either through custom hardware, or special purpose integrated circuits. Some of these systems were developed by Bederson(1992), Burt(1988), and Horswill(1993), which will be discussed later. While this is an improvement in cost and complexity over transputers and Warp machines, there are limitations to these approaches as well. These approaches used a relatively fixed or highly regular sampling strategy.

This results in our primary motivation for this work. We want to achieve the savings in power, cost, and complexity of sub-sampled systems, while retaining the flexibility of higher cost processors as much as possible. A system of this type would have many applications in the field of active vision. Some of the primary interests of active vision are attention mechanisms, spatially variant sensing, and real time tracking (Blake, 1992).

The RCV attempts to tackle these issues by combining the flexibility of high end vision systems with the low cost and low power advantages of sub-sampled systems. This is done by building a vision system which costs less than $500, and can perform low level vision tasks in real time. The real time processing was accomplished in two ways, by limiting the amount of data that can be processed per frame, and by providing a mechanism in which intelligent sampling choices can be made to maximize the amount of information present in the reduced sample.

To accomplish this goal, a concept called *dynamic foveation* was developed. Dynamic foveation allows the user of the system to change the sampling scheme of the device on the fly. For example, when initially surveying a scene, a sparse sampling of the entire field can be taken. This sub-sampled image, which at most can be 4Kbytes of data (generally a 64 by 64 pixel array), can be quickly analyzed for any areas of interest.

When an area of interest is found, it can be fixated on and analyzed in more detail. Simultaneously,