

52-63
108
188122

Knowledge-Based Vision for Space Station Object Motion Detection, Recognition, and Tracking

P. Symosek, D. Panda, S. Yalamanchili, and W. Wehner III
Honeywell Systems and Research Center
Minneapolis, MN 55418

HY 989092

Abstract

Computer vision, especially color image analysis and understanding, has much to offer in the area of the automation of Space Station tasks such as construction, satellite servicing, rendezvous and proximity operations, inspection, experiment monitoring, data management and training. Knowledge-based techniques improve the performance of vision algorithms for unstructured environments because of their ability to deal with imprecise a priori information or inaccurately estimated feature data and still produce useful results. Conventional techniques using statistical and purely model-based approaches lack flexibility in dealing with the variabilities anticipated in the unstructured viewing environment of space.

Algorithms developed under NASA sponsorship for Space Station applications to demonstrate the value of a hypothesized architecture for a Video Image Processor (VIP) are presented. Approaches to the enhancement of the performance of these algorithms with knowledge-based techniques and the potential for deployment of highly-parallel multi-processor systems for these algorithms are discussed.

1.0 Introduction

A major consideration in the design and deployment of the NASA Space Station is the definition of automation techniques which will guarantee the timely and reliable performance of the Space Station's missions. During specification of the initial design of the Space Station, NASA has identified three criteria for the justification of the development of an automation technique:¹

1. The automation capability should be of substantial value toward the objective of accomplishing Space Station functions, such as user experiment monitoring, user production activities and satellite servicing in a timely and reliable manner.
2. The safety of the crew must not be compromised.
3. The Space Station should operate autonomously with as little support from ground-based facilities as possible.

A Video Image Processor will be a very valuable automation tool on-board the Space Station for several reasons: Image processing, specifically the identification of the objects seen in the image and the formulation of a 3-dimensional model of a scene, is a pre-requisite capability for the development of autonomous robots. These autonomous robots could perform many of the mundane tasks such as experiment monitoring and proximity operations that at the present time require crew member supervision. Image processing is also a pre-requisite capability for the task of bandwidth reduction which will be necessary for the Space Station because of limited on-board storage and the restraints of secure channel downlinks from the Space Station to ground-based facilities. For semi-autonomous implementations, image processing is employed to execute repetitive tasks such as color image enhancement/restoration or operator cueing, and an operator is required only for verification confirmation of the actions of the algorithms. A Video Image Processor can perform each of the preceding tasks, thus increasing the efficiency of crew members of the Space Station.

A Video Image Processor is a dedicated processing unit for image data that is modularly extendable and is to be built from commercially available components. The VIP architecture was defined conceptually under contract to NASA by Honeywell Systems and Research Center on the basis of several criteria, which are: maintainability, extensibility, programmability, physical aspects of deployment and the performance specifications defined by current Space Station applications². The candidate architectures for the VIP were quantitatively evaluated with architecture analysis tools to obtain a high degree of confidence in achieving the desired functionality. To do this a set of example image processing algorithms had to be specified and their performance evaluated for imagery acquired during previous Space Shuttle missions to simulate the algorithms' behavior under realistic conditions. In this way, the processing requirements of the algorithms could be estimated for the unique set of environmental, lighting and imaging constraints found in space.

The goal of the selection process was to develop an algorithm suite that would benefit a sufficiently large number of space station tasks. The various space station tasks that benefit from an image processing capability can be classified into eight generic categories:³

- Construction
- Satellite servicing
- Rendezvous and proximity operations
- Inspection
- Payload delivery and retrieval
- Experiment monitoring
- Data management and communication
- Training

In order for the VIP to assist in the automation of these tasks, it should have a substantial array of image processing algorithms that it can apply in accordance with the changing demands of the application. These image processing algorithms can be grouped into six major families for the space station scenario:

- Color image enhancement
- Tracking
- Surveillance
- Identification

- Proximity operations
- Bandwidth reduction

These six families do not represent the entire breadth of the state of the art in image processing, but most of the image processing algorithms required for the automation of space station tasks belong to one of these families. In addition, algorithms in each of these categories are sufficiently mature for the design and build of a prototype system.

A cross-reference of Space Station algorithmic functions and Space Station tasks is presented in Table 1. There are several important observations to be made from Table 1. Most important, it is apparent that color image enhancement is required for all of the selected Space Station tasks.

This can be attributed to two factors. First, imaging systems are not perfect and provide noise-degraded images even under the best conditions. Imaging systems that employ electron-beam scanning mechanisms for transforming the optical image into an analog signal always exhibit random noise in their outputs. This noise is caused by thermal and electrical noise in the imaging system hardware. When an analog signal is transformed into a digital array of image intensities, there is additional noise superimposed onto the image because of the finite response time of the digitization amplifiers.

Even if the image's quality is very good, which usually will be the case if the imaging system is built with charge-coupled devices (CCD), the performance of the image processing algorithms can be improved by attenuating the remaining noise with color image enhancement algorithms. The reason that noise may remain even for a good imaging system is that the images being used are real-world images. Objects in the real world will always exhibit small perturbations in reflectivity because of variations in surface smoothness, and therefore images of these objects will always appear to be noisy. Color image enhancement algorithms can sufficiently attenuate the unwanted noise information so that its magnitude will be below the detection thresholds of the image processing algorithms.

Generic Tracking algorithms are also a pre-requisite capability for all of the Space Station tasks considered. This can be explained as follows: Scene interpretation algorithms can be decomposed into four stages:

1. Segmentation
2. Feature Detection
3. iconic (Pixel-Based)-to-Symbolic Feature Mapping
4. Classification

The algorithms that together form the Tracking function are equivalent to the first three stages described above. Therefore, an evaluation of the characteristics of an architecture for the Tracking functions defines a dependable measure of the performance of the architecture for most high-level image processing applications. Also, the partitioning of the image into regions of interest and extraneous background is the initial phase of all image processing algorithms that are designed to obtain symbolic information from raw image data.

Of the remaining algorithm categories, Bandwidth Reduction was chosen for verification because 1) It is a pre-requisite for five of the eight Space Station tasks considered and 2) The algorithms that perform the Bandwidth Reduction function are exactly those that are required for the Tracking function, with the exception the temporal silhouette matching algorithm required for Tracking.

Knowledge-based techniques are a means of employing the efficient symbolic pattern matching and high-level reasoning capabilities of artificial intelligence for image interpretation applications. Knowledge-based techniques for region labeling can tolerate large errors in feature data and still produce meaningful results. They can be designed in stages because of their modular rule database: as new contexts are discovered for classification of features, the system is reconfigured by the definition or modification of a few rules. Knowledge-based systems are very efficient for the task of performing retrieval operations on large symbolic databases on the basis of relational and contextual constraints on the data.

Due to the unpredictable nature of imagery obtained in space, especially during construction of the Space Station itself with remotely guided robots, and other factors unique to a space environment such as rapid diurnal changes while a vehicle is in orbit, knowledge-based techniques will play a very important role in improving image interpretation algorithm performance. Knowledge-based techniques have been applied extensively for all of the four stages of image interpretation algorithms. These systems have been used for photointerpretation applications,⁴⁻⁶ autonomous weapon delivery systems,⁷ and the labeling of features in arbitrary urban scenes.⁸⁻⁹ These expert systems have several features in common: A database of calculated image features is matched with predicates of production rules, which are represented as logical statements of the form "If ..., then..." and a control system that supervises rule activation.

The system developed by Nagao and Matsuyama⁵ uses a knowledge base representing relational, contextual and geometric constraints for the task of region labeling for multi-spectral imagery obtained from low-flying aircraft. The region boundaries are detected by a variety of low-level image segmentation algorithms and the resultant information is archived on a blackboard shared by each of the experts of the system. Each expert is optimized for locating a specific kind of object or region. They devised an approach for the reliable classification of vegetational regions that is independent of the time of year, using the ratio of two distinct spectral bands to discriminate the vegetation regions from the non-vegetation regions. They demonstrate that knowledge-based techniques permit the reliable identification of houses and roads in congested urban scenes where other classical approaches normally fail.

Ohta⁹ developed a hierarchical region labeling scheme for color images of urban scenes. The approach is hierarchical because an initial plan image is derived and labeled before a more detailed, data-directed segmentation is carried out. The plan image is defined by a region-based color image segmentation algorithm. The macro-level regions of the plan image are: sky, tree, building and road. These region categories are detected using top-down contextual and spectral constraints. The algorithm is very reliable and can correctly label regions in urban outdoor scenes using only 57 rules.

However, very little work has been done in the area of the application of knowledge-based techniques for tracking or motion understanding. This is primarily due to the unconstrained nature of the problem. It is exceedingly difficult to specify an expert system that can characterize the dynamic behavior of arbitrary objects as they translate and rotate in three dimensions. Work has been done on dynamic environment understanding for a mobile robot employing an intelligent system for reasoning about the three-dimensional structure of a

stationary environment¹⁰. This approach uses path planning techniques for the identification and avoidance of obstacles detected by sonar sensors. The technique was successfully demonstrated for the task of obstacle avoidance while navigating an M113 autonomous vehicle, built by FMC Corporation for the DARPA ALV program, through a maze of obstacles. However, no procedure is specified for the detection and interpretation of sensor data that results from moving objects.

Honeywell is carrying out research on the utilization of knowledge-based techniques for the discrimination of moving objects from stationary objects in imagery obtained from a moving autonomous vehicle's camera.¹¹ This approach, which we have labeled Dynamic Reasoning from Integrated Visual Evidence (DRIVE), identifies visual cues from a sequence of images that defines a global dynamic reference model. Object recognition, world knowledge and the accumulation of evidence are used to disambiguate the situation and refine the global reference model.

The design principles for the identification of the optimal architecture for the Video Image Processor were reviewed in this section. The figures of merit for the design were stated and the approach utilized to validate the design for space imagery were summarized. The rationale for selecting the three image processing algorithms which were studied was explained. The next section discusses the technical details of the three image processing algorithms and specifies approaches for executing the same algorithms with knowledge-based techniques. Section 3 summarizes approaches that may be pursued for the development of image interpretation systems that employ knowledge-based techniques. The fourth section describes experimental results obtained for the algorithm validation task for the Video Image Processor. Section 5 discusses a few important problems that are yet to be solved and that can produce significant increases in algorithm efficiency and reliability for image processing in a space scenario.

2.0 VIP Algorithms

The details of the implementations of the three image processing functions chosen for the establishment of space processing requirements for the VIP are discussed in this section. Special attention is paid to citing how the performance of each algorithm is enhanced with knowledge-based techniques.

2.1 Color Image Enhancement--The following is a description of three of the algorithms that were evaluated for Color Image Enhancement. Each algorithm is designed to restore a particular feature of the color images, e.g., dynamic range, sharpness, etc. It is therefore conceivable that an actual implementation may use combinations of these algorithms to produce imagery with specific color characteristics.

2.1.1 Color Image Balanced Histogram Equalization--Color image balanced histogram equalization enhances image contrast and increases image dynamic range. The algorithm operates with the same fundamental principle that monochrome image histogram equalization employs, namely, that the gray levels of the original image are redistributed so that the histogram of the transformed image will take the form of a uniform distribution across a specified range of gray levels. This range is usually the display range of the display device. The mapping is one-to-one; thus, for each gray level of the original image, every pixel that appeared with that gray level will appear with a unique gray level in the transformed image. However, multiple gray levels from the original image can map to a single gray level in the transformed image.

For color images, histogram equalization is not a computationally simple process because of the requirement that the hue of each region remains the same before and after histogram equalization. To meet this constraint, the color image balanced histogram equalization algorithm calculates the equalization mapping for the intensity image, where the intensity image is obtained as the average of three primary images. The transformed primary images are calculated from the histogram-equalized intensity image. In this manner, the hue of each region of the image remains constant. The algorithm operates as follows. The offsets of the color image intensities from the average intensity level are calculated, and the transformed color levels are calculated as the transformed intensity level plus the original offsets. For example, consider a single pixel. If the three original images' intensity values were red = 140, green = 150, and blue = 110, then the average intensity at that point is $I = 133$. If the mapping derived by histogram equalization was $133 \rightarrow 175$, then the output color levels for that location are red = 182, green = 192, and blue = 152.

Images transformed with this algorithm will exhibit full dynamic range, and the hues of the regions of the image will not change. This may be shown as follows. A three-channel color image can be equivalently represented by an HIS image, where HIS stands for hue-intensity-saturation. The hue image represents the color of the regions of the original color image, where the magnitude of the hue is proportional to the percentages of the three primary colors, red, green, and blue. The intensity image is simply the arithmetic average of the three color images. The saturation image represents the strength of the color. The range of the hue image is 0 to 359, the range of the intensity image is 0 to 255, and the range of the saturation image is 0 to 1. (The hue and intensity images can be archived as integer arrays, but because the range of the saturation image is 0 to 1, it is archived as a real-valued array.)

For the HIS color space, the hue is calculated as a function of the ratio of linear functions of the three color image intensities. Specifically, this function is

$$\text{Hue} = \cos^{-1} \left[\frac{\frac{1}{2} [(R-G) + (R-B)]}{\sqrt{(R-G)^2 + (R-B)(G-B)}} \right]$$

where R, G, and B are the red, green, and blue intensities. If $B > G$, then the hue = 2π -hue. When $R = G = B$, the hue is undefined.

Let the three original chromatic levels at each pixel be represented as $R = I + \Delta_R$, $G = I + \Delta_G$, and $B = I + \Delta_B$, where I is the intensity value of the pixel in the original image. Let the output color levels be $R' = I' + \Delta_R$, $G' = I' + \Delta_G$, and $B' = I' + \Delta_B$, where I' is the intensity value of the pixel after transformation. Because $R-G = R'-G'$, $R-B = R'-B'$, and $G-B = G'-B'$, the magnitude of the hue is unchanged by the transformation. Therefore, the color information that was present in the original scene, but was not discernable because of the low dynamic range of the image, is preserved. This characteristic of the algorithm will guarantee that the transformed image is a good representation of the original scene because its colors are faithfully reproduced.

A knowledge-based approach for adaptive Color Image Balanced Histogram Equalization would use a measure of an image's contrast in a local region to determine whether the dynamic range was low (perhaps caused by shadowing) and apply the algorithm to that region. The algorithm could employ information obtained from previous frames' processing results to assist in the identification of shadows.

2.1.2 Color Image Accentuation--Color image accentuation is a process whereby the image's sharpness is augmented by increasing the saturation of the image. Color saturation may be increased as follows. The offsets of each of the original image intensities from the intensity image values are calculated, and each of these offsets is amplified by a factor K, where $K > 1$.

We can represent the quantities MAX and MIN (where MAX and MIN are the maximum and minimum of the three primary image intensities):

$$\begin{aligned} \text{MAX} &= I + \Delta_{\text{MAX}} \\ \text{MIN} &= I + \Delta_{\text{MIN}} \end{aligned}$$

The magnitudes of MAX and MIN, after transformation, are defined as

$$\begin{aligned} \text{MAX}' &= I + K \Delta_{\text{MAX}} \\ \text{MIN}' &= I - K \Delta_{\text{MIN}} \end{aligned}$$

The transformed image's saturation is

$$\begin{aligned} S &= \frac{\text{MAX}' - \text{MIN}'}{\text{MAX}'} \\ &= \frac{I + K \Delta_{\text{MAX}} - I + K \Delta_{\text{MIN}}}{I + K \Delta_{\text{MAX}}} \\ &= \frac{\Delta_{\text{MAX}} + \Delta_{\text{MIN}}}{\frac{I}{K} + \Delta_{\text{MAX}}} \end{aligned}$$

As $K \rightarrow \infty$, the term I/K in the denominator will decrease, thereby causing the saturation to increase. With the constraints that Δ_{MAX} is positive, Δ_{MIN} is positive, $K\Delta_{\text{MIN}} \leq I$, and $K\Delta_{\text{MAX}} \leq 255 - I$, the maximum value that S' can attain is 1, as $K \rightarrow \infty$. It can be seen from the following arguments that the hue of each region is unchanged. The color levels of the image after accentuation can be represented as $R' = I + K\Delta_R$, $G' = I + K\Delta_G$, and $B' = I + K\Delta_B$. The magnitudes of each of the subtracted pairs-- $R' - B'$, $R' - G'$, and $G' - B'$ --are equal to K times the magnitude of the respective subtracted pair before accentuation. When these values are used to calculate the hue for a specific pixel of the image, the factor K can be brought out of the numerator and the denominator, which means that the magnitude of the hue component will not change. Therefore, this algorithm also faithfully represents the hue of the original image.

This technique may be employed to increase the saturation of colors of each of the regions of the image. The effect is to make small surface detail more distinguishable.

2.1.3 Constrained Inverse Filtering--Constrained inverse filtering is a technique whereby degradations of the imaging process, such as a dispersing medium between the imaging system and the object of interest or out-of-focus optics, are corrected with digital signal processing. Constrained inverse filtering is effective when the point spread function (PSF) of the distorting medium or the imaging system optics is known or can be estimated fairly accurately.

Constrained inverse filtering is a specific form of inverse filtering. It is a restoration technique that attempts to invert the effects of an optical transfer function on an image. Inverse filters are implemented to minimize the sum of squared errors between the original image and the restored image for a specific model of the image formation process. Constrained inverse filters attempt to minimize the same error function, with a constraint that the norm squared of the restored image is as small as possible. This constraint is applied to prevent the noise present in the observed image from appearing at too great a level in the restored image. The model of the image formation process employed is

$$g(t, w) = h(t, w) * f(t, w) + n(t, w)$$

where $g(t, w)$ is the observed image, $h(t, w)$ is the PSF of the degradation function, $f(t, w)$ is the original, undistorted image, $n(t, w)$ is 0-mean, white Gaussian noise process, and $x * y$ represents the two-dimensional convolution of x and y .

The discrete representation of the image is obtained by sampling $g(t, w)$ at a set of points on a Cartesian grid: $t = kT$; $k = -1/2, \dots, 1/2 - 1$; $w = lT$; $l = -1/2, \dots, 1/2 - 1$. Let $g_{i,j}$ ($0 \leq i \leq I-1$; $0 \leq j \leq J-1$), $f_{i,j}$ ($0 \leq i \leq I-1$; $0 \leq j \leq J-1$), $h_{k,l}$ ($0 \leq k \leq K-1$; $0 \leq l \leq L-1$), and $n_{i,j}$ ($0 \leq i \leq I-1$; $0 \leq j \leq J-1$) represent the discrete measurements of the observed image, the original image, the PSF, and the additive noise, respectively. The model equation for discretized images is

$$g_{i,j} = \sum_{k=0}^{K-1} \sum_{l=0}^{L-1} h_{k,l} f_{i-k, j-l} + n_{i,j}$$

If the dimensions of the spatial images, $\{g_{i,j}\}$, $\{f_{i,j}\}$, and $\{h_{k,l}\}$, are the same, we can transform the previous equation into a frequency domain representation. Let N and M be the row dimension and column dimension, respectively, of the three images. These dimensions can be selected arbitrarily, but the values selected should be greater than $J+L-1$ and $I+K-1$ in the horizontal and vertical directions, respectively, to prevent the convolution window from extending off the image in the previous equation. They can also be selected as a power of 2, so that efficient implementations with the Fast Fourier Transform (FFT) are possible.

If Fourier Transforms of the observed image $\{g_{i,j}\}$, the restored image $\{f_{i,j}\}$, and the PSF $\{h_{i,j}\}$ are defined as $G(u,v)$, $F(u,v)$, and $H(u,v)$, respectively, then the frequency domain representation of the constrained inverse filter is:

$$\hat{F}(u,v) = \frac{H(u,v)^*}{H(u,v)^* H(u,v) + \gamma} G(u,v)$$

where $H(u,v)^*$ is the complex conjugate of $H(u,v)$, and γ is an arbitrary constant that controls the magnitude of the norm squared of the estimate of $\{f_{i,j}\}$.

2.2 Tracking and Bandwidth Reduction--The tracking algorithm operates on a monochrome image to detect man-made objects in the field of view and track them over multiple images. The major elements of the tracking algorithm are multithreshold segmentation, boundary tracing, linearity filter, connected component analysis, and silhouette matching. Multithreshold segmentation is used to identify regions of the image with relatively constant intensity. A boundary tracing algorithm produces boundaries of regions that are passed through a linearity filter to determine which regions contain straight edges identifying them as part of a man-made object. Once the man-made pieces are assembled into objects by connected component analysis, tracking is performed by the silhouette matching algorithm, which compares silhouettes of objects in successive images to determine relative motion. Figure 1 is a data flow diagram of the tracking and bandwidth reduction algorithm.

The following subsections describe the algorithm for each component function of the tracking algorithm. Due to the large degree of commonality between tracking and bandwidth reduction, the latter is discussed here in subsection 2.2.8.

2.2.1 Window Average--The window average is a simple data-independent operation transforming the original intensity image into a smoothed intensity image. Its function is to help remove sensor noise in the image, ensuring that distinct regions in the image have relatively constant intensity. The operation computes a new value at each pixel position by averaging the input intensities of pixels in a window about the point. A 5×5 window was found to work well on test imagery.

2.2.2 Monochrome Segmentation--Monochrome segmentation divides the smoothed input image into different regions, where each region is characterized by a "nearly" uniform gray level. Labels are selected to represent different intensity ranges, and each region is appropriately labeled. This is accomplished in three major steps. First, an intensity histogram of the image is computed. Second, this histogram is searched for local minima and maxima, which define the intensity ranges corresponding to interesting regions in the image. Each distinct range is assigned a label and an upper and lower threshold defining the range. The third step is to apply the thresholds to the image, replacing the value at each pixel with the appropriate label. A list containing the labels and the average intensity values of their corresponding regions is also output for use by the bandwidth compression algorithm.

2.2.3 Boundary Tracing--Boundary tracing is a data transformation algorithm whereby encoded region boundaries are obtained from a labeled image. The encoded region boundaries (silhouettes) are much more compact than a complete image, requiring about four bits to store each image pixel that is on the boundary of a region. Conceptually, the algorithm interrogates each pixel of the image in an orderly fashion. At each point, the current pixel is examined to determine if it is on the boundary of a region that has not been traced yet. If so, the algorithm traces the boundary, ending at the same pixel where it began. The algorithm proceeds to examine the next point in search of more regions. When every image point has been examined, all of the regions have been traced.

2.2.4 Linearity Filter--A linearity filter is applied to the set of silhouettes. This filter computes a measure of linearity for each silhouette to determine if they correspond to man-made objects. The output is a binary image with nonzero values at boundary points of regions whose silhouettes were relatively linear.

The computation of the linearity filter is illustrated in Figure 2. Using a sliding window of width w , the angles s_1 and s_2 formed by the current point and the endpoints of the window are determined. The difference between these two angles is the curvature. For each silhouette, the linearity measure is the average curvature of all boundary points. If this measure exceeds a predetermined threshold, the boundary points of the region are set in the output binary image, indicating that the silhouette corresponds to a man-made object.

2.2.5 Connected Component Analysis--Connected component analysis is a general-purpose function that identifies connected sets of pixels in an image. Each connected set is identified with a label in an output labeled image and a location in an output feature file. In this instance, connected component analysis operates on the binary image output from the linearity filter to determine which segmentation silhouettes (and therefore regions enclosed by these silhouettes) belong to the same object. Since different components of an object may have different intensities, an object may be segmented into several adjacent regions. Since the regions are adjacent, they can be grouped into one component by connected component analysis.

2.2.6 Trace Components--This algorithm uses the starting locations and the labeled image output from connected component analysis to trace the boundary of each component (which in this case is a single target). Because the starting locations are known, this algorithm is considerably more efficient than the boundary tracing algorithm described in subsection 2.2.2. Thus, in this algorithm, it is not necessary that each image point be visited; however, the same method for tracing a region boundary is used.

2.2.7 Fast Silhouette Matching--Fast silhouette matching compares the silhouettes of targets found in the current frame to stored silhouettes of targets tracked in previous frames. Depending on which new targets match which previous targets, the tracked target information is updated, and any new targets are added to the track list. To match new and previous targets, each new silhouette must be compared to each old silhouette. The match scores are used to determine which new and previous targets correspond.

To compare one new silhouette with one previous silhouette, the (x,y) translation is determined, which maximizes the number of coincident boundary points of the two silhouettes. In theory, this is done by considering each possible (x,y) translation and counting the number of coincident boundary points. In practice, each point of the previous silhouette is considered as the point that ideally matches the first point of the target silhouette. Then the two silhouettes are traversed simultaneously, incrementing counters in a two-dimensional histogram. Each counter corresponds to the (x,y) translation necessary to make the new and previous boundary points in question correspond. On completion, the histogram is searched for the maximum. The maximum value is compared to the length of the silhouettes to determine the accuracy of the match. Essentially this is a variation of the Hough Transform. Choosing corresponding starting points is equivalent to making a hypothesis about the relative motion between images. Incrementing a counter is equivalent to "voting" for a particular (x,y) motion vector. The peaks correspond to the most likely motion vectors. The actual motion vector generally garners the most votes. The advantage of such a technique is its robustness and relative insensitivity to noise.

2.2.8 Bandwidth Reduction--This is a very simple operation that combines several intermediate tracking results to be transmitted to a remote location for storage or viewing. The essential information from the image is the target signature, complete with as much detail of the target itself as possible. The region silhouettes produced by the boundary tracing algorithm are processed by the linearity filter and marked indicating whether that silhouette passed the linearity filter or not. Then the bandwidth reduction operation can look at each region silhouette and select the ones that were considered part of a man-made object.

3.0 Knowledge-Based Image Interpretation Concepts

Knowledge-based techniques for image interpretation are more robust than conventional techniques because they can identify symbolic image features on the basis of incomplete or imprecise information obtained from the image. Classical techniques, in general, detect objects or specific features of objects from images on the basis of the degree of match between the actual features and a fixed, a priori model of the features. When the degree of mismatch is sufficiently great, as determined by composing the magnitude of a degree of match metric to a threshold, the algorithms reject the conclusion that the feature was observed. However, knowledge-based techniques do not categorically reject the hypothesis; they associate a "confidence factor" with the hypothesis transfer the positive and negative evidence obtained to date for that feature to a database. If evidence is found that either confirms or refutes the existence of the specific feature, the database can be revised. It is this characteristic of knowledge-based systems that makes them invaluable for image interpretation in unstructured environments, such as the Space Station construction environment. These systems have the capability of deriving conclusions from imprecise or conflicting sources of information and maintaining the history of deductive steps applied to reach those conclusions to permit optimal utilization of all available information at any single instant of time.

There are four generic categories of knowledge-based scene interpretation algorithms, which are:

1. Scene Labeling
2. Temporal Resolution
3. Context-based Resolution
4. Knowledge-based Feedback Control for Resegmentation

Each of these techniques will be explained in succeeding paragraphs.

3.1 Scene Labeling--The reliability and accuracy of each of the image processing functions tabulated in Table 1 will be enhanced with an understanding of the scene context for the image being evaluated. The context is deduced from a set of labels applied to the scene by a scene labeling algorithm.

Honeywell has developed a Reasoning Region Classifier (RRC)¹² to identify and test the knowledge pertinent to each of a specific class of regions. RRC is a production rule system, with explanation facilities, whose goal is to characterize image sub-regions of interest, based on vision system observable features, such as region uniformity, texture smoothness, topological features, etc. This system is currently implemented for the classification of man-made and natural objects in air-to-ground imagery, but it could be easily modified to discriminate objects for the space scenario. The model of the scene structure is a hierarchical database, which has the label "entire scene" at the root, and is subdivided at each level of the hierarchy as the classification of scene objects becomes more specific. The search for the true classification of a specific region or object is performed on the subtree which has the highest confidence level based on the production rules.

3.2 Temporal Resolution--Temporal Resolution is a technique for the resolution of conflicts that result from region classification for region labels. It consists of the following steps:

1. Identify a sequence of frames which have been segmented into regions each of which display a region in the neighborhood of the candidate region R.
2. Determine whether the classifier result on the candidate region, say R, in the present frame, is consistent with the classifier results on the portion of the image corresponding to this region in past frames of the sequence.
3. Otherwise, modify the classifier result R by multiframe decision smoothing.

3.3 Context-Based Resolution--Context-based resolution conflict removal combines region information and relational context information from the current scene, for modifying classifier decisions that are inconsistent with the world model as represented in the hierarchical database.

Conflict removal is performed by detecting inconsistent configurations in the scene. The production rules that are used by the context-based resolution technique are based on a priori world knowledge.

3.4 Knowledge-Based Feedback Control for Resegmentation--A scene is composed of multiple regions that have different sizes and shapes. A single segmentation algorithm may not suffice to properly segment all these regions. Appropriate choices of segmentors based on the ancillary information are crucial. Further, each algorithm has an associated set of parameters. Proper setting of the values of these parameters has a major impact on the algorithm regardless of how robust it is. For example, a large window size in noise smoothing techniques can blur the edges between two regions, thus, resulting in an erroneous region classifications. In this case, adaptive thresholding can remedy the problem.

There are two types of knowledge-based control methodologies that can perform equally well under variations of scene characteristics encountered in Space Station scenario.

3.4.1 Open-Loop Control--The first type of control is called open-loop knowledge-based control. This is a process that directs the low-level processing through proper image operator selection and the associated parameter value selection. Open-loop control processing governs simple data reduction tasks such as noise removal and region of interest selection.

The Open-Loop Control scheme derives the process goal from the information stored in Short-Term Memory (data obtained from the current image) and the knowledge base. The process goal is usually based on temporal and ancillary information such as previous frame processing results, the lighting conditions and a priori information for the radiometric and topological characteristics of the current scene. Rules in the knowledge base are used to derive the process goal. An example of process goal derivation is:

IF (mission goal IS (satellite detection) IN (high clutter area))
THEN ((locate region with two parallel linear borders) AND (remove high frequency noise))

Based on the derived processing goal, the selection control identifies the proper image operators with their associated values. The selection is also generated by the rules stored in the knowledge-base. An example of the selection rules is:

IF (remove high frequency noise)
THEN RUN (window average routine)

The process module then executes the selected image operators with the given parametric values. The output of the process is passed directly to the next low level process. The process module can also generate some knowledge, such as the image contrast, which can be used by other processing modules. This generated knowledge is fed into the knowledge-base for subsequent use.

3.4.2 Feedback Control--The second type of control is called the feedback knowledge-based control process. It is designed for governing complex low-level processes such as segmentation and color image enhancement processes.

Similar to the open-loop control process, feedback control derives the process goal from the knowledge in the knowledge-base and short-term memory, and selects the appropriate image operators and their parametric values. Then, the image operators are applied to the image and the results are passed to a process evaluation module. The process evaluation module determines the next processing step. The evaluation module either: 1) accepts the output and passes it to the next processing module, 2) feeds the image back to the same process module, recommending different image operators or parameter values for more refined processing, or 3) rejects the results and bypasses the process module.

The evaluation decision is also based on rules and information acquired from the scene stored in the knowledge-base. An example of a region evaluation rule is the following:

IF (Goal Size = small) AND (Goal Shape = rectangular)
AND (Region Size = large) AND (Region Shape = rectangular)
THEN (Resegment region with a lower threshold)

These knowledge-driven control processes make the best use of all available information about the scene. Each processing module can achieve the best possible performance in satisfying the processing goal. Therefore, a high performance scene analysis system can be developed by synergistically integrating the low level processing results.

4.0 Architecture Analysis for Parallelized, Multi-Processor Implementation of Knowledge-Based Algorithms

Previous sections of this paper have briefly touched on the key Space Station tasks which can benefit from knowledge-based vision processing. Details of specific Space Station vision functions and their implementation have also been discussed. In this section, we overview key architectural issues in developing a hardware architecture and software methodology for implementing these vision functions.

Developing real-time architectures for imaging systems is acknowledged as a difficult problem in many respects and remains a highly active research area. The key issues include: how to attain necessary and sufficient performance; how to program and maintain real-time systems; whether to use homogeneous or heterogeneous hardware; how to integrate processors with the environment; and how to develop planned/evolutionary approaches based on standards. A general solution to these central issues does not exist. Instead they must be revisited for each new application considered.

Statements of hardware performance requirements and capabilities usually are given simply in terms of the millions of operations per second (MOPS) needed for a set of functions or available from a system. A more critical measure of system performance would look at: operations per second (OPS) as a function of algorithmic requirements; power requirements; physical size and weight; and cost. In short:

$$\text{Performance Measure} = \text{OPS}(\text{algorithm}) / \text{Watt cm}^3 \text{ } \$$$

Because transportation costs and limited space and weight budgets are key drivers in Space Station construction, the key elements of this metric should be throughput as a function of algorithm performed and total volume required to achieve this throughput. Weight and power are typically correlated to volume for a given technology, and the desire is always to minimize cost consistent with achieving functionality. Typically, computer architects design systems in an attempt to keep functional units (e.g., arithmetic logic units or multipliers) maximally busy because algorithmic performance requirements are specified in terms of the number of adds, multiplies, etc. Applying this approach to image processing architectures leads to designs in which 90%+ efficiency is achieved but on only 2-5% of the total processor hardware. Maximizing the throughput-to-volume ratio leads to more compact systems in which functional units are not necessarily fully utilized and is a logical approach for signal and image analysis architectures destined for the Space Station.

The tradeoff between using a heterogeneous or homogeneous processor architecture is a crucial tradeoff for any image processing system. The tradeoff is driven by algorithmic requirements as well as issues of system expandability, programmability, and flexibility. Current robust algorithmic paradigms for imaging systems subdivide the processing steps into various categories. An image understanding paradigm which has been useful in developing computer architectures is shown in Figure 3. This paradigm categorizes algorithmic functions according to data structures and processing functions. It is straight-forward, practical, and robust to directly translate such paradigms to hardware systems as indicated in Figure 4. Such an approach leads by nature to a heterogeneous architecture and from experience tends to minimize system volume. In general, more specialized hardware modules lead to a more compact system, but maximizing the throughput-to-volume ratio in this fashion must be balanced with expandability, programmability, and flexibility requirements in the Space Station application.

Two aspects of programmability become issues for real-time image processors. First, image processing hardware must be designed to utilize a high degree of parallelism at all levels to achieve high performance. The software methodology and tool set must provide adequate means to deal with parallelism and must bridge the gap between coarse grained high-level languages (e.g., Ada, FORTRAN, Pascal, etc.) and fine-grained machine languages (e.g., microcode). Any inefficiency in the translation or compilation process directly impacts the total system hardware requirement. Second, a software methodology intended for use with heterogeneous architectures must support all processor types in an integrated fashion. These are especially important issues in the Space Station setting where software development and maintenance costs will likely be the dominant portion of total imaging system cost.

The application environment affects imaging system architecture in many important ways. In the Space Station environment, factors such as fault tolerance and recovery, reliability, and testability are clearly important to safe and effective use of any mission critical computing equipment. In addition to these more or less generic considerations, very specific design details can be influenced by the environment. For example, electrical and radiation induced noise effects of the space environment lead naturally to consideration of optical interconnect for high data rate sensor channels. It is also logical to consider performing sensor specific preprocessing functions local to the sensor to reduce or eliminate channel induced noise. A broader environmental issue is the type and number of video sensors which can be active simultaneously. An architecture is needed which can readily switch between sensors and sensor types.

A final high-level issue with significance to the Space Station application is the ability of the selected architecture to adapt in an evolutionary fashion to evolving mission requirements. To achieve such an adaptation capability requires a so-called "open" architecture in which modules may be added or replaced. Designing an open but heterogeneous architecture is difficult in that each element of the architecture brings specialized interconnection, software, and other requirements. Maximum use of standards is a necessity to successfully developing such an open architecture.

A specific image processor implementation for Space Station applications has been developed and is reported in a companion paper [13]. This Video Image Processor (VIP) design is based on careful consideration of the broad issues discussed above and on the specific requirements of the image processing tasks and algorithms discussed in earlier sections of this paper. Over 150 architectural variations were analyzed using advanced computer modelling techniques. The result, illustrated in Figure 5, is a two-level architecture using special purpose high-performance image pixel processing hardware operating in a pipelined fashion combined with a distributed shared-memory multiprocessor. These two levels perform the image frame processing and combined region and symbolic processing functions from the taxonomy of Figure 3. The relatively low update rates specified for VIP allow the array processing and general purpose computing functions of Figure 4 to be combined in the multiprocessor.

Although the VIP architecture satisfies the essential processing requirements for the knowledge-based vision algorithms previously described and provides essential growth room, numerous architectural research areas with direct application to the Space Station remain to be explored. These include:

Sensor Preprocessing. Gallium arsenide technology provides the capability to integrate analog, digital, and optical interconnection circuitry monolithically. This capability may be used to advantage in Space Station sensor preprocessing by combining analog to digital conversion hardware, preprocessing logic (for noise suppression, detector compensation, and bandwidth compression), and high speed optical data channels on a single chip located at the sensor.

Programming Methodology. Two research areas relevant to programming heterogeneous signal and image processing systems are being explored by us. The first is a hardware array processor architecture designed to perform certain run-time resource management functions through special hardware constructs [14]. This approach can out-perform static (e.g., compile-time) resource allocation and leads to a more productive throughput-to-volume ratio than software-based dynamic allocation schemes. The second approach is a normal form language, IMP, which provides the programmer with manageable access to hardware parallelism rather than attempting to "hide" parallelism. This approach gives the programmer a homogeneous software environment for programming a heterogeneous system -- hardware modules may be readily added or modified within the context of IMP.

Cellular Architectures. Image processing architectures based on collections of simple cellular processors [15] hold significant potential for maximizing the throughput-to-volume ratio in space-borne applications. A new pixel-processing architecture based on a parallel recirculating pipeline (PREP) is under development by us. This architecture avoids the classical computation-I/O-memory balance problem to achieve high pixel-processing performance in an extensible and high-order language programmable fashion.

Evolutionary Architectures. One research effort recently completed by us involved definition of an integrated signal and image processing subsystem using hardware, software, and mechanical standards in an open architecture configuration. Our architecture research laboratory (ARL) combines a multiprocessor environment with special purpose hardware in just such a configuration and allows us to plan and rapidly execute new processor module and system development in an evolutionary fashion.

5.0 Experimental Results

For the Color Image Enhancement Algorithms, the performance of the three algorithms was evaluated by synthesizing degradations which might be encountered for real space imagery for the high-quality photographs and then quantitatively comparing the degraded images, after enhancement with each of the three algorithms, to the original image. A block diagram of the quantitative evaluation procedure is shown in Figure 6. For the Tracking and Bandwidth Reduction algorithms, a sequence of 8 frames at 1 second intervals were digitized from the Mission 41-C "Video Highlights" video tape, where the imagery depicts the SYNCOM satellite rotating in space near the Space Shuttle, shortly after

deployment, against a cloud-covered earth background. The Tracking function's accuracy was empirically evaluated by comparing the algorithm's estimate for the two-dimensional change in location of the satellite (displacement vectors) to the best-guess at the actual displacement vectors, which were estimated by visual inspection of the gray levels of each successive pair of images. The maximum error in the estimated displacement vectors for the eight frame sequence was 1 pixel vertically and horizontally.

Each of the Color Image Enhancement algorithms were evaluated for the task of restoring degraded imagery for a range of image degradations in order to accurately characterize the algorithm's performance in terms of an empirically derived model for its behavior. The Color Image Balanced Histogram Equalization algorithm and the Color Image Accentuation algorithms are efficient computational techniques for the restoration of dynamic range for color imagery. Here the dynamic range is the difference between the maximum intensity value and the minimum intensity value of the luminance image. A set of three degraded color images were generated, one which had a dynamic range equal to 50% of the original image's dynamic range, one with 62% dynamic range and one with 85% dynamic range. These images were then restored with the Color Image Balanced Histogram Equalization algorithm and the Color Image Accentuation algorithm, in that order. Mean-square-error measures were then employed to quantitatively evaluate the accuracy of the restorations. These mean-square-error measures were the output luminance image signal-to-noise ratio and the output chromatic signal-to-noise ratios. For all cases but one, the quantitative measures demonstrated that the Color Image Enhancement algorithms did restore full dynamic range to the test imagery and also did not distort the intensity or chromatic information of the images. Further details can be found in 2.

A few of the results of the experiments with the Color Image Histogram Equalization and the Color Image Accentuation algorithm are presented in Figures 7 through 10. Figure 7 is the original image. Figure 8 is the degraded image with a 50% reduction of dynamic range with respect to the original image. Figure 9 is the result obtained by processing the image with the Color Image Balanced Histogram Equalization algorithm and Figure 10 is the result of increasing the image's saturation after histogram equalization. Inspection of these images demonstrates that full dynamic range has been restored.

6.0 Conclusions

There are a multitude of applications where knowledge-based techniques may be employed to improve the performance of image interpretation algorithms, for space applications. Because the benefits of knowledge-based image interpretation algorithms; increased algorithm reliability and increased robustness, are of great importance in the unique space environment, it is apparent that any future architectural concept development efforts should take knowledge-based techniques into consideration.

7.0 References

- [1] NASA/KSC "Space Station Mission Requirements Report, NASA/KSC Document Number MRWG-001, February 1984.
- [2] Honeywell Systems and Research Center, Video Image Processor for the Space Station, Final Report, December 1986.
- [3] Honeywell Systems and Research Center, Space Station Video Image Processor Concept Development, Final Report, 1985.
- [4] D.M. McKeown, Jr., "Knowledge Based Aerial Photo Interpretation," *Photogrammetria* 39 pp. 91-123 (1984).
- [5] M. Nagao and T. Matsuyama, *A Structural Analysis of Complex Aerial Photographs*, Plenum Press (1980).
- [6] W.A. Perkins, T.J. Laffey, and T.A. Nguyen, "Rule-based interpreting of aerial photographs using the Lockheed Expert System," *Optical Engineering* 25(3) pp. 356-362 (March 1986).
- [7] L. Sauer and J. Taskett, *Cultural Feature and Syntax Analysis for Automatic Acquisition*, SPIE Conference on Processing of Images and Data from Optical Sensors (1981).
- [8] M.D. Levine and A.M. Nazif, "Low Level Image Segmentation: An Expert System," *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6(5) pp. 555-577 (September 1985).
- [9] Y. Ohta, *Knowledge-based Interpretation of Outdoor Natural Scenes*, Pitman Publishing, inc. (1985).
- [10] J.J. Nitao, A.M. Parodi, "An Intelligent Pilot for an Autonomous Vehicle System", Proceedings of the 2nd Conference on Artificial Intelligence Applications, Miami, FL, December 11-13, 1985, pp. 176-183.
- [11] B. Bhanu and W. Burger, *DRIVE - Dynamics Reasoning from Integrated Visual Evidence*, Proceedings of the DARPA Image Understanding Workshop (Feb. 1987).
- [12] Honeywell Systems and Research Center, Bandwidth Reduction/Intelligent Target Tracker (BRITT) Demonstrator, Final Report, March 1984.
- [13] Yalamanchili, S., et al, The Architecture of a Video Image Processor for the Space Station, Proc. of the NASA Workshop on Space Telerobotics, Pasadena CA, January 1987. (These Proceedings)
- [14] Wehner, W.W., An Image Understanding System Based on Macro Data Flow, Proc. of SPIE Conference on Intelligent Robots and Computer Vision, Cambridge MA, October 1986.
- [15] Preston, K., et al, Basics of Cellular Logic with Some Applications in Medical Image Processing, Proc. of the IEEE, Vol. 67, No. 5, May 1979.

Table 1. Cross reference between applications and algorithms.

	Color Image Enhancement	Tracking	Surveillance	Identification	Proximity Operations	Bandwidth Reduction
Construction	X	X		X	X	
Satellite Servicing	X	X		X	X	
Redevious and Proximity Operations	X	X	X	X	X	
Inspection	X	X			X	X
Payload Delivery/Retrieval	X	X	X	X	X	X
Experiment Monitoring	X	X		X		X
Data Management and Communications	X	X		X		X
Training	X	X		X		X

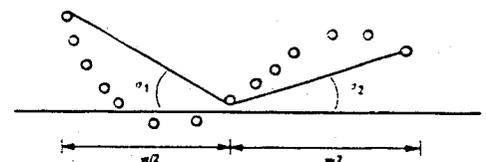


Figure 2. Computation of the Linearity Filter.

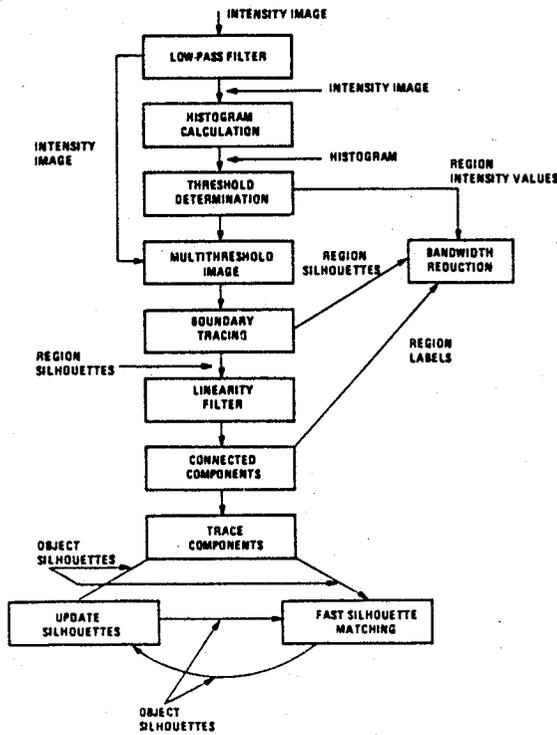


Figure 1. Functional decomposition of the Tracking and Bandwidth Reduction algorithms.

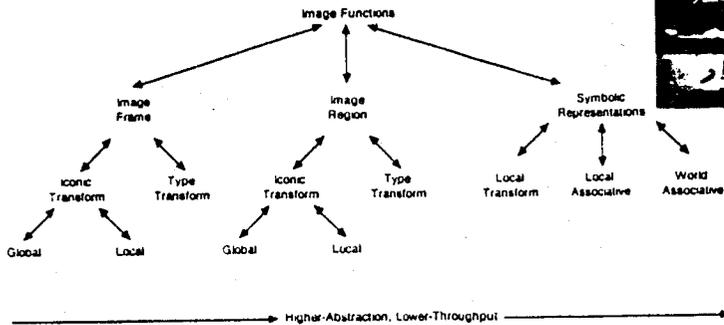
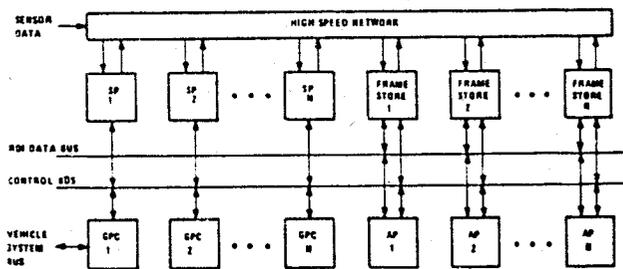


Figure 3. A Taxonomy of Image Understanding Operations.



- SP = signal processor
- AP = array processor
- GPC = general purpose computer

Figure 4. A Hardware Architecture Based on the Taxonomy of Figure 1.

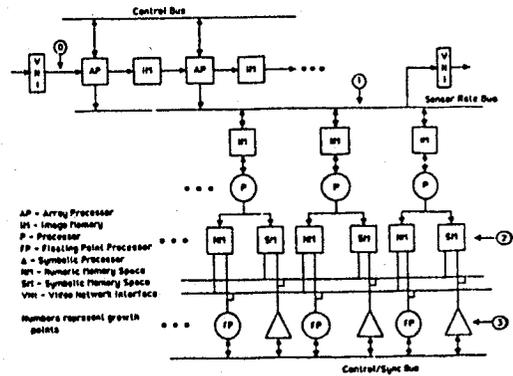


Figure 5. Video Image Processor (VIP) Conceptual Architecture.

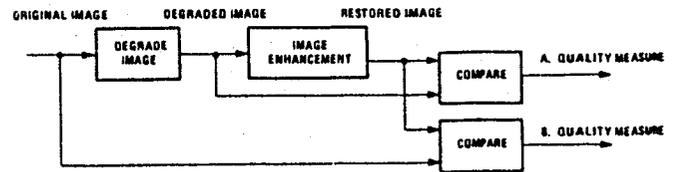


Figure 6. Color image enhancement algorithm performance is evaluated with various error measures.



Figure 7. Original image.



Figure 8. Degraded image: 50% dynamic range.



Figure 9. Restored image.



Figure 10. Restored and enhanced image.