論文 87-24-6-6

An Intelligence Image Compression System through Image Understanding

(영상 이해를 통한 지능형 영상압축 시스템)

金 鎭 衡*

(Jin-Hyung Kim)

要 約

본 논문은 통신선에 가해지는 제한에 따라 영상 압축률을 1：1에서부터 12,000：1까지 자유자재로 변화할 수 있는 지능형 영상압축 시스템인 AIIC에 대하여 기술하고 있다. 이 시스템은 통신 공학적인 알고리즘과 영상이해를 통한 지능형 알고리즘을 적절히 혼합함으로써 극도의 높은 수준의 영상 압축률을 보이고 있다. 이 시스템은 다수의 마이크로 컴퓨터로 구성된 네트워크 상에서 소프트웨어적으로 시뮬레이션되었다.

Abstract

This paper describes an intelligent image compression system called AIIC which is capable of adjusting image compression ratios ranging from 1:1 to 12,000:1 depending on available bandwidth. This system utilizes not only conventional image compression algorithms but also intelligent techniques through understanding image contents to achieve ultra-high compression ratios. This system was simulated on a micro-computer network.

I. Introduction

The transmission of images over data-links of varying channel capacity requires an intelligent application of image compression techniques. Since channel capacity can vary from time to time, a fixed compression ratio cannot be utilized for all time. The system described here, called Adjustable Intelligent Image Compression (AIIC) system, is capable of adjusting image compression ratios ranging from 1:1 to 12,000:1 depending on available bandwidth.

Image compression techniques incorporated in the system may be placed into two categories; conventional and intelligent. The conventional techniques, which are known as image coding to preserve overall image fidelity, yield up to approximately 50:1 compression ratios [1]. Significantly higher compression ratios can only be achieved by incorporating intelligent techniques such as image-content analysis and temporal compression. Image-content analysis determines what is in the scene and where it is. Only information pertinent to the mission objectives is transmitted in the compressed image. That is, important objects are transmitted with high spatial resolution and high frame update rate, while less important objects, such as background, are transmitted in a highly compressed form and slow update rate. Temporal compression is

---

*正會員, 韓國科學技術院 電算學科

(Dept. of Computer Science, KAIST)

接受日字：1987年 5月 19日

a technique designed to gain additional compression in multiple frame transmission. Here, new frame is compared with old frames and only the "new" information is transmitted. The AIIC system achieves various levels of compression ratios by combining intelligent techniques with conventional techniques.

The system has been developed by utilizing rule-based approach [2] for flexible system control. The algorithms used for image content analysis include well-established segmentation [3,4,5], feature extractions and classification algorithms for the object detection and identification [6], and line finding [7], and symbolic scenematching algorithms for background description and temporal compression. The AIIC system was implemented on a distributed micro-computer network consisting of several Z-80 based processors which communicated by message passing through the controller's blackboard. Design and implementation of the AIIC system, together with the operations of the system component modules, are presented in this paper.

II. Image-compression Strategies

The AIIC system provides 14 options of compression ratios ranging from 1:1 to 12,000:1. A specified bandwidth compression ratio is achieved by a combination of image compression techniques. Some of these techniques can be categorized as intelligent because scene content is derived in order to select the compression scheme to used.

For the lowest compression ratios, simple frame rate reduction and spatial resolution reduction, are employed in the AIIC system. Because a sequence of image frames is transmitted in a certain time interval, transmitting only every Nth frames easily achieves N:1 compression. Also, reduction of image resolution by a factor of M achieves a M:1 compression.

A transform coding scheme has been used to yield from 4:1 up to 16:1 compressions without significant degradation of image quality. One dimensional 16-point Hadamard coding scheme [1,3,8] has been employed in the system. It was determined that two bit-per-pixel and one and half bit-per-pixel coding schemes generated acceptably small distortions. However, higher ratio compressions, one bit-per-pixel and half bit-per-pixel coding schemes, produced too blocky, unacceptable results.

In contrast to the conventional frame rate and spatial resolution reduction, the idea of the intelligent compression schemes is to apply the frame rate reduction and spatial resolution reduction method non-uniformly over the image, depending on image content. After determination of scene content, the assignment of high resolution target windows and low resolution background features is determined. The windows containing important moving objects are updated more frequently, while the background less frequently. Levels of various compression ratios are derived by adjusting bandwidth reduction parameters. These parameters include window types and size, the number of the windows in a frame, updating rate, spatial resolution, and coding schemes of window. A typical example of intelligently compressed image is shown in Fig. 1.

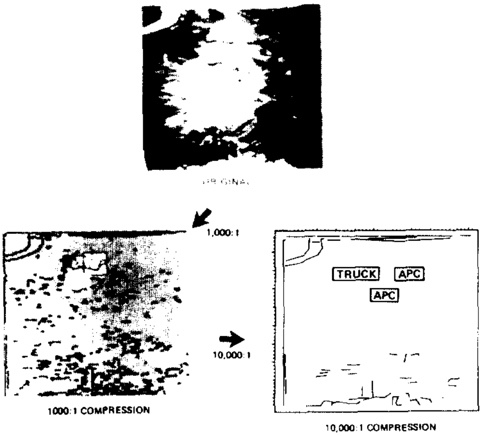


Fig.1. Intelligent image compression.

To achieve even higher ratio compressions, the background is represented as binary edge map or a line model. The edge map represents binary pixel data thresholded from the edge magnitudes at each pixel location, while the line model is formed by linking the edge pixels. Individual line segments are represented by two endpoints, thereby realizing a very large savings in bandwidth.

Temporal compression has been incorporated to gain large improvements in multiple-frame transmission. By comparing feature contents in consecutive frames, the displacement between frames caused by sensor motion can be determined. This knowledge allows the system to avoid transmitting background information which has been sent as a part of previous frames. By sending only the new information, and appending it to the old information at the receiving station, another large reduction in bandwidth is achieved. Table 1 shows the combination of compression techniques for achieving the varying degrees of bandwidth compression ratios discussed.

Table 1. Compression ratios.

SWITCH POSITION	BANDWIDTH COMPRESSION RATIO	COMPRESSION METHOD				REMARKS
1	1	NO COMPRESSION				CONVENTIONAL BANDWIDTH COMPRESSION SCHEMES ONLY
2	4	PICTURE REDUDEC TO 256×256				
3	16	PICTURE REDUCED TO 256×256 7.5FPS				
4	64	PICTURE REDUCED TO 256×256 7.5FPS 2bit HADAMARD CODING				
5	125	PRIORITY WINDOW BACKGROUND	128×128 256×256	7.5FPS 2FPS	2bit HADAMARD 1.5bit H	*DEPENDING ON NUMBER OF SECONDARY TARGETS IN THE SCENE
6	250	PRIORITY WINDOW BACKGROUND	64×64 256×256	7.5FPS 2FPS	2b-H 1.5b-H	
7	400~500*	PRIORITY WINDOW SECONDARY WIDNQOW** BACKGROUND	64×64 32×32 256×256	7.5FPS 3FPS 1FPS	2b-H 1.5b-H 1b-H	**UP TO SIX SECONDARY WINDOWS
8	700~1000*	PRIORITY WINDOW SECONDARY WINDOW** BACKGROUND	64×64 32×32 128×128	7.5FPS 3FPS 1FPS	1.5b-H 1.5b-H EDGE-MAP	
9	1000~1750*	PRIORITY WINDOW SECONDARY WINDOW** BACKGROUND TARGET DESCRIPTION	32×32 32×32 128×128 120 CHARACTERS/sec	7.5FPS 3FPS 1FPS	1.5b-H 1.5b-H EDGE-MAP	
10	1300~3000*	PRIORITY WINDOW SECONDARY WINCOW** D BACKGROUND TARGET DESCRIPTION	32×32 32×32 200LIENS 120CHARACTERS/sec	7.5FPS 3FPS 1FPS	1.5b-H 1.5b-H 36BIT/LINE	
11	3700	PRIORITY WINDOW BACKGROUND TARGET DESCRIPTION	32×32 200LINES 120CHARACTERS/sec	7.5FPS 1FPS	1b-H 36BIT/LINE	
12	6000	PRIORITY WINDOW BACKGROUND TARGET DESCRIPTION	32×32 200LINES 120CHARACTER/sec	1FPS 1FPS	1b-H 36BIT/LINE	
13	9000	BACKGROUND TARGET DESCRIPTION	200LINES 120CHARACTER/sec	1FPS	36BIT/sec	
14	12000*	TEMPORAL COMPRESSION TARGET DESCRIPTION	100LINES*** 120CHARACTER/sec	1FPS		***ASSUMING 50% OVERLAP

Ⅲ. System Overview

1. Blackboard system architecture

The AIIC system is a complex, distributed image understanding system which takes a temporal sequence of digitized gray level images as input and produces a sequences of compressed images and/or symbolic descriptions of interesting objects in the scene as output. The AIIC system consists of a number of independent special purpose modules (SPMs) which perform specific tasks under control of an executive controller. The controller is configured as a blackboard system in that system control knowledge, such as how and when to invoke SPMs, is encoded in the form of production rules. It is also the executive controller's function to determine a proper image-compression strategy to achieve a required compression ratio. The status of the system at any time is represented on the controller's blackboard. The controller monitors the state of processing and invokes appropriate SPMs when certain conditions are met. The execution of a SPM causes various functions to be performed and produces data to be added to the blackboard.

The system controller was implemented in a production system language called HAPS, developed at the Hughes Research Laboratories [9]. HAPS, a general purpose tool for constructing forward-chaining mechanism by the use of variables in conjunction with match restriction capability. A version of HAPS was implemented on a Z-80 processor which serves as a node in the micro-computer network.

One major advantage of the blackboard approach is its high modularity. The system may have different internal structures and may be written in different programming languages. However, they communicate with each other by means of message passing between the SPMs. Thus, internal properties of a SPM are transparent to the rest of the system. In our current implementation, the SPMs were written in various programming languages including LISP, C, assembler language, and HAPS production system language.

There are two dominant data flow paths in the AIIC system, as shown in the data flow diagram of Fig. 2. One path, which appears in the upper portion of the diagram, is for object detection and classification; the other path, which appears in the lower portion of the diagram, is for background description and temporal compression. For the upper path, the object detection and identification algorithms for an auto-cuer program [6] are utilized. In addition to the data-driven bottom-up approach, structural and contexual models were incorporated in the model-driven, top-down approach. For the lower path, the line-finder module produces line models of the background of sensed images, and the line models of two consecutive images are matched, and a displacement vector calculated for use in temporal compression.

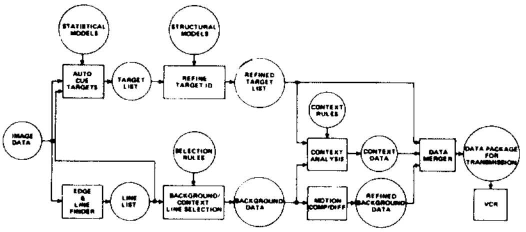


Fig.2. AIIC data flow.

2. Object detection and classification

The image understanding algorithm employed in the current implementation of the AIIC system utilizes a two-level approach to automatic object detection and classification. A low-level processor examines the full-frame image and selects points of interest where objects are likely to be located. These points of interest are then sent to the high-level processor for further, and more detailed, examination in window regions centered at the interest points. Here, object segmentation and feature extraction take place, and feature vectors are sent to a rule-based classifier for identification of object type. An overview of the object detection and classification process is given by the block diagram in Fig.3

In determining the low-level points of interest, a few easily calculated statistical parameters of the image are considered. The motivation here is to form, by a proper choice of low-level parameters, a linear discriminant of

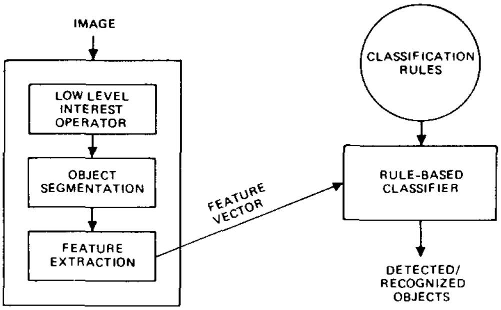


Fig.3. Object detection and classification.

these parameters so that object areas are more likely to possess high discriminant values. By appropriately thresholding the discriminant at this stage, a small number of interest points are produced and passed on to the high-level stage for further processing. The remaining, below threshold, points are excluded from further consideration and processing. The objective of the low-level processor is to locate as many of the object areas as possible using interest points, while discarding as many areas of the image as possible from complex and time-consuming computations by the high-level processor.

A window is centered at each interest point in the image, and object segmentation takes place to extract objects from the background. It is the function of the segmentation algorithm to separate those areas believed to be interior to an object from the background. A convolution algorithm suggested by Nevatia [7] is used to compute the edge values of the pixel within the window. A threshold is applied to keep only the dominant edges within the window. Then, interior point determination takes place. A provisional interior point is defined as a pixel that is surrounded in six of eight directions by above-threshold edge points. Because of the coarseness of the interior pont determination, convex portions of objects tend to be filled in; the same is true for regions between two objects in close proximity to each other. To overcome this drawback, a maximum likelihood assignment of interior versus exterior points is carried out using intensity histograms of the provisional interior and exterior points. This assignment eliminates the shadow artifacts and gives a true representation of object's silhouette. Smoothing and gap-filling algorithms are applied to obtain smooth boundary lines. For each segmented object, feature values such as area, average intensity are measured and sent to classifier for object recognition.

The classifier is a rule-based system that distinguishes one class from the other. Classification rules encode relationships between object classes and their feature characteristics in bottom-up fashion in the form of "IF features satisfy certain conditions THEN the object belongs to a certain class." Although contextual information as well as structural model information is incorporated with the classifier, the primary classification knowledge has been extracted from the statistical model which has been trained on a set of sample images. The contextual and structural information is used only for subsequent refinement, confirmation or rejectoin, of initial classifications. An advantage of having the classifier configured as a rule-based system is the synergic integration of all the available classification knowlege into an unified format. Although statistical classification knowledge may be encoded and utilized more efficiently in different formats (such as decision tree), these formats are not suited for contextual and structural knowledge representation.

3. Edge map and line model generation

In order to achieve the required high bandwidth reductions, the information content of a scene must be represented in a more symbolic form, rather than the original intensity representation. The AIIC system approaches this problem by representing the background information as either an edge map or a line model. To achieve the intermediate level compression ratios (around 1000:1 compression), edge maps are constructed for the background by thresholding the edge magnitude produced by convolving the image with a series of directionally weighted masks. The resultant edge picture is transmitted at reduced resolution at varying frame rates. This not only provides significant bandwidth reduction, but also preserves key features (such as roads and terrain features) which provide an understanding of the background scene

A further bandwidth reduction has been achieved by line model generation. Line segments are identified by a line-finding algorithm and transmitted as two end points, producing a very large savings in bandwidth. The AIIC system employs the Nevatia-Babu line-finding algorithm [7] in that edge pixels are thinned and linked in a heuristic way. The number of generated line segments is dependent on two thresholds: edge magnitude threshold and line length threshold. Increasing these two thresholds produces less line segments, and accordingly high ratio compressions can be achieved (Fig 4.)

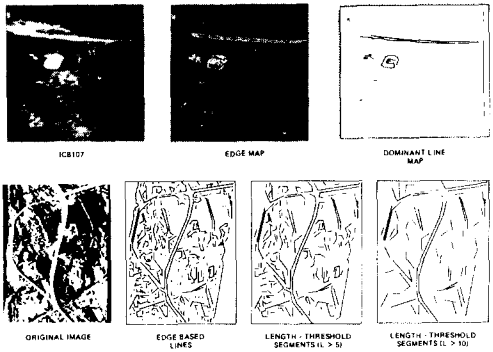


Fig.4. Examples of line modeling.

4. *Temporal compression*

While the line finding scheme for background representation has provided a significant improvement in single-frame bandwidth reduction, temporal compression has been implemented to gain even larger improvements in multiple-frame transmissions. The key to successful temporal compression lies in the ability to accurately determine the global offset between two images (represented by line models). In general, this offset is determined by locating instances of certain invariant image features, and computing the displacement between these features. In the AIIC system, near vertex feature, which is formed by line pairs with nearly intersecting end points, has been explored as a possible basis for scene matching among many candidates. The near vertex matching requires a complex algorithm involving the length of segments, their orientation, the ratio of segments forming the vertex, and the included angles. Since this kind of features is generally consistent between images and they do not occur many times, matching near vertices yields a robust, reliable, and fast result. Having determined the offset between two consecutive images by matching near vertex features, the final step is to actually perform the temporal compression and transmit the compressed data. The set of edges which extends outsides of the portion of the image already sent is determined and transmitted.

IV. Micro-Computer Network

The AIIC system has been implemented on a distributed micro-computer network. By its nature, this network approach produced a more modular and flexible design. Each of these independent processors communicates with the other processors via simple messages through the network hub machine. Separating the tasks onto the individual machines provides the desired modularity and the simple message-passing communications between them provide a high degree of flexibility.

The network developed for the prototype AIIC system consists of six separate Z-80 processors (Fig. 5). Using the network message passing system, any of these machines can send a message packet to any other machine via RS-232 data lines. This asynchronous message system allows easy modification and integration of the various components of the system, even to a level where additional processors and modules could be added without altering the network structure or any the existing nodes.

For the network controller, a modified M operating system was used. Any input

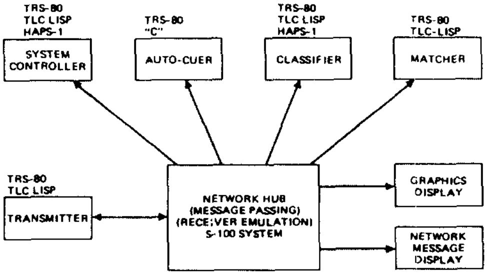
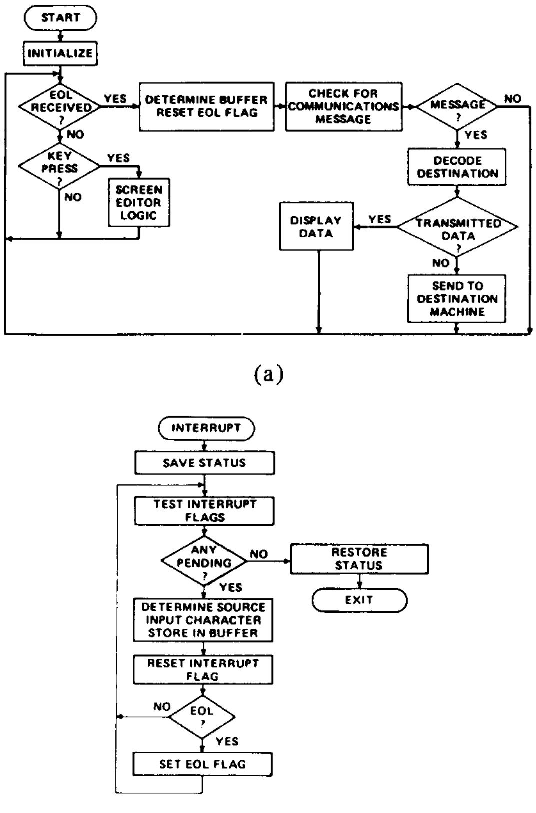


Fig.5. AIIC micro-computer distributed network.

signal appearing on the machine's RS-232 port would be handled by CP/M as if it were typed on the keyboard and similarly, any output directed to the video display would simultaneously appear as an output on the RS-232 port. The overall network which links five micro-computers in a radial network configuration was built around the hub machine. Each network node communicates with the use of the network hub by passing formatted messages made up of standard ASCII characters. The message passing sequence is initiated when a network node machine type a message on its screen. The modified CP/M would essentially echo this message on the RS-232 output data lines, generating an interrupt at the network hub. The interrupt service routine then would examine the interrupt flag, determining source of the interrupt, and get the character from the source. This character would then be stored in the input buffer with identification of the source. If this character happens to be an end-of-line character, the message is decoded to determine the destination node for the message. Then the message is placed in the transmit buffer for the destination node, and finally appeared on the machine as if typed at the keyboard. These service routines are illustrated in Fig. 6.



(a)

(b)

Fig.6. (a) Network message passing and receiver emulation.
(b) Network interrupt service.

V. Conclusion

An intelligent image compression system has been described which is capable of adjusting compression ratios from 1:1 to 12,000:1 depeding on the restrictions placed on bandwidth capacity. The system achieves high ratio compressions by incorporating image understanding techniques, i.e., image-content analysis is performed on the original image and only information pertinent to a given mission objective is transmitted. The AIIC system is organized as a blackboard system in which several special purpose modules are processed co-operatively under the control of a rule-based executive controller. The system has been implemented on a distributed micro-computer network in that each special purpose module is processed in parallel with other modules. The modules communicate by passing messages through the network hub machine. In this fashion, the system achieves a high degree of modularity and flexibility.

References

[1] W. Pratt, *"Digital image processing"*, John Wiley, New York, NY, 1978.

[2] B. Buchanan and E. Shortliffe *"Rule-based expert systems,"* Addison-Wesley, Reading, Massachusetts, 1984.

[3] R. Duda and P. Hart, *Pattern Classification and Scene Analysis,* John Wiley, New York, NY, 1984.

[4] D. Panda "Segmentation of FLIR images pixel classification," *Proceedings of Image Processing Workshop,* April 1977.

[5] R. Hartley, et al., "Segmentation of FLIR images: a comparative study," *IEEE Transactions on the Systems, Man, and Cybernetics,* July/August 1982.

[6] L.M. Rubin, and D.Y. Tseng, "Automatic target cueing," *Proceedings of the Fourth International Joint Conference on Pattern Recognition*, Kyoto, Japan, Nov. 1978.

[7] R. Nevatia and K.R. Babu, "Linear feature extraction and description," *Computer Graphics and Image Processing,* 13, pp. 269, 1980.

[8] D.Y. Tseng, et al., "Intelligent bandwidth compression," *Processings of the Society of Photo-Optical Instrumentation Engineers,* San Diego, August 1979.

[9] J.H. Kim, "HAPS2 user's manual," Research Report, Hughes Research Laboratories, Malibu, CA, 1983.

[10] L. Erman and V. Lesser, "A multi-level organization for problem solving using many diverse, co-operating sources of knowledge," *Proceedings of International Joint Conference on Artificial Intelligence,* 1975.