Implementation of Real-time DIS H.264 Encoder for Airborne Recorder

Ju-Hun Nam, Seong-Jong Kim, Sung-Min Kim and Nam-Sik Lee
DANAM SYSTEMS INC.
jhnam@danam.co.kr

Jin-Hyung Kim
LG Electronics

ABSTRACT

When developing a video compression system in black box for aircraft, it is necessary to consider the characteristic of the images and the surrounding environment. The images captured in and out of aircraft have excessive movement-related issues, which make the results difficult to analyze and interpret. Failure to remove the tremors in the video component inevitably leads to poor compression efficiency and degrades the video imaging performance in the airborne black box. Therefore, it is necessary to develop a Compression System which can stabilize the video-image and efficiently utilize high compression recording for aircraft without special hardware.

Based on the current situation, we suggest a real-time electronic video stabilization algorithm for airborne recorder which recovers shaky images simply and efficiently to work beside a developed stabilization system based on the H.264 Encoder using DSP.

KEYWORDS

Digital image stabilization, H.264, CODEC, DIS.

INTRODUCTION

It's essential to consider the characteristic of the images of aircraft and the surrounding environment when developing video compression system to be loaded on aircraft-use black box. Generally, the compression efficiency of the video in which excessive shaking exists is degraded. The video acquired from aircraft basically has excessive shakiness for people to recognize and the limitation of loaded hardware and communication environment also exists. Therefore, it's necessary to develop compression system that stabilizes shakiness of video and has excellent compression rate. In this study H. 264[1], which is recently in the limelight as the standard of video compression, is adopted. Also the algorithm that can simply and efficiently single out video's shakiness is suggested.

Whereas H. 264 is increased in compression efficiency compared with existing compression standard, it has a problem of being delayed in processing as the amount of encoding algorithm's calculation is increased. Thus, it's a trend to develop real-time video encoding system by applying optimized high-speed video processing algorithm with use of DSP. In this study electronic stabilization[2][3] module and video encoding instrument is realized, which operates in real time with DSP being basic processing device.

GENERAL DIGITAL IMGAGE STABILIZATION

Table 1. Various motion-predicting algorithms to get local motion vector

2-dimensional stabilization technique	Bit-plane matching (BPM)
	Representative point matching (RPM)
	Selected areas matching (SAM)
	Edge pattern matching (EPM)
3-dimensional stabilization technique	Optical flow based technique
	Multi-resolution, iterative process algorithm
	Multiple visual cues

Global motion estimation

As most of image stabilization techniques are sensitive to speed and calculation amount, the method to select local part in entire images and find out local characteristics and finally predict global motion by suitably utilizing this is used to find out characteristics of global image as previously mentioned. Table 1 summarizes various local motion estimation algorithms. Fig. 1 shows the case that the partial motion among local motion vectors(LMV) is completely different from other area motion.

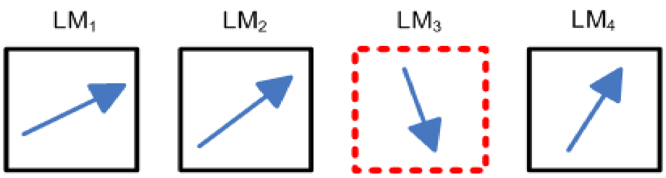


Fig. 1. The motion vector completely different from other motion

The motion of some areas completely different from other area's motion among measured area motions in Fig. 2 is ignored.

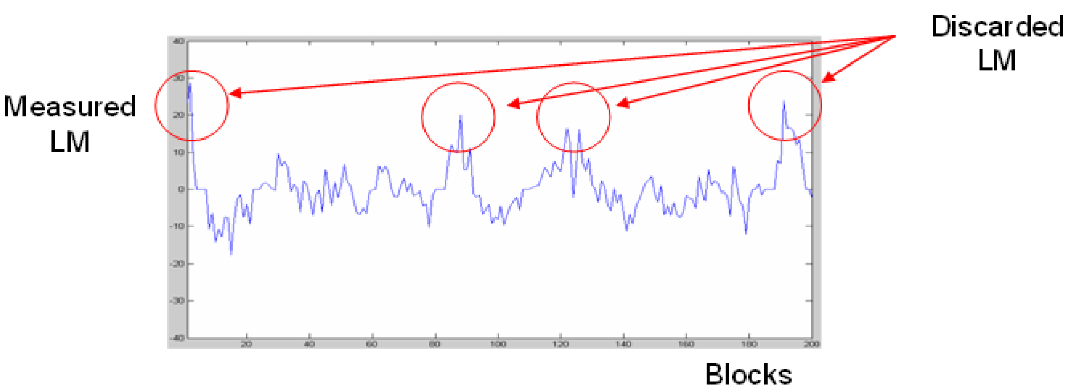


Fig. 2. Division of meaningless area motion vector

Unintentional motion compensation

After carrying out prediction of global motion, create final stabilized image after obtaining unintended camera motion by dividing unintended camera motion and intended camera motion from real image. Global motion in above course contains all of intended motion(pan, tilt) and unintended motion(hand shaking, wind). Therefore, we get to find out unintended motion except practically intended motion by using noise suppression technique like IIR filter or moving average in signal processing field.

Fig. 3 shows intended motion obtained with use of global motion vector, IIR filter through global motion vector and moving average method. We can see that moving average method obtains more stabilized motion vector compared with the method that uses general IIR filter.

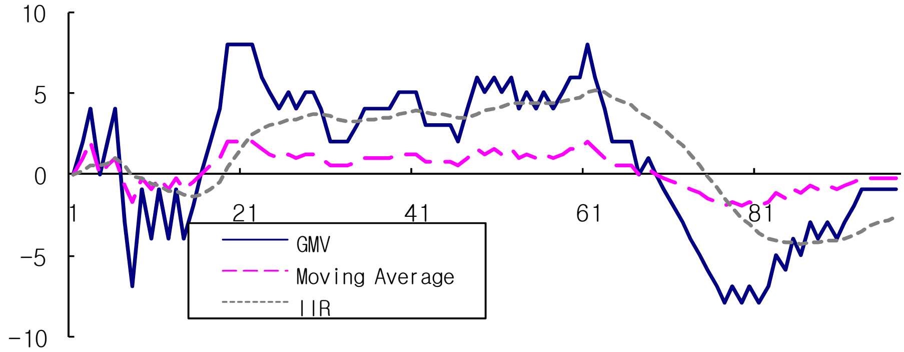
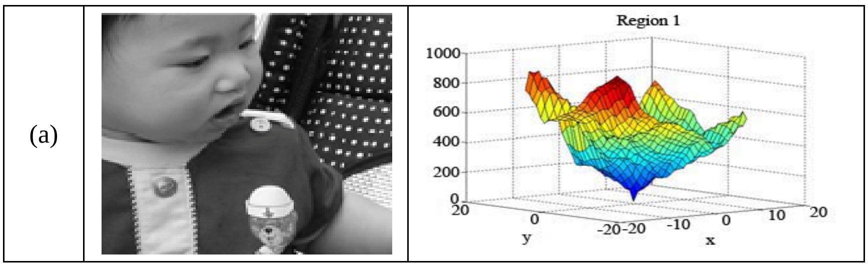


Fig. 3. Intended motion vector

PROPOSED DIGITAL IMGAGE STABILIZATION

Block matching algorithm, BMA

As a widely used method for estimation of motion vector, block matching algorithm is the technique which makes the motion vector by searching a minimum sum of absolute difference(hereunder SAD) point between each pixel in the current frame's certain block and previous frame's search area. This time, the distribution of SAD value calculated within search area is called SAD map, and if the part whose image within block has features and the image within search area has little part to be confused with the image within the block, it shows the pattern that only a specific point's SAD value is low when observing SAD map. On the other hand, in case of low SNRs without distinct image as the repeated pattern where the image within block repeatedly exists in search area, lack of feature in which the image within block has no feature, it shows SAD map that can't clearly sort out the smallest SAD value as distribution of SAD values are complex. Fig. 4 shows various pattern of SAD map.



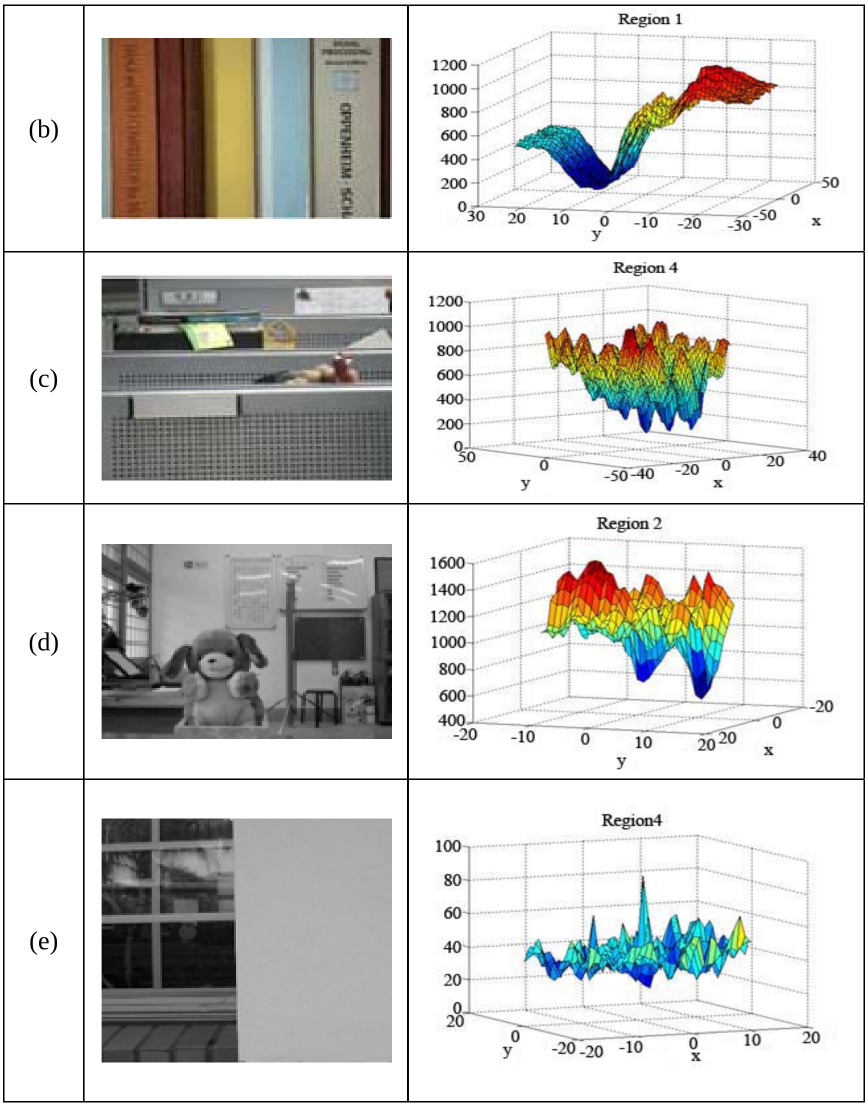


Fig. 4. Various pattern of SAD map.

In (a) of Fig. 4 shows distribution of SAD value in area of normal status, it's the best case to determine minimum SAD value as it distinctly points out only one place with rapid decrease of SAD value in middle point. On the contrary, in case of (b) in Fig. 4, we can see that distribution of SAD map is also not clear and spread in direct line. Also the other image in Fig. 4 is not clear with the minimum value of SAD and each image shows different distribution.

Median based algorithm

- Location of block when estimating area motion vector

Estimate total of 25 area motions with 3-Step block matching algorithm as for 5 blocks in each area, after dividing the image into 5 areas. The location of 5 regions R_1, R_2, R_3, R_4, R_5 in image is as Fig. 5. W in Fig. 5 means the width of image and H means the height of image. The location of 5 blocks in one area is as Fig. 6. W_R in Fig. 6 means the width of the area, and H_R means the height of the area. L_{BS} means length of a side of block that has size of $L_{BS} \times L_{BS}$.

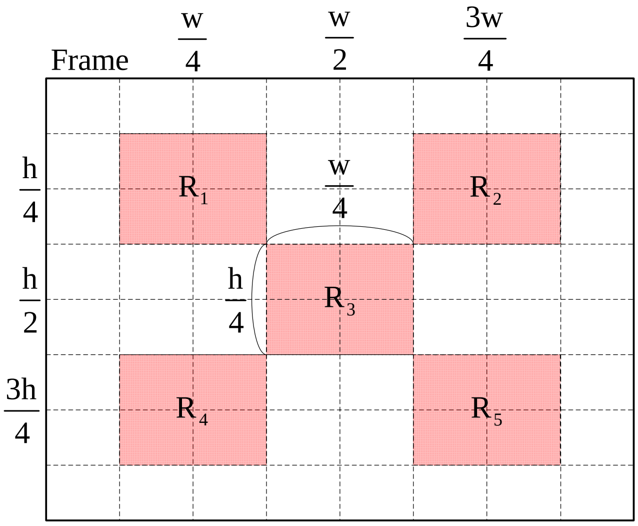


Fig. 5. Location of 5 regions in image

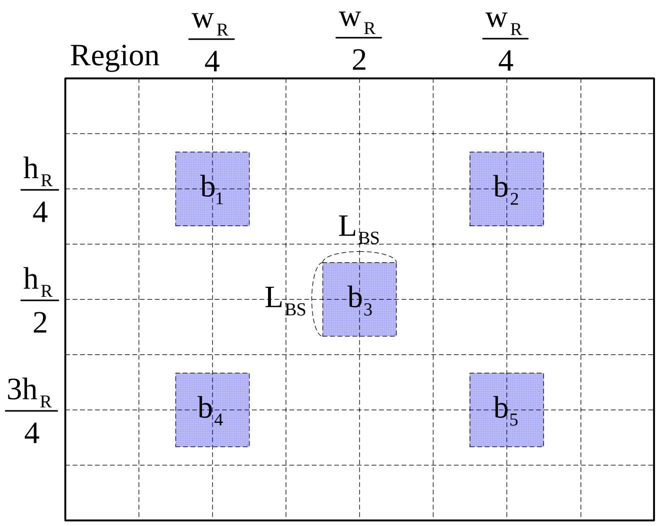


Fig. 6. Location of 5 blocks in area

- Estimation of entire motion vector in Full-pixel unit

When defining the motion vector as $V_{L_{ij}}$ estimated at j th block in i th area, $V_{xL_{ij}}$, $V_{yL_{ij}}$ are defined as x, y elements of $V_{L_{ij}}$. And when defining the motion vector representing i th area as V_{R_i}, it also has x and y elements and each can be defined as V_{xR_i}, V_{yR_i}. The value of V_{xR_i} and V_{yR_i} is obtained as follows,

$$\begin{aligned} V_{xR_i} &= median(V_{xL_{i1}}, V_{xL_{i2}}, V_{xL_{i3}}, V_{xL_{i4}}, V_{xL_{i5}}) \\ V_{yR_i} &= median(V_{yL_{i1}}, V_{yL_{i2}}, V_{yL_{i3}}, V_{yL_{i4}}, V_{yL_{i5}}) \\ & where \;\; i \in \{1,\ 2,\ 3,\ 4,\ 5\} \end{aligned} \tag{1}$$

In the same method, when defining global motion vector in full-pixel unit as $V_{G_{\text{int}}}$, its x, y element $V_{xG_{\text{int}}}$, $V_{yG_{\text{int}}}$ are defined as follows,

$$\begin{aligned} V_{xG_{\text{int}}} &= median(V_{xR_1}, V_{xR_2}, V_{xR_3}, V_{xR_4}, V_{xR_5}) \\ V_{yG_{\text{int}}} &= median(V_{yR_1}, V_{yR_2}, V_{yR_3}, V_{yR_4}, V_{yR_5}) \end{aligned} \tag{2}$$

- Estimation of motion in Sub-pixel unit

We operate Sub-pixel motion estimation for minute shake unavailable to detect in full-pixel unit. One pixel practically can’t be broken in half. When estimating motion vector after creating twice or 4 times image by operating search area in linear interpolation in the course of using block matching algorithm, the estimated motion vector gets to have twice or 4 times value of original motion vector and when dividing this two equal parts or 4 equal parts, we can get 0.5, 0.25 unit of motion vector value.

If we carry out image interpolation for all search areas, the calculation amount gets increased. To reduce this calculation amount, we assume 2 kinds as follows.

1. The estimated global motion vector is the true motion vector in full-pixel unit.
2. The margin of error of estimated global motion vector is ± 0.5.

Under this assumption, we carry out image interpolation in the region around a point that is located far off as much as global motion vector in full-pixel unit from standard point. Therefore, we can greatly reduce calculation amount compared with the method which carries out image interpolation of whole search area. Fig. 7 shows the area to be copied within search area of reference image, to estimate global motion vector in sub-pixel unit toward x direction. This area is located far off as much as $(V_{xG_{\text{int}}}, V_{yL_{ij}})$ from standard point, and has size of $(L_{BS}+2)\times L_{BS}$ in which the size of block is widened to each 1 pixel to left and right. x element $V_{xG_{\text{int}}}$ of global motion vector in Fig. 7 is the same as motion vector $V_{xL_{ij}}$ of i th area, j th block.

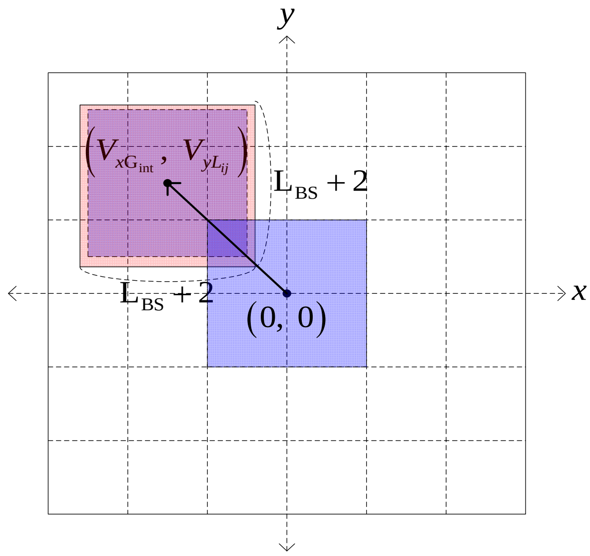


Fig. 7. Location to be copied for estimation of motion in sub-pixel unit

With use of copied area create image in $(2\cdot L_{BS}+1)\times L_{BS}$ size linear interpolated as Fig. 8.

F_0	F_1	F_2	$\cdots$	$F_{L_{BS}-1}$	$F_{L_{BS}}$	$F_{L_{BS}+1}$

original image

h_1	F_1	h_2	F_2	$\cdots$	$F_{L_{BS}-1}$	$h_{L_{BS}}$	$F_{L_{BS}}$	$h_{L_{BS}+1}$

interpolated image

Fig. 8. Image interpolation method for estimation of motion in sub-pixel unit

F becomes the location of integer unit pixel, and h becomes the location of 0.5 pixel. h is obtained as follow,

$$h_i = \frac{F_{i-1}+F_i}{2}, \quad i \in (1, 2, \cdots, L_{BS+1}) \tag{3}$$

Estimate motion vector in sub-pixel unit through calculation of (-1, 0, 1) for x element, using block matching algorithm with current frame's block, after operating linear interpolation of the location block of referenced image. For the method of estimating motion vector in sub-pixel unit of y element, it's the same as previously described method and only the direction is changed to up and down. When y element of global motion is selected at m th area, n th block, the area to be copied for image interpolation is located far off as much as $(V_{xL_{mn}}, V_{yG_{int}})$ from m th area, n th block's search area standard and has size of $L_{BS} \times (L_{BS}+2)$ in which widened to each 1 pixel in block size up and down. And the size of image created and interpolated is $L_{BS} \times (2 \cdot L_{BS}+1)$. When defining the motion vector in sub-pixel estimated in above method as $V_{G_{sub}}$, the range of $V_{G_{sub}}$'s x, y element $V_{xG_{sub}}$, $V_{yG_{sub}}$ is defined as follows,

$$V_{xG_{sub}} \in \{-0.5, 0, 0.5\},\\ V_{yG_{sub}} \in \{-0.5, 0, 0.5\} \tag{4}$$

Final global motion vector V_G is calculated as follows,

$$V_G = V_{G_{int}} + V_{G_{sub}} \tag{5}$$

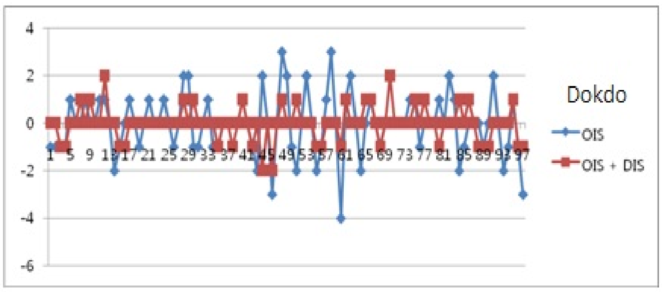
- Intended motion calculation

To divide intended motion V_I and shaky element V_H from calculated global motion V_G, use multi IIR filter as in (6). In (6), n means n th frame of video.

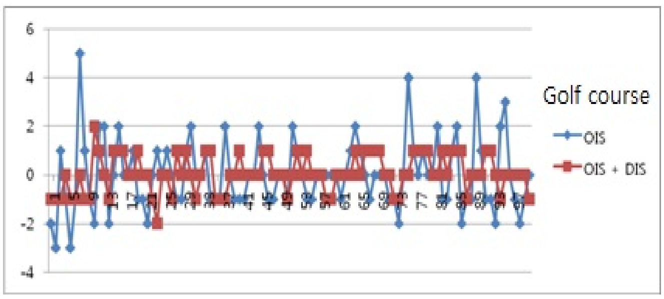
$$\begin{aligned} \tilde{V}_I(n) &= \alpha \cdot \tilde{V}_I(n-1) + (1-\alpha) \cdot V_G(n), \\ V_I(n) &= \beta \cdot V_I(n-1) + (1-\beta) \cdot \tilde{V}_I(n), \\ V_H(n) &= V_G(n) - V_I(n), \\ &where\ \alpha \in (0,1),\ \beta \in (0,1). \end{aligned} \tag{6}$$

EXPERIMENTAL RESULTS

Fig. 9 shows the compared result in which HMV value, the shake element of each experimental image that only optical image stabilization(OIS) are applied and result in which double-applied OIS and digital image stabilization(DIS). Table 2 shows the comparison of applying RMSE value of each image's shake element before and after applying DIS. In case of double-applying DIS and OIS, we could check that RMSE element is decreased from least 19% up to 57% largely, compared with image that applied only OIS.



(a) Width shake element of 'Dokdo' image.



(b) Height shake element of 'Golf course' image.

Fig. 9. Comparison of HMV value.

Table 2. Comparison of each image’s RMSE value

	RMSE value	
	Before DIS	After DIS
‘Dokdo’	0.107	0.087
‘Unpaved road’	0.130	0.095
‘Golf course’	0.070	0.055
‘Thermal image’	0.044	0.019

The proposed method has been realized using DSP (Fig. 10). Fig. 11 (a) shows the sample sequence, and Fig. 11 (b) illustrates the stabilized image using the proposed method. This shows that shake elements are compensated by the proposed method.

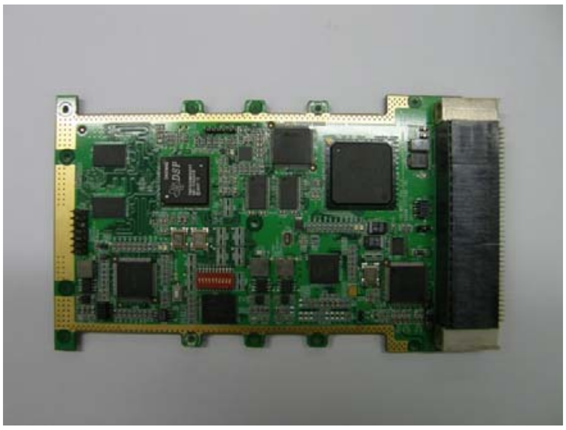


Fig. 10. Developed Real-time DIS H.264 Encoder for Airborne Recorder

(a) Origial (b) Proposed DIS

Fig. 11. Experimental result

REFERENCES

[1] *Advanced Video Coding for Generic Audiovisual services, ITU-T Recommendation H.264 - ISO/IEC 14496-10(AVC)*, ITU-T and ISO/IEC JTC 1, 2003.

[2] T. Kinugasa, N. Yamamoto, H. Komatsu, S. Takase, and T. Imaide, “Electronic image stabilizer for video camera use,” *IEEE Trans. Consumer Electron.*, vol. 36, no. 3, pp. 520-525, Aug. 1990.

[3] S.-J. Ko, S.-H. Lee, and K.-H. Lee, “Digital image stabilizing algorithms based on bit-plane matching,” *IEEE Trans. Consumer Electron.*, vol. 44, no. 3, pp. 617-622, Aug. 1998.