AVC-167

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION
ORGANISATION INTERNATIONALE DE NORMALISATION
ISO-IEC JTC1/SC2/WG11
CODING OF MOVING PICTURES AND ASSOCIATED AUDIO

ISO-IEC/JTC1/SC2/WG11
MPEG 91/231
November 1991

Source: R. Mokry, Y.B. Yu, T.H. Chiang, A. Eleftheriadis and D. Anastassiou, Columbia University

Title: COLUMBIA UNIVERSITY'S MPEG-2 PROPOSAL

Registration Number: 39

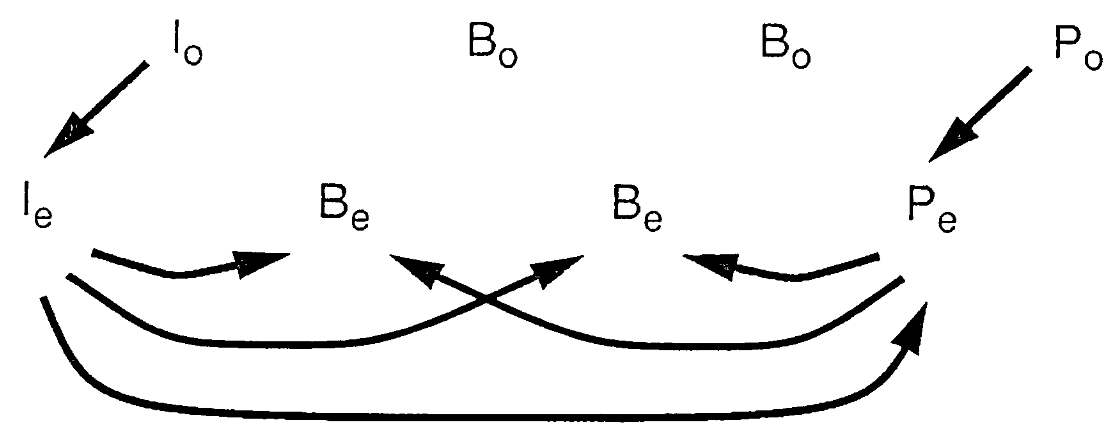
CONTENTS

PART 1: SUMMARY

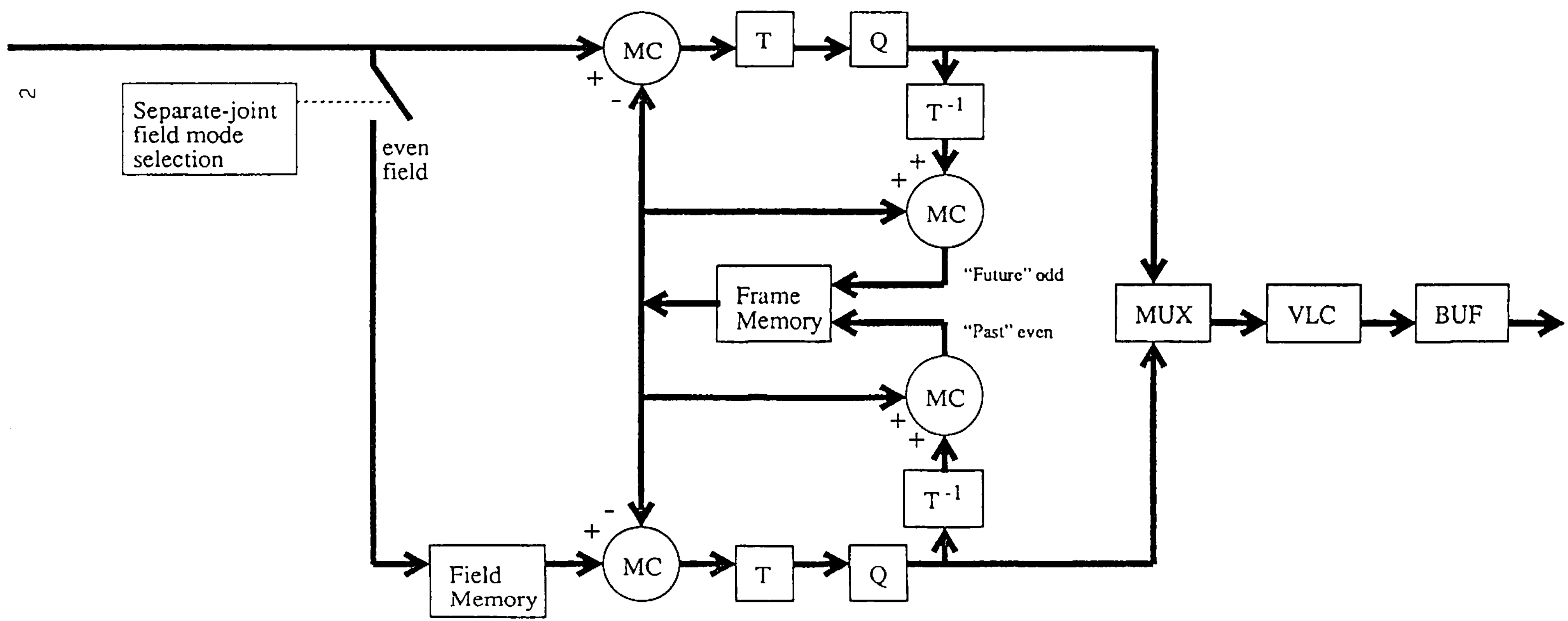
Abstract: A DCT-based scheme is proposed for interlaced video coding, as a natural extension of MPEG-1. We found that separate coding of the odd and the even fields so that even fields use information from the already coded odd fields has many advantages and near-optimum coding performance, when coding at full horizontal resolution. For slightly better coding performance, at the expense of compatibility, the two fields in each macroblock can be coded either separately, or jointly, in an adaptive block-by-block manner.

"Continuous scalability" can be achieved in the DCT domain by appropriate filtering, even in the presence of motion compensation, if care is taken to attenuate error propagation using predictive leak, preferably in the DCT domain [10]. The same approach can be used for an MPEG-1 compatible scheme.

The following figure shows a concise block diagram, together with an illustration of the allowed modes when even fields are coded separately.



Block Diagram of Basic Algorithm



Synopsis of technical findings and recommendations:

1. Hybrid, motion compensated, uni- and bi-directionally predictive DCT coding, similar to MPEG-1, is well suited for efficient coding of CCIR-601 video at 4 and 9 Mbits/s.
2. Separate coding of odd and even fields of interlaced video is desirable because:
   a. it enhances coding performance, at least for some macroblocks
   b. it facilitates scalability
   c. it facilitates compatibility with MPEG-1
   d. it allows for graceful degradation using multiresolution channel coding
3. For each frame, coding the "future" odd field first, allows subsequent coding of the "past" even field of the same frame, using the already coded future odd field with bidirectional prediction. This is particularly useful for "P" frames, as they are used as "anchor" frames for the "B" frames.
4. Bidirectional prediction is essential, because motion compensation can accommodate uncovered background.
5. Continuous scalability is possible in the DCT domain using appropriate filtering, at the expense of slightly worse coding performance.

1. Separate coding of odd/even fields

The even fields of interlaced video can be coded assuming that the neighboring odd fields have already been coded at full horizontal resolution. This can be done by either independent coding of the high horizontal frequencies of the odd fields, for full MPEG-1 compatibility; or by having coded the full-horizontal-resolution odd fields using MPEG-1, perhaps with modified DCT coding. For each macroblock there is access to the previous co-sited even field, in addition to both past and future odd fields.

We have proposed [2,3] using a multiplicity of recursive (predictive) and interpolative modes, including combined (averaged) hybrid recursive/interpolative modes. Like MPEG-1, a large number of modes can be selected, including the intrafield mode. The following four modes are the most widely used - particularly the first one:

1. Averaged mode between the previous co-sited even field and the future odd field.
2. Averaged mode between the past and future odd fields.
3. Single mode using the co-sited field.
4. Single mode using the future odd field.

Therefore, all even fields can be coded using future information. Using information from the future is desirable, because it allows for uncovered background.

In order to reduce frame memory requirements, it is possible to discard mode 2 above, using only the co-sited and the future odd field. All modes are MPEG-type, so generalization of hardware is easy. Any error propagation is stopped whenever the encoder selects use of purely interpolative modes, without otherwise notifying the decoder, resulting in a finite-memory system.

An average compression improvement of 40% was found in the even fields, when compared with independent MPEG-1 coding of the even fields, with the same observed quality. This translates to about 20% overall bit rate savings for interlaced video coding, compared with independent field coding.

There are two additional advantages of this approach:

1. Graceful degradation in the presence of channel errors, particularly for broadcasting applications, via multiresolution channel coding and transmission. Higher protection is

assigned for the coded bits of the odd fields than those of the even fields. When the eve fields cannot be coded due to channel errors, they can be interpolated, resulting in grace degradation. It should be noted that digital video coding with two classes of coded bi for prioritized transmission has also been proposed in the U.S. to the F.C.C. [6,7] for a digital Advanced Television terrestrial broadcasting system.

2. HDTV/Standard TV Compatibility: it is desirable to have first independently cod the odd fields, not only for MPEG-1 compatibility, but also because they can be used resampling to other interlaced standards. For example, if an MPEG-type technique is us for interlaced HDTV coding, an interlaced Standard-TV version can be created from t odd HDTV fields. This, of course, could also be used for graceful degradation in terrest broadcasting, as described above.

2. Processing order of fields

There are various proposed extensions of MPEG-1 (e.g. "MPEG++" in [6]) in whic two fields are combined into one frame, which is then DCT-coded). We believe that th approach is suboptimal if it does not allow escape to a mode of separate field coding, unles the interlaced data were artificially created from progressive scanning. If it is desired (in [6]) to treat two fields jointly as a pair in an MPEG-like fashion, then at least the futu field should be coded first, and then the other field should be coded making possible u of the just-coded future field [4]. In other words, intra even fields (Ie) can still use the just coded intra odd (Io) fields. "Predicted" even fields (Pe) can use the next Po and th previous Pe and Po (or Ie and Io). Bidirectional even fields (Be) can also use other fields for prediction [1], but, for simplicity of implementation, we have only used the same fields in the neighboring "anchor" frames.

Still, however, we found that processing the even fields in their actual "chronological" order (and blocking recursive modes every N=12) gives better results than separating the even fields into three classes (Ie, Pe and Be). As mentioned above, higher protection could be assigned for the coded bits of the odd fields than those of the even fields. When the even fields cannot be coded due to channel errors, they can be interpolated, resulting in graceful degradation.

3. Adaptive separate/joint coding of fields

Compared to entirely separate field coding, performance can be slightly improved by allowing joint coding of even and odd fields. This approach is included in [8], and is also been proposed in [1]. This, however, will be done, if scalability, multiresolution channel coding, and possible compatibility with MPEG-1 are sacrificed or severely compromised.

The following reasoning can be used to explain this fact: Hierarchical pyramidal coding provides convenient multiresolution coding, but the performance at the highest resolution is reduced, as was shown, e.g., in the JPEG standard. For example, given an interlaced freeze-frame of a moving object, it makes sense to compress the first field, and then the second field based on the reconstructed values of the first field (see, e.g., [9]). However, if the frame contains no moving objects, the coding performance of the final result will be compromised by this two-stage approach. It is then preferable to code the fields jointly, because they are properly aligned. Similarly, in motion compensated predictive video coding, if an object has uniform motion, the error image contains properly aligned fields and it may be preferable to compress them jointly. This is not the case, however, when the object is accelerating, or at the critical areas around the edges of moving objects.

Accordingly, the decision on whether to code the fields separately or jointly must be taken on a macroblock-by-macroblock basis.

If background is uncovered, accurate prediction can only be made using future fields. Uncovered background may not occur often, and the SNR will not be affected by local inaccurate coding, but artifacts can be quite visible. Therefore, if fields are coded separately, the future (odd) field is coded first, so that the previous even field can then be coded using the results of the future odd field, with backwards motion estimation.

This implies that the whole odd field must be coded at a first stage. The macroblocks for which the even field was not coded jointly with the odd field are tagged. Coding of the even field is completed at a second stage, using the adjacent future odd field, in addition to past even and odd fields, as in [5].

4. Conclusions

Compatibility, robustness for multiresolution channel coding, and scalability can be achieved by coding the odd fields, followed by even field coding, using the already coded future odd field to accommodate uncovered background. We have found, however, slightly better coding performance of the final signal, by occasionally (but not always) allowing the fields to be coded jointly. The optimum approach will highly depend on the weights of relative importance that will be assigned on the various issues.

5. Contents of D1 tape

For the demo in Kurihama we have used a fully automatic adaptive technique, in which the two fields in each macroblock are coded either separately, or jointly (if the two field-based motion vectors are identical), as described in section 3 above, and in the accompanying Figures. This adaptive switching among the two modes is allowed for P, B and I frames. The technique is similar to the one outlined in [1]; some differences are indicated below (see also following figure):

1. The even field of P frames can be coded using the already coded (future) odd field of the same frame.

2. We use M=3 (rather than 2) for B frames

3. B fields are predicted only from the same fields in the "anchor" frames.

4. Occasional adaptive use of BTC is done for bilevel blocks, in lieu of DCT.

It should be noted that the algorithm used for the D1 tapes has not been optimized in various ways, e.g. we use VLC's identical to those of MPEG1, identical treatment of 8x8 DCT blocks whether or not they belong to a field or frame, nonoptimal quantization matrices, etc. We also do not use any pre- or post- processing, although we have found that specific such schemes yield significant improvement [9]. As a result, visual quality does not indicate the optimum coding performance of the proposed technique.

References

[1] VADIS COST Algorithm Group 2, "Overview of Algorithm", ISO-IEC JTC1/SC2/WG11 MPEG 91/159, 16 August 1991.

[2] F.M. Wang and D. Anastassiou, "High-quality coding of the even fields based on the odd fields of interlaced video sequences", IEEE Transactions on Circuits and Systems, Vol. 38, No. 1, January 1991, pp. 140-142.

[3] F.M. Wang and D. Anastassiou, "Compatible even field coding from odd fields using the co-sited field in single and averaged modes", ISO-IEC JTC1/SC2/WG11 MPEG 90/308, December 1990.

[4] D. Anastassiou, T.H. Chiang, A. Eleftheriadis, R. Mokry, F.M. Wang and Y.B. Yu, "Even field coding using adjacent future odd field, in addition to past even and odd fields", ISO-IEC JTC1/SC2/WG11 MPEG 91/014, May 1991.

[5] D. Anastassiou, T.H. Chiang, A. Eleftheriadis, R. Mokry, F.M. Wang, and Y.B. Yu, "Additional research results on MPEG-2", ISO-IEC JTC1/SC2/WG11 MPEG 91/131 August 1991.

[6] "Advanced Digital Television" system description, submitted to the FCC/ACATS by the ATRC (David Sarnoff Research Center, NBC, North American Philips, Thomson Consumer Electronics), February 1991.

[7] AT&T/Zenith "Digital Spectrum Compatible" technical description, submitted to the FCC, September 23, 1991.

[8] General Instrument Corp. "DigiCipher HDTV System" technical description, submitted to the FCC, June 1991.

[9] D. Anastassiou, W. B. Pennebaker and J. L. Mitchell, "Gray-Scale Image Coding for Freeze-Frame Videoconferencing", IEEE Transantions on Communications, vol. COM-34, pp. 382-394, Apr. 1986.

[10] T.H. Chiang, R. Mokry, F.M. Wang, A. Eleftheriadis, Y.B. Yu and D. Anastassiou, "Compatible Coding of Digital Interlaced HDTV Using Prediction of the Even Fields from the Odd Fields", Proceedings, Fourth International Workshop on HDTV and beyond, Torino, Italy, September 1991.

PART II: SPECIFIC ITEMS

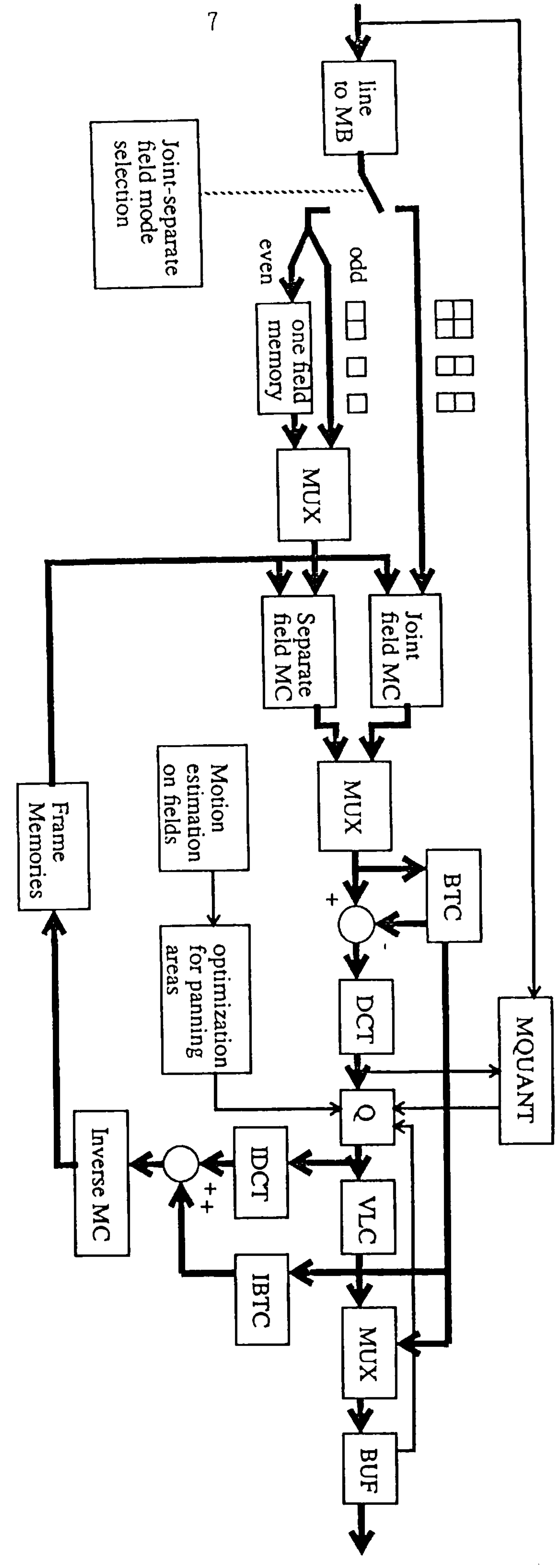
1. Block Diagram

(shown in next page)

2. Escape to BTC coding

A modified Block Truncation Coding (BTC) is applied for coding the luminance components of intra-blocks, to be used for blocks that have bilevel appearance, like text or animation material. If DCT is applied to these structures, ringing can be quite visible for the same number of bits. In the current implementation, BTC is based on 8x8 blocks. After a macroblock is identified as suitable for BTC, each block is decomposed into lower and higher mean values and a bit-map. The mean values are DPCM coded with reference to the previously coded block. The computational expense of using BTC is small, since most operations are simple.

The difference between BTC coding results and original data can be encoded in an interframe fashion. This is virtually a contour/texture coding approach to exploit the characteristics of both the human visual system and image signals. By allowing escape to BTC mode, the bitrates can be reduced by about 0.2 bits/pixel, and some sharp edges can be well preserved, e.g. in the "mobile-calendar" sequence.


line to MB
Joint-separate field mode selection
odd
even
one field memory
MUX
Joint field MC
Separate field MC
MUX
Motion estimation on fields
optimization for panning areas
Frame Memories
BTC
DCT
MQUANT
Q
IDCT
VLC
IBTC
Inverse MC
MUX
BUF

3. Coding/Decoding delay

Encoding and decoding delays are identical to those of MPEG1 for N=12 and M=3 for a 30 frames/s system.

4. VLC tables

In our implementation, variable Length Coding tables were taken from MPEG1 and not optimized for the higher resolution sequences, or for the different geometric structures of fields and frames. This has resulted in significant deterioration of coding performance, so the results do not represent the optimum performance for the proposed technique. Specifically we have used MPEG1 specifications for motion vector coding, DC luminance and chrominance, AC coefficients and coded block pattern.
Regarding MQUANT, we have allowed a one-bit code ('1') for the case of unchanged value of QP as determined by the buffer status. Otherwise, the following table of codes and actual values of QP is used:
0001 QP-8
0010 QP-12
0011 QP-16
0100 QP+12
0101 QP+16
0110 QP+20
0111 QP+24
For interpolated macroblock types, the following codewords are used:
10, 11 for IPRED coded and noncoded
010, 011 for BPRED coded and noncoded
0010, 0011 for FPRED coded and noncoded
00011 for INTRA

5. Statistics

The following figures indicate the SNR's and number of bits used in the various sequences for 4 Mbits/s coding and 9 Mbits/s coding.

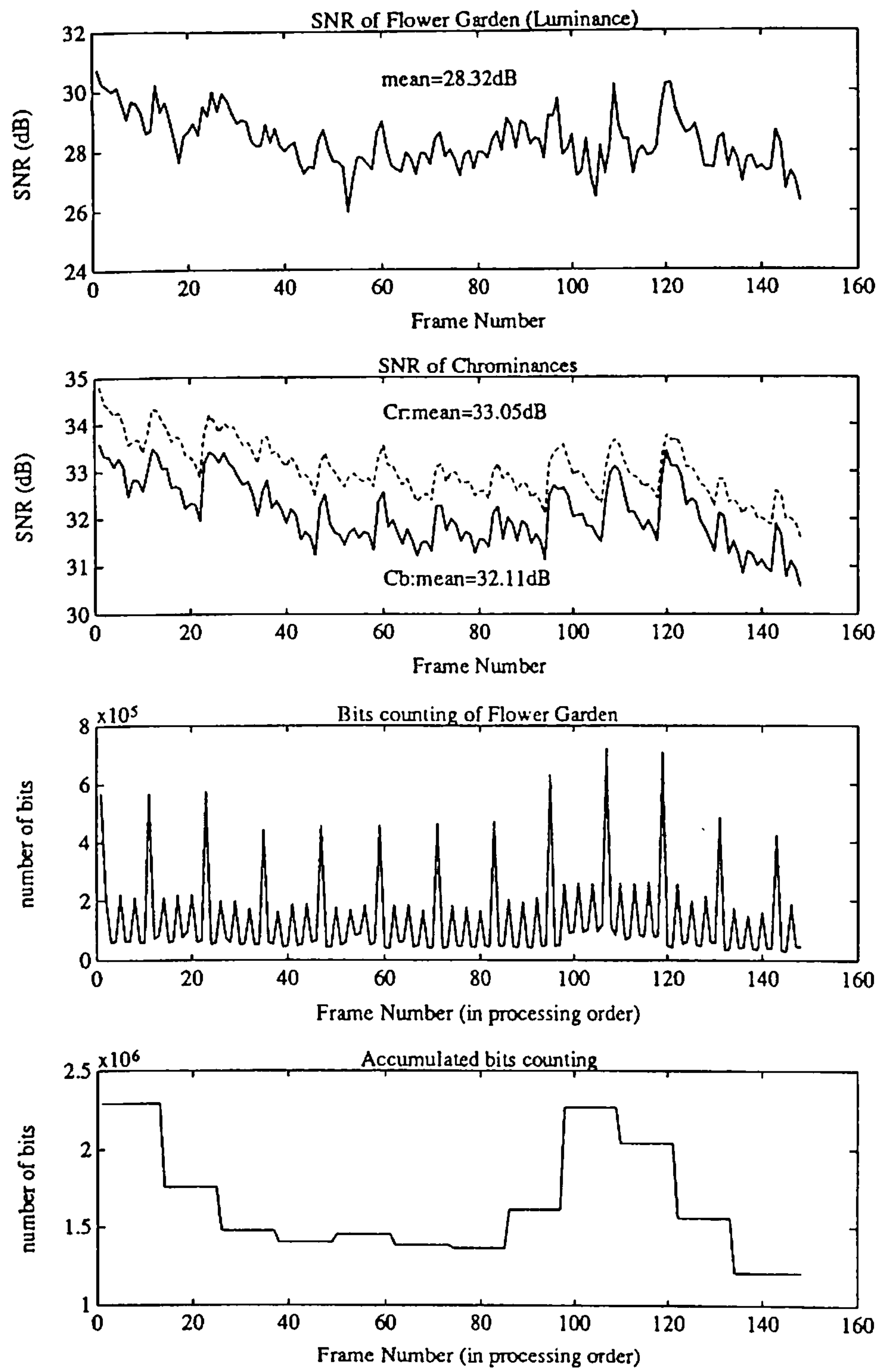


Figure 1: Statistics of Flower Garden Sequence at 4Mbits/s

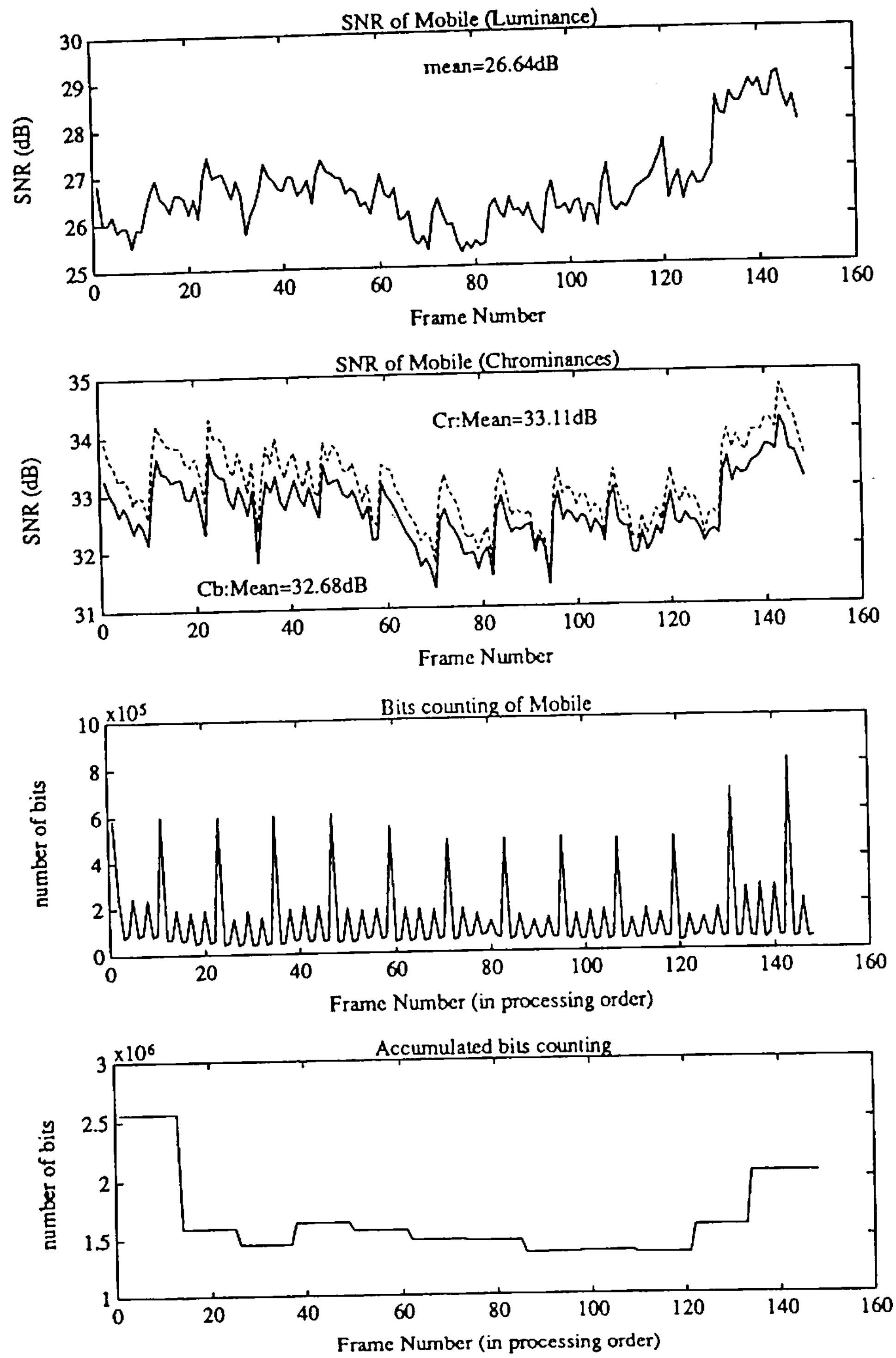


Figure 2: Statistics of Mobile and Calendar Sequence at 4Mbits/s

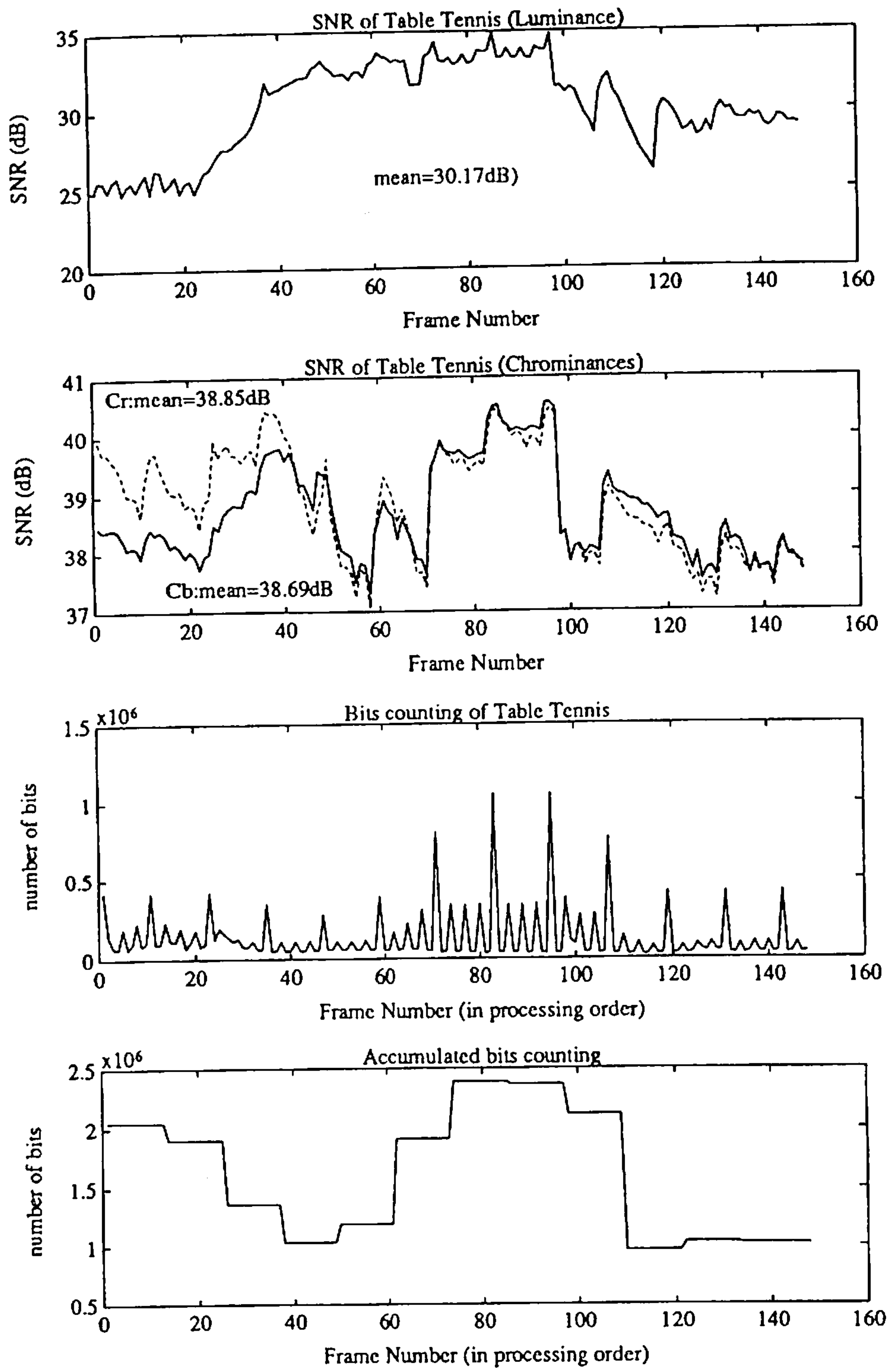


Figure 3: Statistics of Table Tennis Sequence at 4Mbits/s

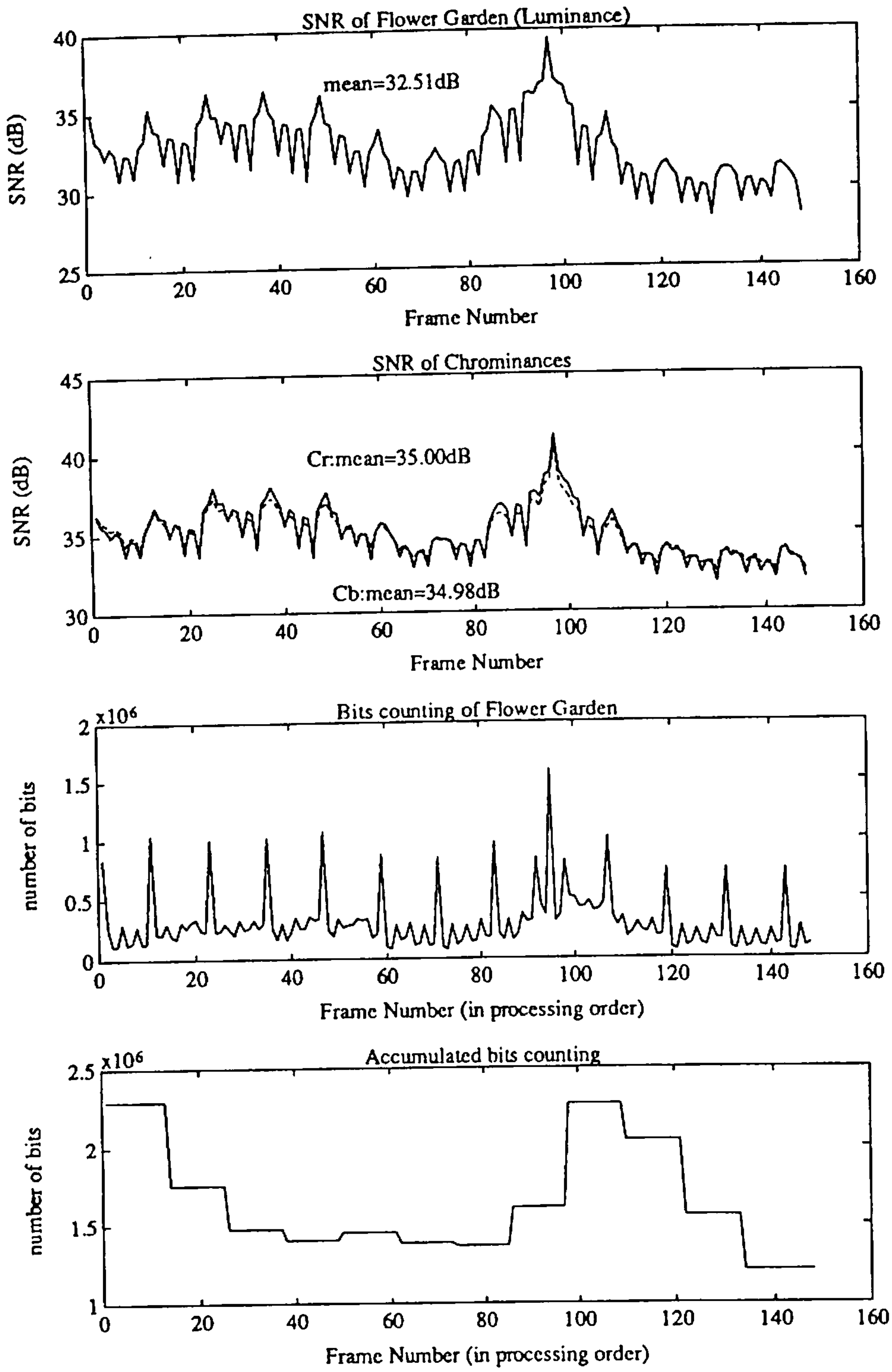


Figure 4: Statistics of Flower Garden at 9Mbits/s

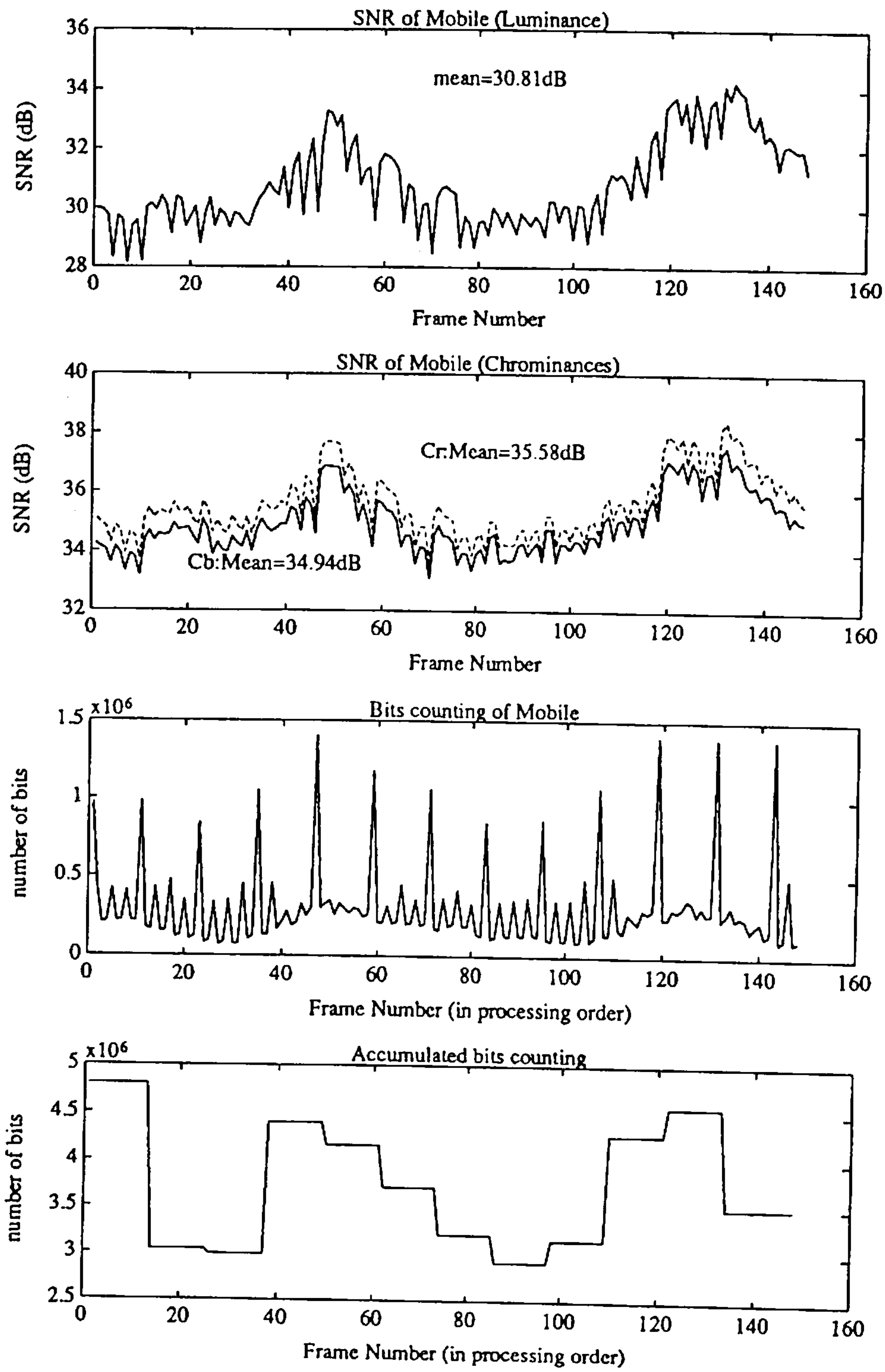


Figure 5: Statistics of Mobile and Calendar at 9Mbits/s

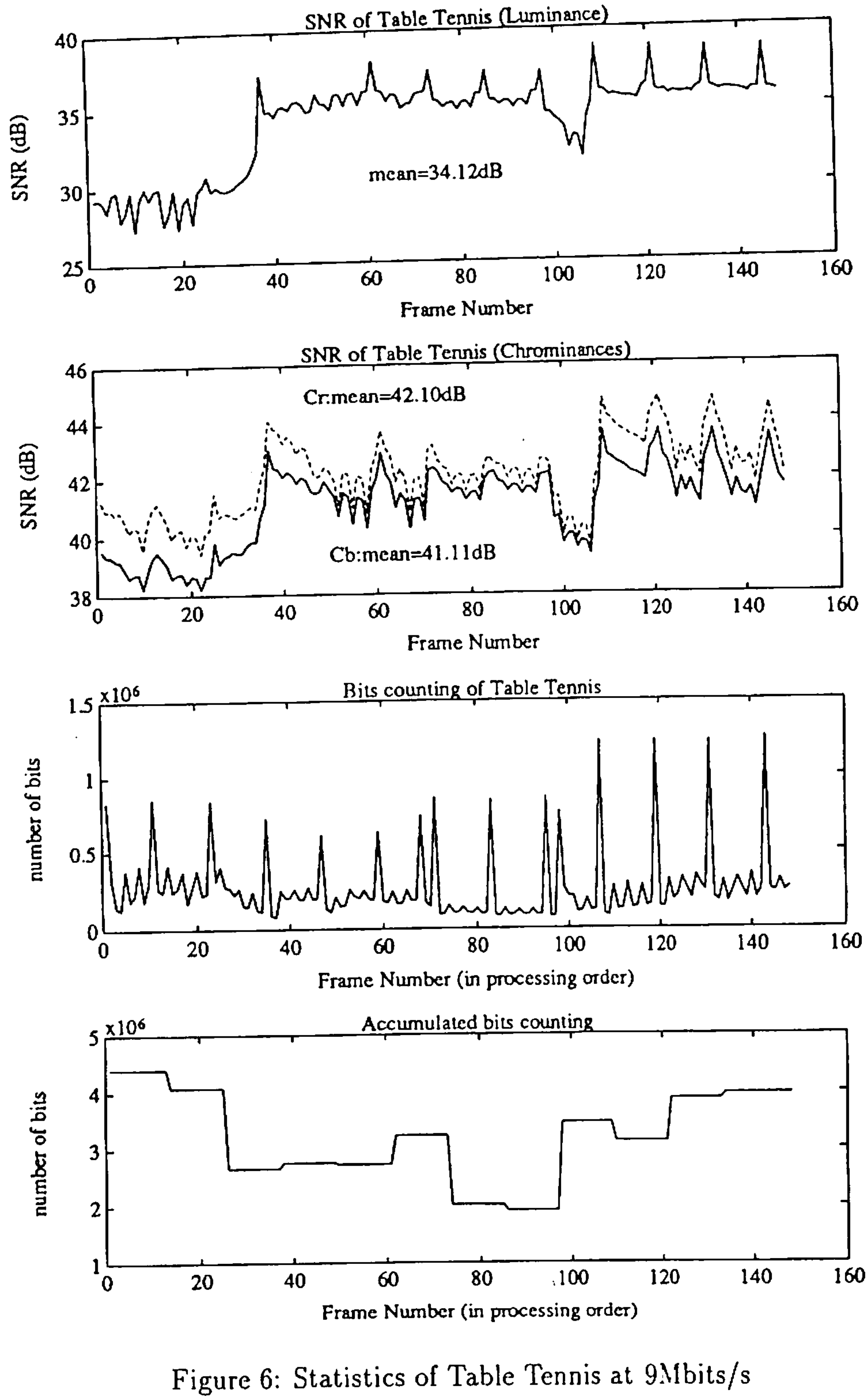


Figure 6: Statistics of Table Tennis at 9Mbits/s

6. Size of coded bitstream

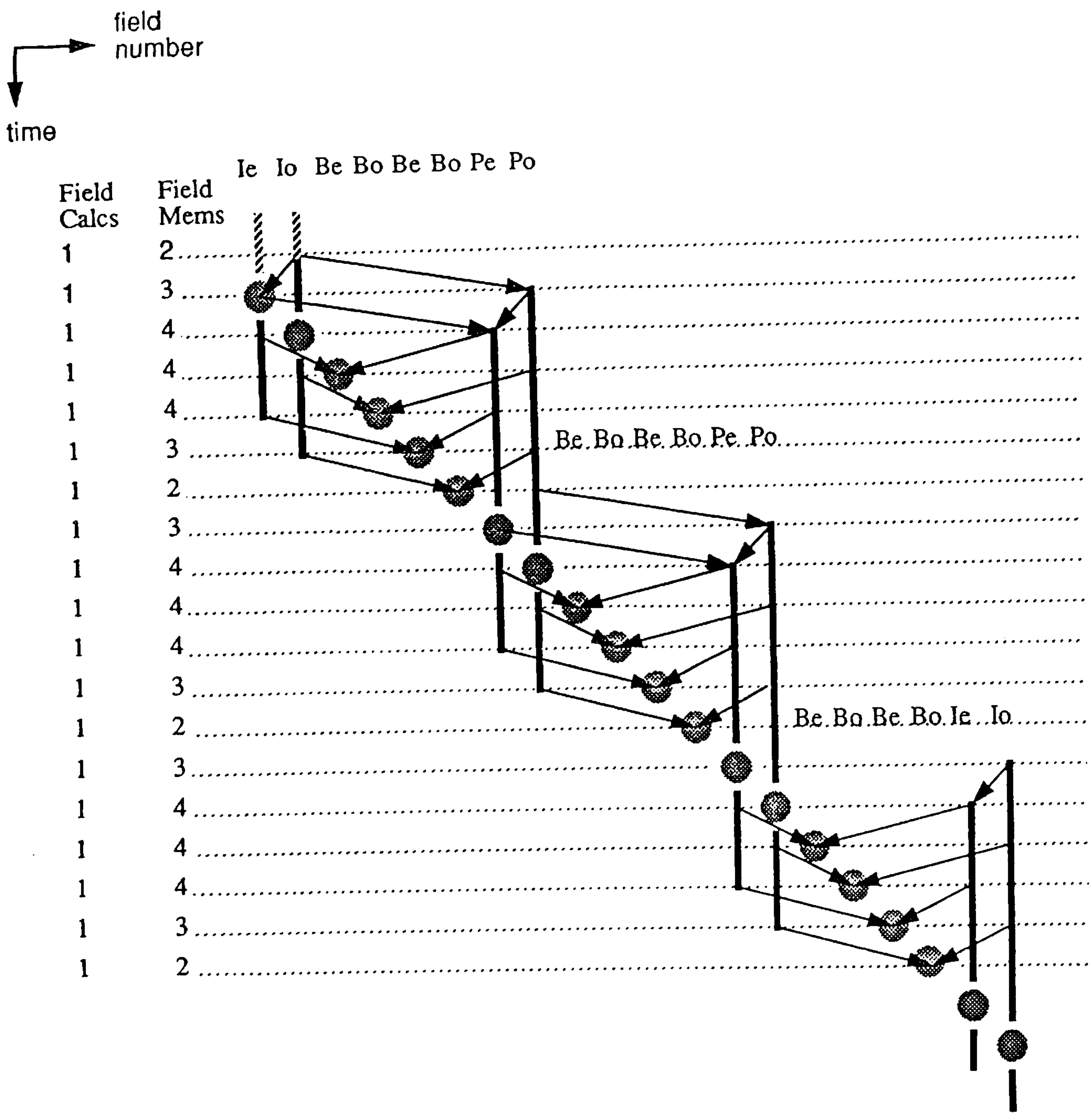
Using the "ls -l" command, the following was the count for the coded bitsream:
Mobile-Calendar at 4 Mbits/s: 19812092 bits, or 4.016 Mbits/s
Flower Garden at 4 Mbits/s: 20106770 bits, or 4.076 Mbits/s
Table Tennis at 4 Mbits/s: 19518339 bits, or 3.956 Mbits/s
Mobile-Calendar at 9 Mbits/s: 45430435 bits, or 9.209 Mbits/s
Flower Garden at 9 Mbits/s: 44661048 bits, or 9.053 Mbits/s
Table Tennis at 9 Mbits/s: 44013907 bits, or 8.922 Mbits/s

7. Hardware considerations

When fields are coded jointly as a frame, the technique becomes identical to MPEG1, for which hardware embodiment aspects have been extensively examined and are known.
Therefore, we only consider separate coding of fields. The figure in the next page describes field memory exchange. The circles are the times that the fields get sent out. The vertical bars occur during the times when the fields must be saved in memories. The arrows correspond to the prediction (motion vectors). The arrows always go exactly one step forward, because motion compensation is done in one time unit.
"Field Mems" is the number of field memories needed in the time unit. It is the number of vertical lines (including beginnings and ends) on the horizontal line. Circles at the end of a vertical line, or circles by themselves, are not included in the count, because the field can be immediately output (with only the small amount of memory needed to make the block-to-line conversion).
"Field Calcs" is the number of fields that must be calculated in the time unit. It is the number of beginnings of vertical lines on the horizontal line. In this case, a circle without a horizontal line is equivalent to the beginning of a vertical line.
There are two issues of importance:

1. Number of field memories. In our case, four such memories are needed.

2. Uniformity of Field Calcs. This is guaranteed in our case (they are all uniformly equal to 1).


field
number
time
Field
Calcs
Field
Mems
Ie Io Be Bo Be Bo Pe Po
1 2
1 3
1 4
1 4
1 4
1 3
1 2
1 3
1 4
1 4
1 4
1 3
1 2
1 3
1 4
1 4
1 4
1 3
1 2
Be Bo Be Bo Pe Po
Be Bo Be Bo Ie Io