

# Low-Complexity Video Error Concealment for Mobile Applications Using OBMA

Tanaphol Thaipanich, *Student Member, IEEE*, Ping-Hao Wu, *Student Member, IEEE*,  
and C.-C. Jay Kuo *Fellow, IEEE*

**Abstract** — *Low-complexity error concealment techniques for mobile applications are studied extensively in this paper. The boundary matching algorithm (BMA) is an attractive choice for mobile video concealment due to its low complexity. Here, we examine a variant of BMA called the outer boundary matching algorithm (OBMA). Although BMA and OBMA are similar in their design principle, it is empirically observed that OBMA outperforms BMA by a significant margin (typically, 0.5dB or higher) while maintaining the same level of complexity. In this work, we attempt to explain the superior performance of OBMA, and conclude that OBMA provides an excellent tradeoff between the complexity and the quality of concealed video for a wide range of test video sequences and error conditions.*

**Index Terms** — Error concealment, boundary matching algorithm, outer boundary matching algorithm, H.264<sup>1</sup>

## I. INTRODUCTION

Due to the advancement of video coding technologies and standards, wireless video applications have become more and more popular in portable consumer electronics, including mobile TV and 3G cellular phones. One common problem in transmitting compressed video over unreliable channels in mobile and/or wireless environments is video quality degradation due to transmission errors. Techniques for combating transmission errors can be classified into three categories [7] based on the role of the encoder and the decoder. Only the encoder is used to handle transmission errors in forward error prevention. Error concealment (EC) is a post-processing technique employed at the decoder side as shown in Fig 1. When the encoder and the decoder work together in error handling, it is called interactive error control.

Among these three categories, error concealment (EC) has the least restriction in implementation and has been studied extensively in the last 20 years. The basic idea of EC is to recover the lost data from decoded video by exploiting its spatial and/or temporal correlation [9,10]. The original EC concept can be traced back to the problem of partial lost in still image [11,12]. Then, it has been gradually applied to video coding. Recent EC developments tend to utilize unique features provided by the emerging H.264/AVC video coding standard, *e.g.*, the variable block size and multiple reference

frames to improve concealed video quality. For example, the coding mode of neighboring macroblocks was used in [13] to estimate the coding mode of the lost macroblock so that it can be concealed on the basis of different block sizes. However, the computational complexity of most EC methods proposed by academia is too high to be practical in mobile applications.

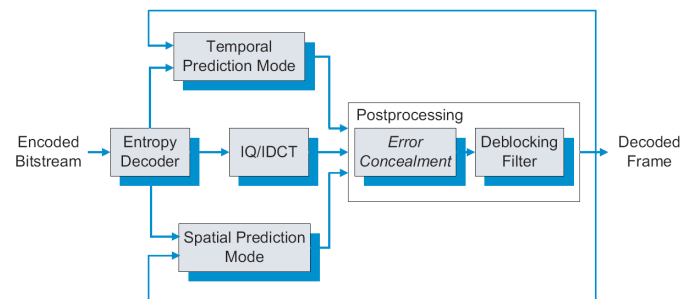


Figure 1: The block-diagram of a decoder with an EC module.

We investigate a low complexity EC algorithm that offers excellent performance in this work. The boundary matching algorithm (BMA) is a good EC candidate for mobile video due to its low complexity. Here, we examine a variant of BMA called the outer boundary matching algorithm (OBMA). Although BMA and OBMA are similar in their design principle, it is empirically observed that OBMA outperforms BMA by a significant margin (typically, 0.5dB or higher) while maintaining the same level of complexity. To the best of our knowledge, such a performance gap has not yet been well explained in the literature. In this work, we offer more insights to the superior performance of OBMA and provide extensive experimental results to demonstrate its advantage for a wide range of test video sequences and error conditions.

The rest of this paper is organized as follows. BMA, OBMA and other derived low complexity EC methods are discussed in Sec. II. The performance evaluation of several low complexity EC techniques is conducted in Sec. III. The superior performance of OBMA is explained in Sec. IV. Two extensions of OBMA and their associated experimental results are presented in Sec. V. The concept of using the flexible macroblock ordering (FMO) to enhance EC performance is discussed in Sec. VI. Finally, concluding remarks are given in Sec. VII.

## II. BMA AND OBMA METHODS

BMA and OBMA methods rely on the boundary smoothness assumption to conceal the lost MB information. The candidate motion vector (MV) set consists of its eight neighboring MVs denoted by MV0, MV1, ..., MV7, and the

Part of this work was presented at ICCE2008, Las Vegas, NV, USA

Tanaphol Thaipanich, Ping-hao Wu and C.-C. Jay Kuo are with the Ming Hsieh Department of Electrical Engineering and the Signal and Image Processing Institute, University of Southern California, Los Angeles, CA 90089 USA (E-mails: thaipani@usc.edu, pinghaow@usc.edu, and cckuo@sipi.usc.edu)

Digital Object Identifier: n/a

zero motion vector (ZMV) denoted by MV8, as shown in Fig. 2.

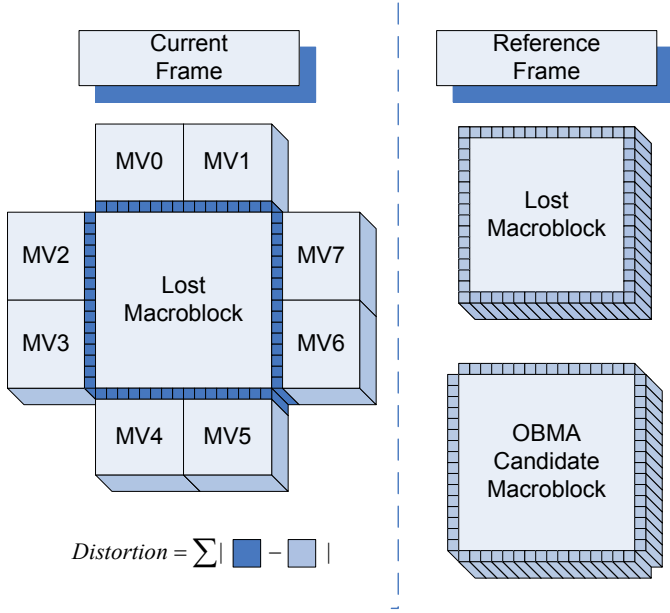


Figure 2: Distortion computation in BMA and OBMA.

#### A. BMA

For each MV candidate, BMA constructs a candidate MB in the reference frame by backward tracing. To recover the lost MB, the mean of absolute differences (MAD) between the boundary of candidate MBs in the reference frame and the neighboring MBs of the lost MB in the current frame is computed as

$$D_n^{BMA} = \frac{1}{N} \left| \sum_{\forall i \in I} P_i^N - \sum_{\forall j \in J} P_j^C \right|, \quad (1)$$

$$n_{opt} = \arg \min_n (D_n^{BMA}), \quad (2)$$

where  $D_n^{BMA}$  is the MAD of the  $n^{th}$  candidate MB with  $n=0,1,...,8$ ,  $P_i^N$  and  $P_j^C$  denote boundary pixel values of neighboring MB  $i$  and candidate MB  $j$ , respectively, and  $I$  and  $J$  are sets of corresponding boundary pixel pairs as shown in Fig. 2 and  $N$  is the total number of boundary pixels. For example,  $N$  is  $16 \times 4 = 64$  for a MB of size  $16 \times 16$ . Then, the candidate MB that yields the smallest distortion is selected as the best candidate ( $n_{opt}$ ) and used to conceal the lost MB.

#### B. OBMA

Instead of using adjacent pixel values for distortion computation as done in BMA, OBMA utilizes a linear translational model to conceal a lost MB as shown in Fig. 2 by assuming that the lost MB can be reconstructed from the reference frame with constant motion in both magnitude and direction. OBMA adopts the same candidate set  $\{MV0, MV1, \dots, MV7, MV8\}$ , where MV8 is the ZMV, but a different distortion computation as

$$D_n^{OBMA} = \frac{1}{N} \left| \sum_{\forall i \in I} P_i^N - \sum_{\forall k \in K} P_k^O \right|, \quad (3)$$

$$n_{opt} = \arg \min_n (D_n^{OBMA}), \quad (4)$$

where  $D_n^{OBMA}$  is the MAD of the  $n^{th}$  OBMA candidate MB,  $P_k^O$  is the pixel value of the outer boundary of candidate MB  $k$ , and  $I$  and  $K$  are sets of corresponding boundary pixel pairs as shown in Fig. 2. The candidate MB that has the smallest distortion is selected as the best candidate. A concept similar to OBMA, called DMVE, was first proposed in [1]. Unlike DMVE, OBMA employs only one layer of the outer boundary and uses all adjacent neighboring MBs in the matching process.

#### C. Four Other Derived EC Algorithms

The other four low-complexity EC techniques under our consideration are: zero motion vector (ZMV), average motion vector (AMV), overlapped motion vector (OMV) and its modified version (OMV+). ZMV uses the corresponding MB in the reference frame for concealment while AMV reconstructs the lost MB based on the MV obtained from averaging all neighboring MVs. OMV restores the lost MB by combining three intermediate subblocks obtained from the side match computation and the MVs of vertically and horizontally adjacent neighboring subblocks. The side match computation used in OMV is essentially a modified version of BMA's matching function. In [4], OMV utilizes a weighting matrix for subblock merging to produce the concealed MB with less blocky artifact along the boundary of subblocks. However, simple averaging is used in our work since it requires less complexity and the deblocking filter in the decoder can proficiently remove some of these effects. OMV+ is similar to OMV. Their main difference is that OMV and OMV+ adopt modified BMA and OBMA matching functions, respectively.

### III. PERFORMANCE EVALUATION

We apply all six EC algorithms discussed in Sections II and III to three video sequences (City, Crew and Harbour) under QCIF ( $176 \times 144$ ), CIF ( $352 \times 288$ ) and 4CIF ( $704 \times 576$ ) resolution. For each sequence, the first 100 frames are used and encoded with IPPP format. The error is modeled as uniformly distributed random with 10% MB loss ratio. For each case, 10 error patterns are employed and average results are reported in Fig. 3. The encoder and decoder module are based on H.264/AVC reference software version 11.0 [2]. BMA and OBMA are very similar to each other and have exactly the same complexity.

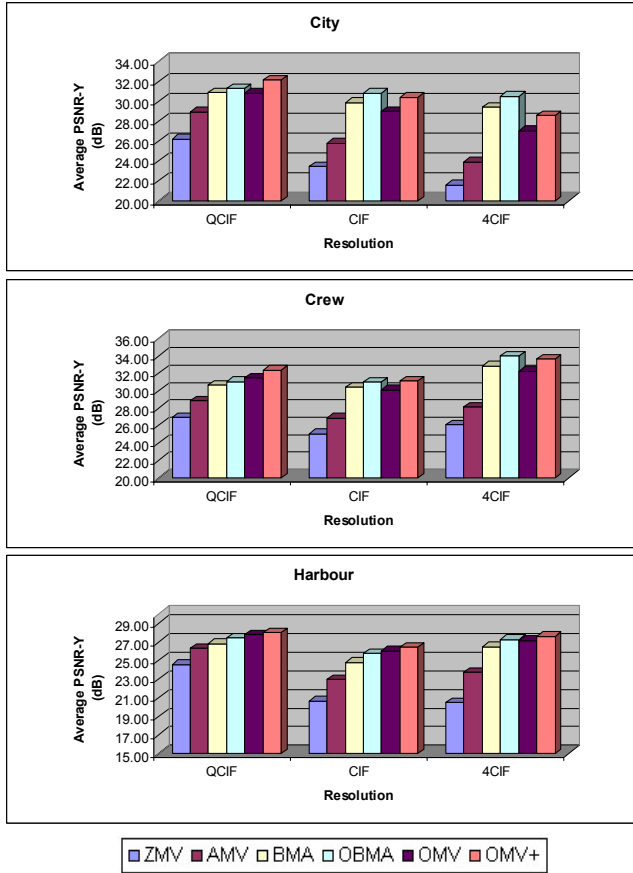


Figure 3: Performance comparison of six EC algorithms.

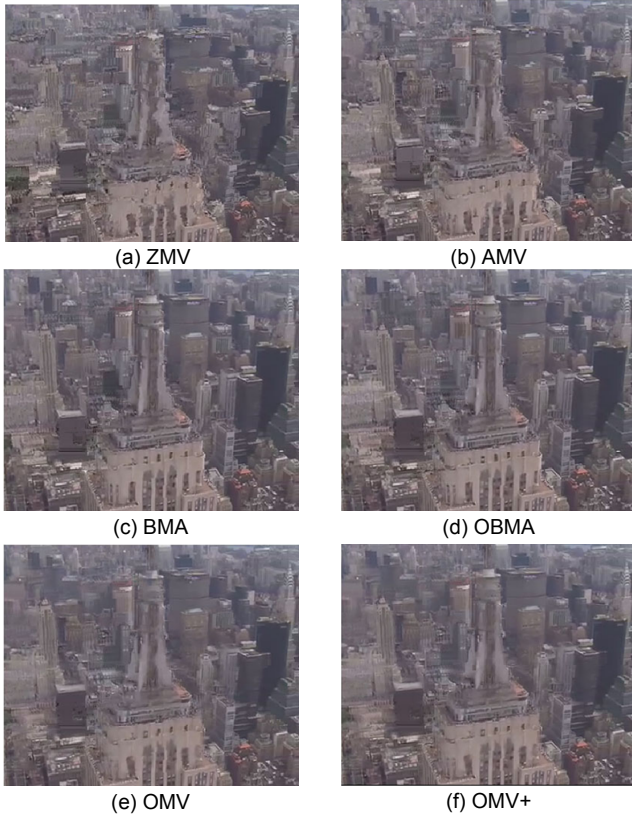


Figure 4: Subjective quality comparison – frame no. 100 of city sequence (CIF) with a loss rate of 10%.

As shown in Fig. 3, OBMA outperforms BMA by a large margin. (They correspond to the 4<sup>th</sup> and the 3<sup>rd</sup> bars in the chart for a given error rate.) Its performance is approximately 0.51, 0.82 and 1.01 dB better than BMA in QCIF, CIF and 4CIF sequences, respectively and is much better than ZMV and AMV. Performance of BMA and OBMA is getting better by comparison for higher video resolutions since they tend to have better spatial features in the matching process. In the low resolution case, OMV yields the same level of quality as OBMA but its performance suffers from ambiguity introduced by the averaging operation. With the modification, OMV+ gains approximately 0.80, 0.96 and 1.13dB improvements from OMV in QCIF, CIF and 4CIF sequences, respectively. This enhancement is mainly due to the performance boost from the use of the OBMA-like matching function. Based on this result, OBMA is very suitable for practical applications since it has an excellent performance while demanding a low complexity. In the next section, we explain the concepts behind performance of BMA and OBMA as well as the reasons why OBMA performs much better than BMA despite its similarity.

#### IV. EXPLANATION OF SUPERIOR OBMA PERFORMANCE

The difference between BMA and OBMA lies in their distortion function. The inner and outer boundaries are used in the matching process for BMA and OBMA, respectively. We can explain the superior OBMA performance from three viewpoints as detailed below.

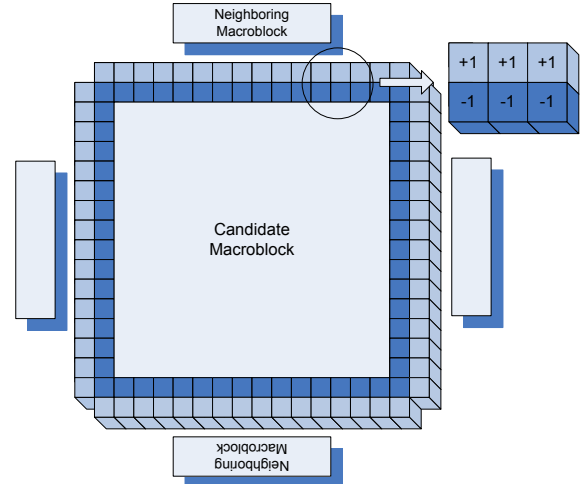


Figure 5: The distortion function of BMA resembles gradient-based horizontal and vertical edge detectors.

##### A. Edge Detection Effect of BMA Distortion Computation

As shown in Fig. 5, BMA's distortion function is similar to the gradient-based horizontal and vertical edge detectors with simple differences. Due to this resemblance, BMA's distortion function actually penalizes an edge located in a block boundary. As shown in Fig. 6, when an edge lies across a block boundary, it results in an uneven surface between the outer and the inner boundaries, which leads to an increased MAD value. As a result, the block of ground truth is less likely to be selected. Except edges that are perpendicular to

boundary, all others across the block boundary tend to hurt the BMA performance. By using the outer boundary, OBMA's distortion function incorporates the edge information in the vicinity of a block boundary into its matching decision accurately. Consider an example given in Fig. 7, which has three matching candidates. Since the 2<sup>nd</sup> and the 3<sup>rd</sup> candidates have inconsistent edge information with the neighboring MB, their MAD calculated by OBMA's distortion function is higher. The original lost MB will give zero value if it is in the candidate set. In other words, the edge in the vicinity of a block boundary helps select the right MB by OBMA.

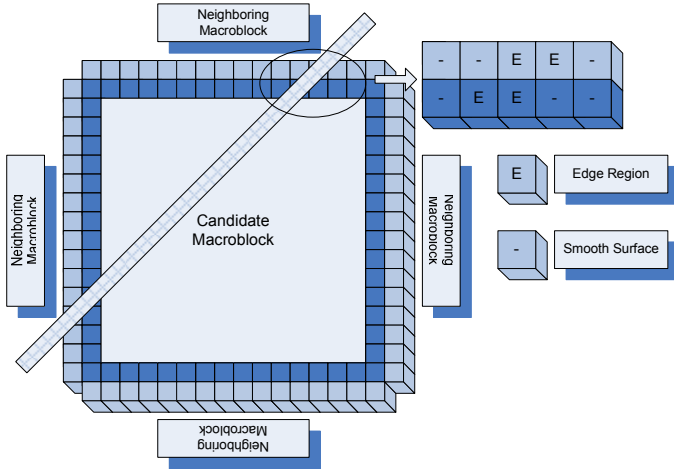


Figure 6: The edge in the vicinity of the block boundary is penalized by BMA.

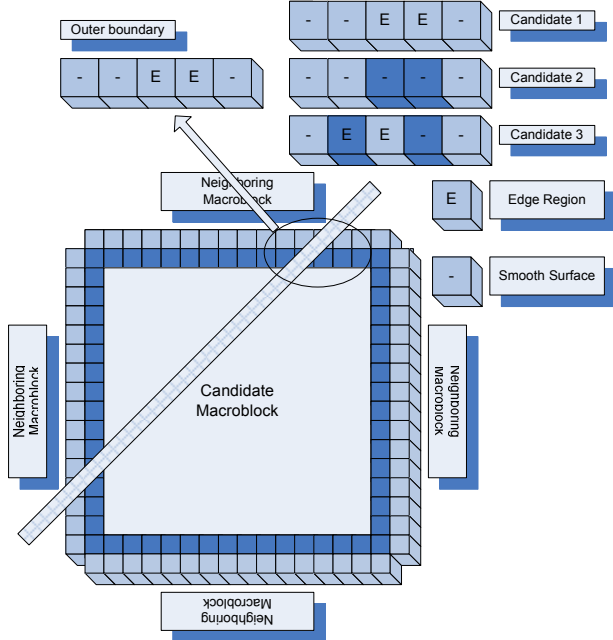


Figure 7: The edge in the vicinity of a block boundary helps select the right MB by OBMA.

### B. Partial Block Matching

The superior performance of OBMA can be explained by the block-based motion estimation concept as well. That is, OBMA can be viewed as a special type of block matching. In video encoding, each MB is compared with all possible candidate MBs in the search range of the reference frame to

determine the best motion vector that yields the minimum residual. The traditional motion estimation is to find the closest match to a block of size 16x16 using the spatial information of all pixels inside the block. In contrast, OBMA performs a search to find the best match to a block of size 18x18 using the spatial information of all pixels located at the outmost layer.

### C. Effect of Error Types

The results shown in Fig. 8 are obtained from testing BMA and OBMA over ten different video sequences. The experiment setting is identical to the one presented in Fig. 3. In most video sequences, OBMA outperforms BMA by a large margin. There are however three cases (i.e. Football, Ice and Soccer) that they offer about the same level of quality. This can be explained as follows. In the encoder, the motion estimation algorithm may yield an inaccurate motion vector (MV) when dealing with fast motion sequences and result in a larger residual. Consequently, the performance of OBMA is also limited by fast motion sequences such as the above three image sequences. On the other hand, OBMA can handle sequences of moderate motion with strong edges and textures since it can use this information to enhance the matching decision. This phenomenon has been observed from sequences such as mobile, bus and city. To confirm the strengths and weaknesses of OBMA, we test BMA and OBMA with selective error patterns instead of uniformly distributed random errors. The purpose is to observe the performance of each error concealment technique under various error conditions. In particular, we manually assign erroneous MBs to different regions. These two error assignment methods are describe below

1. *Type I errors*: Missing MBs are located in regions consisting of strong edges, textures and smooth areas. Examples of this type include background, static objects, slow moving objects, clear object contours, etc
2. *Type II errors*: Missing MBs are located in regions of fast motion objects, highly deformed regions such as mouths, eyes, etc.

Based on the arguments given above, we expect OBMA to perform well for type I errors but not for type II errors due to the poor performance of motion search in the latter case. This conjecture is confirmed by experimental results shown in Fig. 9. The experiment setting is the same as that given in Fig. 8 except that only 5% MB loss is tested due to a limited number of classified MBs. We see clearly that the performance gap between OBMA and BMA is quite large for type I errors but narrows significantly for type II errors. Although OBMA yields mediocre quality in the Football, Ice and Soccer sequences in Fig. 8, it performs very well in the same three sequences when only type I errors are considered in Fig. 9. On the other hand, the performance of OBMA degrades significantly for type II errors for the Foreman and Harbor sequences, even though the overall results were good in Fig. 8. Please also note that there are no sufficient classified MBs to synthesize type II errors for the Carphone sequence and type I errors for the Bus and the City sequences. Thus, their results



type I errors for the Bus and the City sequences. Thus, their results are skipped in Fig. 9. To conclude, OBMA will significantly outperform BMA if the motion vector provides an effective temporal prediction.

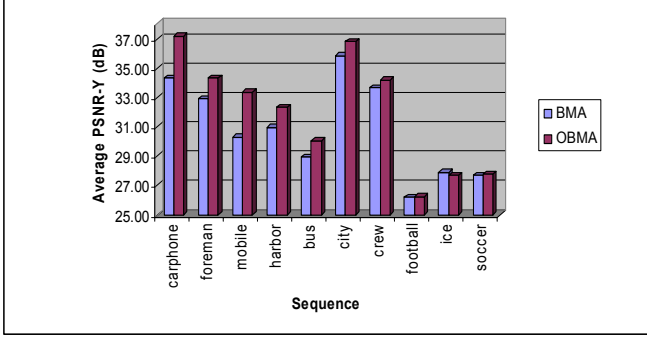


Figure 8: BMA and OBMA performance comparison for different video sequences with an MB loss rate of 10%.

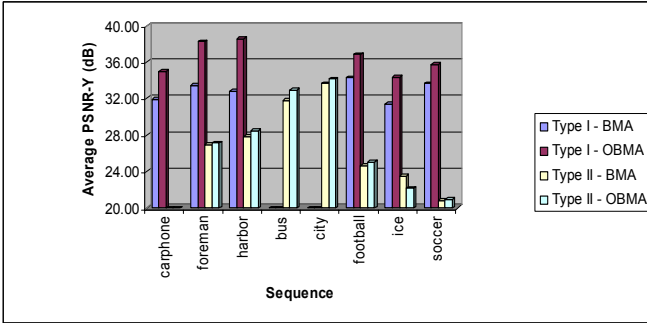


Figure 9: Performance comparison of BMA and OBMA against two error types.

## V. TWO EXTENSIONS OF OBMA

Two extensions of OBMA are considered in this section. The first one is the search pattern extension. With this technique, OBMA can trade additional complexity for improved performance. Another extension is to use multiple overlapped layers, which is an extended version to an idea described in [1].

### A. OBMA with Refined Local Search

OBMA in its original form only searches the reference frames with a small set of neighboring motion vectors (MVs). The overall performance of OBMA can be improved by increasing the number of MV candidates at the cost of higher computational complexity. To evaluate the trade-off between the concealed video quality and the complexity, we compare the PSNR results as well as the complexity for the following three search patterns. Four variations are examined in each case.

1. OBMA with full search (OBMA-FS). OBMA-FS uses all possible candidates in a large search area pointed by the estimated MV of the lost MB. The search is performed with a quarter-pel resolution and can be extended to multiple reference frames.
  - OBMA-FS5. We set the number of reference frames to 5 and the search range to 16x16. It includes all possible candidates.

- OBMA-FS1-R16, OBMA-FS1-R8 and OBMA-FS1-R4. We use only 1 reference frame and set the search range to 16x16, 8x8 or 4x4, respectively
- The complexity of OBMA-FS can be computed by

$$N_C = N_R * (2 * R_F * 4 + 1) * (2 * R_F * 4 + 1) \quad (5)$$

Where  $N_C$  and  $N_R$  are the numbers of candidates and reference frames, respectively and  $R_F$  is the search range in pixels.

2. OBMA with refined local search (OBMA-RS). OBMA obtains matching candidates from a small search region pointed by the MV of neighboring MBs. The complexity is drastically reduced from the previous case since the search region is much smaller.

- OBMA-RS2. It uses a search range of 2x2 with the quarter-pel resolution.
- OBMA-RS1-QP, OBMA-RS1-HP and OBMA-RS1-FP: They use search range of 1x1 with quarter-pel, half-pel and full-pel resolutions, respectively.

The complexity of OBMA-RS is give by

$$N_C = 8 * (2 * R_S * R_Q + 1) * (2 * R_S * R_Q + 1) \quad (6)$$

Where  $R_S$  is the search range of a small search region,  $R_Q$  is the search resolution equal to 4, 2 and 1 for quarter-pel, half-pel and full-pel resolution, respectively

3. OBMA with selective search (OBMA-SS). OBMA-SS is similar to OBMA-RS except that it only considers a small search region pointed by the MV of neighboring MBs that gives the minimal MAD. It can be viewed as a two-stage operation.

- OBMA-SS2. It uses a search range of 2x2 with the quarter-pel resolution.
- OBMA-SS1-QP, OBMA-SS1-HP and OBMA-SS1-FP. They use a search range of 1x1 with quarter-pel, half-pel and full-pel resolutions, respectively.

The complexity of OBMA-SS can be calculated via

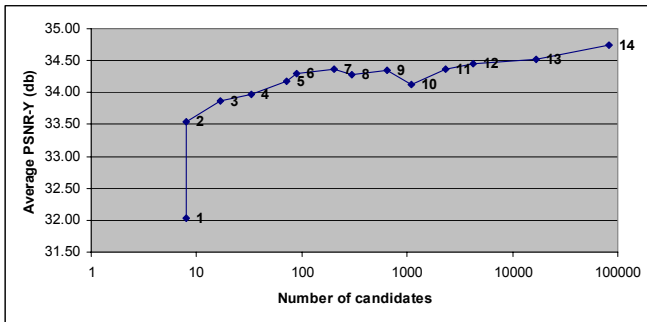
$$N_C = (2 * R_S * R_Q + 1) * (2 * R_S * R_Q + 1) + 8 \quad (7)$$

The original BMA and OBMA use a set of eight neighboring MBs for matching. Thus, their complexity is equal to 8 candidates. The complexity of all techniques and their performance are shown in Table 1. The plot of quality versus complexity is given in Fig. 10. The experimental setting is similar to that in Fig. 3 with MB loss rate equal to 10%. OBMA-FS5 outperforms OBMA by 1.3 dB at the cost of 83,205 matching candidates. This is not suitable in mobile video applications. In refined and selective search cases, the performance of OBMA is not affected by the reduction in the search range from 2x2 to 1x1. Besides, the reduction of search resolution from quarter-pel to half-pel slightly decreases the overall performance. In general, to increase the number of matching candidates gives rise to quality improvement as observed in Fig. 10 as shown by the positive slopes between two adjacent points. However, there are some segments that

have a negative slope such as the transition from OBMA-RS1-QP (9) to OBMA-FS1-R4 (10) and from OBMA-RS1-HP (7) to OBMA-SS2 (8). Overall, we see that OBMA with a selective search pattern offers an excellent trade-off between quality and complexity.

**Table 1: Complexity and quality comparison of several EC methods (the index is used in Fig. 10).**

Error concealment technique			Average PSNR-Y	Complexity	Index
BMA			32.04	8	1
OBMA			33.54	8	2
OBMA-FS	FS5		34.74	83205	14
	FS1	R1	34.52	16641	13
		6			
		R8			
		R4	34.13	1089	10
OBMA-RS	RS2		34.36	2312	11
	RS1	QP	34.36	648	9
		HP	34.36	200	7
		FP	34.17	72	5
	OBMA-SS	SS2		34.27	297
SS1		QP	34.29	90	6
		HP	33.96	33	4
		FP	33.86	17	3

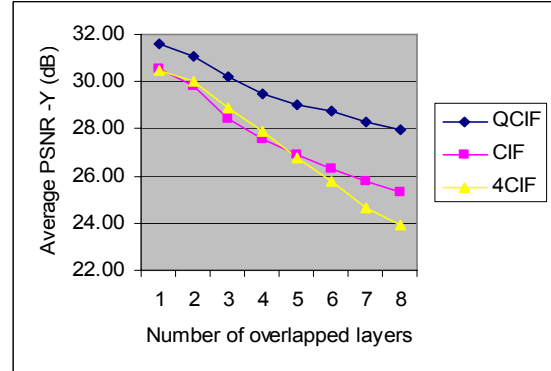


**Figure 10: Quality and complexity comparison of various EC methods given in Table 1, where the number is the index of each method.**

### B. Multiple Boundary Layers

In this extension, we may increase the number of boundary layers from one to multiple layers. The outer boundary layer number ranges from 1 to 8 in Fig. 11 and three different resolutions (QCIF, CIF and 4CIF) of the City sequence was tested. The sequence consists of 100 frames and it is encoded in the IPPP format. The MB loss rate is 10%. It turns out that the single-layer OBMA yields the best results for all three video resolutions. Its performance is decreasing when more outer layers are used in the distortion function. To explain this, we interpret OBMA from the motion estimation viewpoint. The original OBMA is equivalent to performing a search similar to motion estimation with a block of size 18x18 using the spatial information of all pixels located at the outmost boundary. When  $N$  overlapped layers is used, the size of searching block becomes  $(2N+16) \times (2N+16)$  and this

corresponds to motion estimation with a larger block size. When the block size is too large, the accuracy of block-based motion estimation can be hurt. As discussed before, a larger residual is equivalent to higher distortion in a restored MB. Actually, we may reduce the block size to achieve better quality. This can be done by performing OBMA in the subblock level (block size of 8x8) as suggested in [5] and [8], which is called the refined estimation with a smaller block size. (RBMA) and proven to be better than OBMA with block size 16x16.



**Figure 11: The performance of OBMA with a different number of outer boundary layers used in the matching process.**

## VI. BMA/OBMA WITH FMO

The flexible macroblock ordering (FMO) is one of many tools used to enhance the error resilience of coded video in H.264/AVC. The main idea behind is to assign a macroblock to one particular slice group via the Macroblock-to-slice Allocation map (MBA map). It was shown in [19] that FMO introduces a small amount of computational overhead and implementation complexity. As discussed earlier, many EC algorithms rely heavily on the spatial information to conceal the lost data. With a well-designed FMO pattern, a lost macroblock may still have some available spatial neighboring information after experiencing a lost slice.

In H.264/AVC, FMO consists of 7 different patterns as shown in Fig. 12. Each FMO pattern is unique and designed to work in different applications. Types 0 and 1 are known as interleaved and scattered patterns, respectively. Type 0 places adjacent macroblocks into a slice while Type 1 distributes the macroblock selection throughout the entire frame in a checkerboard pattern. Type 2 uses several marked rectangular regions to define a slice, which can be used in an application where some areas of the video frame are independent or more important than the rest. Types 3-5 are more dynamic in the sense that their slice pattern changes over time to accommodate more advanced applications. Finally, users can define their preferred FMO pattern in Type 6.

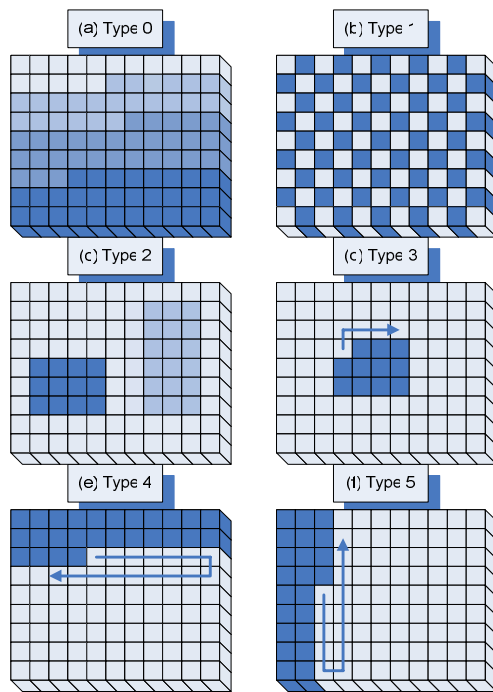


Figure 12: The FMO patterns offered in H.264/AVC, where each color represents a different slice group [16].

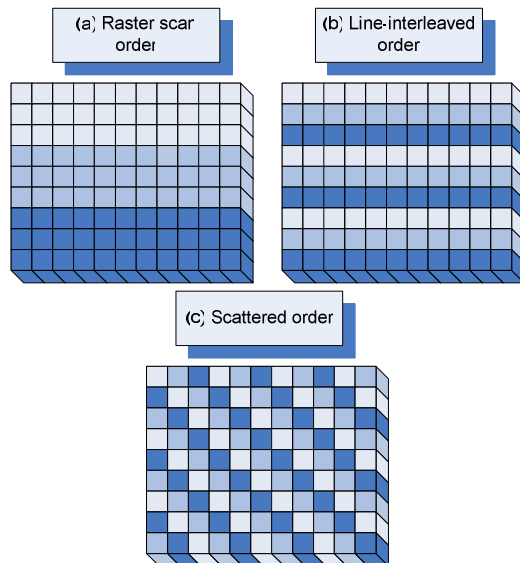


Figure 13: An FMO pattern under consideration, where each color represents a different slice group.

Table 2: Burst error conditions, where the error condition index is used in Figures 14 and 15.

Error condition	Error probability	Burst length (slice)	No. of slice per frame
T1	0.05	1	3
T2			5
T3			8
T4		2	3
T5			5
T6			8

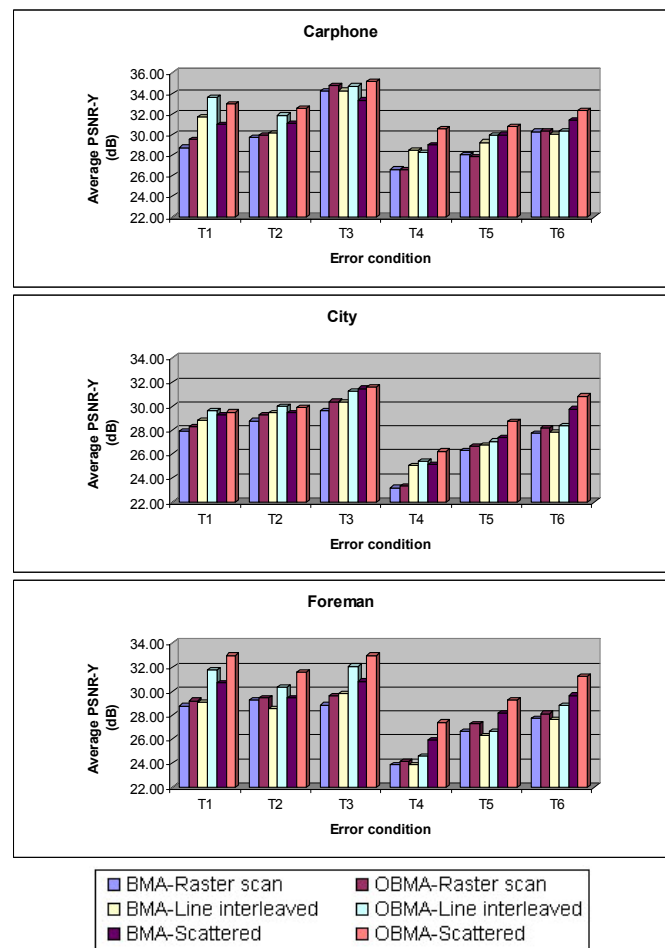


Figure 14: Performance comparison of three selected FMO patterns under various error conditions and video sequences.

In our studies, we compare three different FMO patterns (*i.e.*, the raster scan, the line-interleaved and the scattered order) as shown in Fig. 13 and various testing error conditions are given in Table 2. The burst length is used to indicate the number of consecutive lost slices. It is limited to 2 since we want to avoid the situation where the entire frame is lost. In addition, H.264 limits the number of slices per frame to 8 to avoid a complicated macroblock assignment scheme. In the experiment, we apply BMA and OBMA to three QCIF video sequences (*i.e.*, carphone, city and foreman). For each sequence, the first 100 frames are encoded with the IPPP format. 10 error patterns are employed and the averaged results are reported in Fig. 14. With the line-interleaved pattern, BMA gains 1.40 and 1.24 dB in T1 and T4 while OBMA obtains 2.68, 1.40 and 1.16 in T1, T4 and T2, respectively. The line-interleaved pattern slightly affects the performance of T3 and T6 since the lost macroblock only gains a small number of additional reliable neighboring macroblock. However, the performance of both BMA and OBMA in T1 and T4 is drastically improved due to a large number of additional reliable neighboring macroblocks. As shown in Fig 14, we see that the scattered pattern outperforms the line-interleaved pattern in all error conditions. As compared with the line-interleaved pattern, BMA and OBMA

gain an additional improvement of 0.87 and 1.23 dB, respectively, using the scattered pattern.

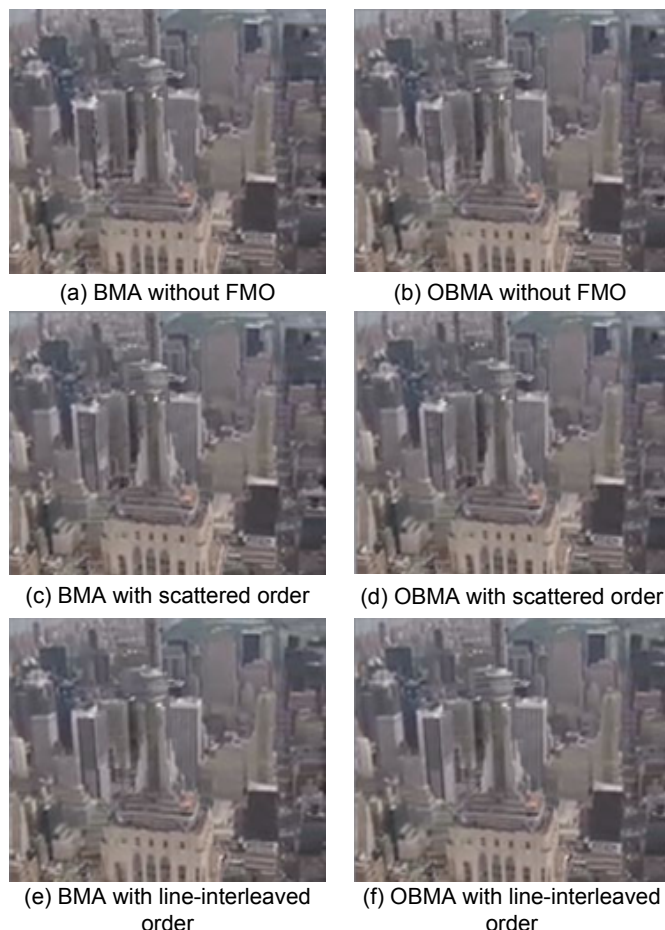


Figure 15: Subjective quality comparison – frame no. 30 of city sequence (QCIF) with error condition T6.

## VII. CONCLUSION

Several low-complexity EC techniques based on the boundary matching criteria were studied extensively in this work. Extensive experiment results were given to demonstrate the large performance gap between BMA and OBMA. This performance gap was explained from several different angles. Two extensions of OBMA and the use of FMO to enhance its performance were discussed to enhance the understanding of OBMA. To conclude, OBMA is suitable for mobile video applications since it has an excellent trade-off between complexity and good concealed video quality. Besides, it can be used to enhance other EC techniques that rely on the block-matching procedure as observed in the OMV+ algorithm.

## ACKNOWLEDGMENT

This work was supported by an industrial grant from Himax Technologies, Inc. (Tainan, Taiwan).

## REFERENCES

- [1] J. Zhang, J. F. Arnold, and M. R. Frater, "A cell-loss concealment technique for MPEG-2 coded video," IEEE Transactions on Circuits and Systems for Video Technology, vol. 10, no. 4, pp. 659-665, June 2000
- [2] JM Reference Software, version JM 11.0. August 2006
- [3] A. Tourapis, K. S  uhrling and G. Sullivan, "Revised H.264/MPEG-4 AVC reference software manual," Joint Video Team, Doc. JVT-Q042, October 2005
- [4] M.-J. Chen, L.-G. Chen and R.-X. Chen, "Error concealment of lost motion vectors with overlapped motion compensation," IEEE Transactions on Circuits and Systems for Video Technology, vol. 7, no. 3, pp. 560-563, June 1997
- [5] T. Chen, X. Zhang, and Y.Q. Shi, "Error concealment using refined boundary matching algorithm," Proceedings of ITRE2003, pp. 55-59, 11-13 August, 2003
- [6] VCEG-N62 - Non-normative error concealment algorithms. ITU - Telecommunications Standardization Sector STUDY GROUP 16 Question 6 Video Coding Experts Group (VCEG), 14<sup>th</sup> Meeting: Santa Barbara, CA, USA, 21-24 September, 2001
- [7] Y. Wang and Q.-F. Zhu, "Error control and concealment for video communication: A review," Proceedings of the IEEE, vol. 86, no. 5, pp. 974-997, May 1998
- [8] Y. Xu and Y. Zhou, "H.264 video communication based refined error concealment schemes," IEEE Transactions on Consumer Electronics, vol. 50, no. 4, pp. 1135-1141, November 2004
- [9] W. M. Lam, A. R. Reibman, and B. Liu, "Recovery of lost or erroneously received motion vectors," in Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process., 1993, vol. 3, pp. 417-420
- [10] Z. Wang, Y. Yu, and D. Zhang, "Best neighborhood matching: An information loss restoration technique for block-based image coding systems," IEEE Trans. Image Processing, vol. 7, pp. 1056-1061, Jul. 1998
- [11] W. M. Lam and A. R. Reibman, "An error concealment algorithm for images subject to channel errors," IEEE Trans. Image Processing, vol. 4, pp. 533-542, May 1995
- [12] Y. Wang and Q. Zhu, "Signal loss recovery in DCT-based image and video codecs," in Proc. SPIE Conf. Visual Communication and Image Processing, vol. 1605, Nov. 1991, pp. 667-678
- [13] D. Kim, S. Yang, J. Jeong, "A new temporal error concealment method for H.264 using adaptive block sizes," ICIP-IEEE, Int. Conf. Image Processing, 2005
- [14] S. Shirani, F. Kossentini, and R. Ward, "Error concealment methods, a comparative study," in Proc. Eng. Solutions For Next Millennium, 1999 IEEE Canadian Conf. Electr. Comput. Eng., Edmonton, AB, Canada, 1999, vol. 2, pp. 835-840
- [15] V. Varsa, M. M. Hannuksela, and Y. Wang, "Non-normative error concealment algorithms," ITU-T VCEG (SG16/Q6), 14<sup>th</sup> Meeting: Santa Barbara, CA, USA, 21-24 Sept., 2001
- [16] Y. Dhondt and P. Lambert, "Flexible Macroblock Ordering, an error resilience tool in H.264/AVC," Fifth FTW PhD Symposium, Faculty of Engineering, Ghent University, Desembre 2004, Paper No. 106
- [17] T. Wiegand, G. J. Sullivan, G. Bj  ntegaard and A. Luthra, "Overview of the H.264/AVC Video Coding Standard," IEEE Transactions on Circuits and Systems for Video Technology, Vol. 13, No. 7, July 2003
- [18] W. Hantanong and S. Aramvith, "Analysis of macroblock-to-slice group mapping for H.264 video transmission over packet-based wireless fading channel," Circuits and Systems, 2005. 48th Midwest Symposium, Vol. 2, pp. 1541- 1544, 7-10 Aug. 2005
- [19] S. Wenger and M. Horowitz, "FMO: Flexible Macroblock Ordering," Joint Video Team, Doc. JVT-C089, Fairfax (USA), May 2002
- [20] G.-T. Jian, M.-J. Chen, and M.-C. Chi, "Effective Error Concealment Algorithm by Boundary Information for H.264 Video Decoder," IEEE International Conference on Multimedia and Expo, July 2006, pp. 2021-2024.
- [21] Y. Chen, Y. Hu, O. C. Au, H. Li and C. W. Chen, "Video Error Concealment Using Spatio-Temporal Boundary Matching and Partial Differential Equation," IEEE Transactions on Multimedia, Vol 10, Issue: 1, pp. 2-15, Jan. 2008
- [22] C. Han and J. Liu, "New Temporal Error Concealment Method for H.264 Based on Motion Strength," International Conference on Computational Intelligence and Security Workshops, 2007. CISW 2007, pp. 858-861, 15-19 Dec. 2007





**Tanaphol Thaipanich** received his B.Eng. degree in Electrical Engineering at Chulalongkorn University in 2002 and M.S. degree in Electrical Engineering at University of Southern California in 2003. He is currently a Ph.D. student in Department of Electrical Engineering at University of Southern California. He is a member of Prof. Kuo research group. His research interests include computer vision and error detection and concealment in video data.



**Ping-Hao Wu** is a Ph.D. student in the Electrical Engineering Department, University of Southern California. He received his B.S. degree in Electrical Engineering and the M.S. degree in Communication Engineering from National Taiwan University in 2004 and 2006, respectively. He is currently a member of Prof. Kuo's research group. His research interests include digital signal processing and image/video coding.



**C.-C. Jay Kuo** received the B.S. degree from the National Taiwan University, Taipei, in 1980 and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1985 and 1987, respectively, all in Electrical Engineering. He is Director of the Signal and Image Processing Institute (SIPI) and Professor of Electrical Engineering, Computer Science and Mathematics at the University of Southern California (USC). His research interests are in the areas of digital image/video analysis and modeling, multimedia data compression, communication and networking and multimedia database management. Dr. Kuo has guided about 87 students to their Ph.D. degrees and supervised 20 postdoctoral research fellows. He is a co-author of about 140 journal papers, 740 conference papers and 9 books. Dr. Kuo is a Fellow of IEEE and SPIE. He is Editor-in-Chief for the Journal of Visual Communication and Image Representation, and Editor for the Journal of Information Science and Engineering, LNCS Transactions on Data Hiding and Multimedia Security and the EURASIP Journal of Applied Signal Processing. Dr. Kuo received the National Science Foundation Young Investigator Award (NYI) and Presidential Faculty Fellow (PFF) Award in 1992 and 1993, respectively.