LETTER

IEICE Electronics Express, Vol.10, No.10, 1–6

Optimal binary encoding scheme for the fast motion estimation based on Hamming distances

Sang-Uk Park[1], Jae-Young Sim[2a)], Chang-Su Kim[3], and Sang-Uk Lee[1]

[1] *School of Electrical Engineering, ASRI and INMC, Seoul National University, Seoul 151–742, Korea*

[2] *School of Electrical and Computer Engineering, UNIST, Ulsan 689–798, Korea*

[3] *School of Electrical Engineering, Korea University, Seoul, 136–701, Korea*

a) *jysim@unist.ac.kr*

Abstract: Optimal binary codes for pixel values are designed for accurate block matching in the fast motion estimation. The conventional binary block matching techniques reduce the computational complexity and the memory bandwidth for the motion estimation, but they degrade the matching accuracy. We find the optimal mapping function between the set of decimal numbers for uniformly quantized pixel values and the set of binary codes, so that the weighted sum of mean squared errors between the absolute differences and the Hamming distances is minimized. Experimental results show that the proposed three-bit binary code set yields about 0.4 dB gain over the conventional techniques.

Keywords: fast motion estimation, video encoding, binary block matching, optimal binary codes

Classification: Electron devices, circuits, and systems

References

[1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, July 2003.

[2] A. Erturk and S. Erturk, "Two-bit transform for binary block motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 7, pp. 938–946, July 2005.

[3] C. Choi and J. Jeong, "Enhanced two-bit transform based motion estimation via extension of matching criterion," *IEEE Trans. Consum. Electron.*, vol. 56, no. 3, pp. 1883–1889, Aug. 2010.

[4] R. W. Hamming, "Error detecting and error correcting codes," *Bell System Technical Journal*, vol. 29, no. 2, pp. 147–160, April 1950.

[5] J. Cong, B. Liu, S. Neuendorffer, J. Noguera, K. Vissers, and Z. Zhang, "High-level synthesis for FPGAs: from prototyping to deployment," *IEEE Trans. Computer-Aided Design Integr. Circuits Syst.*, vol. 30, no. 4, pp. 473–491, April 2011.

DOI: 10.1587/elex.10.20130160
Received March 05, 2013
Accepted April 22, 2013
Published May 25, 2013

IEICE Electronics Express, Vol.10, No.10, 1–6

1 Introduction

There has been an increased demand for real-time video encoding with constrained hardware resources to facilitate various multimedia applications. However, conventional video coding techniques, such as H.264 [1], require a huge amount of computation and memory bandwidth, especially to perform the motion estimation. Therefore, fast motion estimation algorithms have been proposed in [2, 3]. To reduce the memory space, these algorithms represent a pixel value using a two-bit binary code. Then, they use the dissimilarity measures based on bitwise operations, instead of the sum of absolute differences (SAD), to alleviate the computational complexity of the block matching. These algorithms, however, degrade the block matching accuracy due to the limited precision of pixel values. In this paper, we introduce the optimal mapping scheme of pixel values into binary codes and improve the performance of the binary block matching for the fast motion estimation.

2 Binary block matching techniques

In [2, 3], a pixel value is quantized with a non-uniform two-bit quantizer, which is adaptively designed for each block in an image. Then, the Gray codes are used to encode the quantized pixel values and to measure the distances to matching blocks for the motion estimation. Let $X_k(\mathrm{p})$ and $Y_k(\mathrm{p})$ denote the k-th bit values of the two-bit representation for pixel p in a current frame and the previous frame, respectively. The number of non-matching points (NNMP) measure [2] computes

$$\mathrm{NNMP(v)} = \sum_{\mathrm{p}\in\Omega} \{(X_1(\mathrm{p}) \oplus Y_1(\mathrm{p{+}v})) \parallel (X_2(\mathrm{p}) \oplus Y_2(\mathrm{p{+}v}))\} \tag{1}$$

where $\parallel$ and $\oplus$ denote the boolean OR and exclusive-OR operations, respectively. Ω is a block, composed of pixels p's, and v is a motion vector candidate of Ω. The NNMP measure simply checks whether the two quantized pixel values are identical or not, and thus it yields relatively low matching accuracy. The extended NNMP (ENNMP) measure [3] considers the difference at each bit position, which is given by

$$\mathrm{ENNMP(v)} = \sum_{\mathrm{p}\in\Omega} \{(X_1(\mathrm{p}) \oplus Y_1(\mathrm{p{+}v})) + (X_2(\mathrm{p}) \oplus Y_2(\mathrm{p{+}v}))\}\,. \tag{2}$$

Note that ENNMP measure is the same as the Hamming distance (HD) [4] between the two binary codes. According to [5], the computation speeds of NNMP and ENNMP are about 2.65 and 2.59 times faster than that of the SAD measure using the same number of bits.

3 Proposed algorithm

The conventional techniques [2, 3] achieve fast motion estimation, but they degrade the matching accuracy as compared with the SAD measure. It is because the binary codes in [2, 3] do not fully reflect the relation between the

DOI: 10.1587/elex.10.20130160
Received March 05, 2013
Accepted April 22, 2013
Published May 25, 2013

IEICE Electronics Express, Vol.10, No.10, 1–6

HD and its corresponding absolute difference (AD). In this work, we investigate the optimal mapping between an absolute pixel value and its binary code, which minimizes the error between the HD and the AD. We consider the general case that each pixel is quantized with an arbitrary number of bits, whereas [2, 3] considers the two-bit quantization only. Also, we employ the uniform quantization, instead of the adaptive quantization [2, 3], to represent the same absolute value with the same binary code and to remove the additional complexity to find an adaptive non-uniform quantizer for each block.

Let A^n denote the set of decimal numbers uniformly quantized with n bits, and B^n denote the set of n-bit binary sequences, *e.g.* $A^2 = \{0, 1, 2, 3\}$ and $B^2 = \{00, 01, 10, 11\}$. There is a one-to-one mapping f^n from decimal numbers in A^n to binary codes in B^n

$$f_n : A^n \rightarrow B^n. \tag{3}$$

A natural encoding scheme assigns the binary code $i \in B^n$ to each decimal number $x \in A^n$, which satisfies the relation

$$x = \sum_{k=1}^{n} \left\{ 2^{k-1} \times i_k \right\} \tag{4}$$

where i_k represents the k-th bit value of i. An alternative approach is to use the Gray code [2, 3], in which only one bit position is changed between two circularly adjacent codes.

These encoding schemes, however, do not guarantee the optimal binary block matching. The traditional motion estimation measures the absolute difference $d_{\mathrm{A}}(x, y)$ between two pixel values x and y in A^n, given by

$$d_{\mathrm{A}}(x, y) = |x - y|, \quad x, y \in A^n. \tag{5}$$

On the other hand, the binary block matching technique [3] measures the Hamming distance $d_{\mathrm{H}}(i, j)$ between two binary codes i and j in B^n,

$$d_{\mathrm{H}}(i, j) = \sum_{k=1}^{n} \{ i_k \oplus j_k \}, \quad i, j \in B^n. \tag{6}$$

Note that the possible range of d_{A} in (5) is $[0, 2^n - 1]$, whereas that of d_{H} in (6) is $[0, n]$. Therefore, the mapping from the set of d_{A}'s to the set of d_{H}'s is not one-to-one. Moreover, the HD counts the number of different bit positions between two binary codes. Therefore, it does not reflect the AD between the two pixel values accurately. In other words, the HD is not proportional to the AD.

To alleviate such mismatch between the HD and the AD, we find an optimal mapping f_n^* that minimizes the error E between HD's and AD's.

$$f_n{}^* = \arg\min_{f_n} E(f_n), \tag{7}$$

DOI: 10.1587/elex.10.20130160
Received March 05, 2013
Accepted April 22, 2013
Published May 25, 2013

IEICE Electronics Express, Vol.10, No.10, 1–6

Table I. Weighting parameters for the cases of $n = 2$, 3, and 4.

	ω_1	ω_2	ω_3	ω_4	ω_5	ω_6	ω_7	ω_8	$\omega_{9\sim12}$	$\omega_{13\sim15}$
$n=2$	0.78	0.19	0.03							
$n=3$	0.59	0.21	0.11	0.05	0.02	0.01	0.01			
$n=4$	0.44	0.18	0.11	0.08	0.06	0.04	0.03	0.02	0.01	0.00

Table II. The proposed binary codes compared to the natural and the Gray codes, when $n = 2$, 3, and 4.

	$n=2$		$n=3$			$n=4$		
	Natural	Gray/ Prop.	Natural	Gray	Prop.	Natural	Gray	Prop.
0	00	00	000	000	000	0000	0000	0000
1	01	01	001	001	001	0001	0001	0001
2	10	11	010	011	011	0010	0011	0011
3	11	10	011	111	010	0011	0010	0111
4			100	101	110	0100	0110	1111
5			101	100	100	0101	0111	1110
6			110	110	101	0110	0101	1100
7			111	010	111	0111	0100	1000
8						1000	1100	1001
9						1001	1101	1011
10						1010	1111	1010
11						1011	1110	0010
12						1100	1010	0110
13						1101	1011	0100
14						1110	1001	0101
15						1111	1000	1101
WMSE	0.62	0.06	1.10	0.41	0.40	1.87	1.19	0.67

where $E(f_n)$ is the weighted sum of mean squared errors (WMSE) for all $d_{\rm H}$'s mapped to a given $d_{\rm A}$. Specifically,

$$E(f_n) = \sum_{m=1}^{2^n-1} \left\{ \frac{\omega_m}{|S_m|} \sqrt{\sum_{(x,y)\in S_m} \{d_{\rm A}(x,y) - d_{\rm H}(f_n(x), f_n(y))\}^2} \right\} \tag{8}$$

where $S_m = \{(x,y)|\, d_{\rm A}(x,y) = m,\ x,y \in A_n\}$ and $|S_m|$ denotes the cardinality of S_m. Also, ω_m is a weighting parameter for each value of $d_{\rm A}(x,y) = m$. In practice, we compute the probability distribution of absolute differences, which occur in the motion estimation of six training video sequences in the common intermediate format (CIF): "Mobile," "Bus," "Paris," "Stefan," "Container," and "News." Then, we take the probability of $d_{\rm A}(x,y) = m$ as ω_m. Table I shows the trained weighting parameters for the three cases of the numbers of bits $n = 2$, 3, and 4. We see that ω_m becomes smaller as m increases in general, since the blocks within a search range in the previous frame tend to yield similar characteristics to the current block.

Table II lists the proposed binary codes, obtained by the optimal mapping $f_n{}^*$ in (7), when $n = 2$, 3, and 4, respectively, in comparison with the

DOI: 10.1587/elex.10.20130160
Received March 05, 2013
Accepted April 22, 2013
Published May 25, 2013

Table III. Comparison of the motion estimation performances in terms of the average PSNR's (dB).

	SAD [1]	NNMP [2]	ENNMP [3]	Proposed algorithm		
				$n=2$	$n=3$	$n=4$
Football	25.39	23.83	24.16	24.13	24.74	24.55
Flower	26.03	25.81	25.86	25.81	26.02	25.85
Soccer	29.54	28.23	28.60	27.83	29.00	28.81
Coastguard	29.61	29.14	29.33	28.04	29.47	29.37
Foreman	33.42	31.62	31.80	31.89	32.69	32.82
Hall Monitor	34.35	33.30	33.47	33.10	33.92	33.72
Mother	36.73	35.76	35.89	34.93	36.32	36.47
Akiyo	42.81	42.23	42.30	41.99	42.58	42.61
Average	32.24	31.24	31.43	30.97	31.84	31.78

natural binary codes and the Gray codes. Note that there can be several sets of optimal codes, but we only show the first code set according to the lexicographical order. It is observed that the two-bit optimal codes are equivalent to the Gray codes. In the cases of $n = 3$ and $n = 4$, the proposed codes yield the similar property to the Gray codes: two successive binary codes differ in only one bit position with the exception of the pair of the last code and the first code. Notice that the computed WMSE in (8) of the proposed codes are equal to or smaller than those of the natural and the Gray codes.

4 Experimental results

We evaluate the motion estimation performance of the proposed algorithm using the first 100 frames of eight test video sequences in the CIF: "Football," "Flower," "Soccer," "Coastguard," "Foreman," "Hall Monitor," "Mother," and "Akiyo." The block size for the motion estimation is 16×16 and the search range is ± 16.

Table III shows the peak signal-to-noise ratio (PSNR) results of the proposed algorithm, as compared with the conventional fast motion estimation algorithms [2, 3]. The proposed algorithm with the three-bit binary codes achieves 0.6 dB and 0.41 dB gains over the performances of [2] and [3], respectively. The proposed algorithm employs the uniform quantization and therefore provides worse performance than [2, 3] in the case of $n = 2$. However, since the proposed two-bit binary codes are the Gray codes as shown in Table II, it achieves the same performance as [3] if it is combined with the non-uniform quantization. Note that even though the quantization error of the four-bit codes is smaller than that of the three-bit codes in general, the proposed algorithm yields a better motion estimation performance with $n = 3$ than $n = 4$. It means that the proposed four-bit binary codes results in a larger mismatch between the HD and the AD than that of the three-bit binary codes. We also compare the performance of the traditional motion estimation algorithm using the SAD measure for 8-bit pixel values [1], and we see that the proposed algorithm degrades the benchmark performance by

DOI: 10.1587/elex.10.20130160
Received March 05, 2013
Accepted April 22, 2013
Published May 25, 2013

IEICE Electronics Express, Vol.10, No.10, 1–6

about 0.4 dB only.

5 Conclusions

We proposed the optimal encoding scheme of pixel values for the fast binary motion estimation. The optimal codes minimize the difference between the AD's of pixel values and the HD's of the corresponding binary codes. Experimental results demonstrated that the proposed encoding scheme yields the best motion estimation accuracy when it uses three bits for each pixel. Also, the proposed algorithm increases the accuracy of the binary block matching, as compared with the conventional techniques [2, 3].

Acknowledgments

This work was supported in part by the National Research Foundation of Korea (NRF) grant funded by the Ministry of Education, Science and Technology (MEST) (No. 2012-0005410), and in part by the Basic Science Research Program through the NRF of Korea funded by the MEST (No. 2012-0003908).

DOI: 10.1587/elex.10.20130160
Received March 05, 2013
Accepted April 22, 2013
Published May 25, 2013