A Hierarchical Decimation Lattice Based on N-Queen With an Application for Motion Estimation

Chung-Neng Wang, Shin-Wei Yang, Chi-Min Liu, and Tihao Chiang, *Senior Member, IEEE*

***Abstract*—We present a novel technique, N-queen lattice, to spatially subsample a block of pixels. Although this lattice is pertinent to many applications, we present an application to speed up motion estimation with minimal loss of coding efficiency. The N-queen lattice is constructed to characterize spatial features in all directions. It can be hierarchically organized for motion estimation with variable nonsquare block size. Despite the randomized lattice structure, we demonstrate that it is possible to achieve compact data storage architecture for efficient memory access and simple hardware implementation. Our simulations show that the N-queen lattice is superior to several existing sampling techniques with improvement in speed by about N times and small loss in peak SNR.**

***Index Terms*—Fast motion search, N-queen lattice, pixel decimation.**

I. Introduction

SEVERAL VIDEO coding standards including MPEG-1/2/4 contain block motion estimation as the most computationally intensive task. There are three categories to improve motion estimation by reducing the number of search points [1], [2], the load for measuring the distortion [3], [4], and the number of matching pixels from a block [5]–[10]. The MPEG-4 reference software has provided two fast algorithms that have significantly reduced the number of search points [1], [2]. The bit truncation or one-bit algorithms reduce the complexity by modifying the bit depth and the distortion measure [3], [4]. When the pixels are represented in a binary format, the block matching can use exclusive-OR Boolean operators and table lookup techniques. The pixel decimation approaches can be easily combined with approaches from the first two categories. Thus, we will focus on pixel decimation to achieve further improvement.

The pixel decimation can be achieved with either fixed [5]–[8] or adaptive patterns [9], [10]. As shown in Fig. 1(b), Bierling used an orthogonal sampling lattice with a 4 : 1 subsampling [5], which is referred to as the “quarter pattern” here. Liu and Zaccarin implemented pixel decimation that is similar to Bierling’s approach with four alternating subsampling patterns selected for each step so that all the pixels in the current block are visited [6]. The pixel decimation can be adapted based on the spatial luminance variation within a picture [9], [10]. Adaptive techniques can achieve better coding efficiency as compared to the uniform subsampling schemes [5]–[8] with an overhead in deciding which pattern is more representative. Due to mispredicted branches, the irregular or adaptive structure is difficult for pipelined implementation.

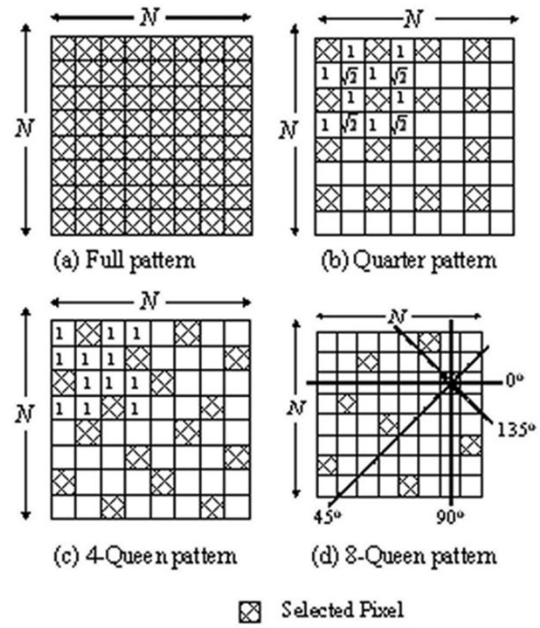

Fig. 1. Pixel patterns for decimation. (a) Full pattern with $N \times N$ pixels selected. (b) Quarter pattern uses 4 : 1 subsampling. (c) Four-queen pattern is tiled with four identical patterns. (d) Eight-queen pattern. (c) and (d) are derived from the N-queen approach with $N = 4$ and $N = 8$, respectively.

The quarter pattern has advantages in pipelining and memory access but fails to represent half of the lines in horizontal, vertical, and diagonal directions. To represent key features and maintain pipelined memory access, we will construct a family of lattices that maintain the regularity and characterize more directional features.

II. N-Queen Pixel Decimation

Pixel decimation is used to reduce the computation for measuring the distortion for each block during the search [5]. The most representative sampling lattice is selected based on how much the texture and edge information are retained with minimal number of pixels. The sampling lattice is analyzed with the spatial homogeneity and directional coverage. The spatial

Manuscript received September 9, 2002; revised October 4, 2002. This work was supported by the National Science Council, Taiwan, R.O.C. under Contract NSC 91-2218-E 009-005.

C.-N. Wang, S.-W. Yang, and C.-M. Liu are with the Department of Computer Science and Information Engineering, National Chiao-Tung University (NCTU), Hsinchu, 30050 Taiwan, R.O.C. (e-mail: cnwang@csie.nctu.edu.tw; swyang@csie.nctu.edu.tw; cmliu@csie.nctu.edu.tw).

T. Chiang is with the Department and Institute of Electronics Engineering, National Chiao-Tung University (NCTU), Hsinchu, 30050 Taiwan, R.O.C. (e-mail: tchiang@mail.nctu.edu.tw).

Digital Object Identifier 10.1109/LSP.2003.814403

TABLE I
COMPARISON OF THE SAMPLING LATTICES FOR AN 8×8 BLOCK. IN MEASURING THE DIRECTIONAL COVERAGE, FOUR ORIENTATIONS DESCRIBED IN FIG. 1(d) ARE USED. FOR HORIZONTAL, VERTICAL, AND DIAGONAL DIRECTIONS, THERE ARE EIGHT, EIGHT, AND 15 POSSIBLE EDGES, RESPECTIVELY, WHILE FOR THE DIAGONAL DIRECTIONS, THERE ARE 15 POSSIBLE EDGES

Pattern	Spatial homogeneity			Directional coverage (θ)			
	μ_d	σ_d^2	σ_d / μ_d	0^o	90^o	45^o	135^o
Full	0	0		$\frac{8}{8}$	$\frac{8}{8}$	$\frac{15}{15}$	$\frac{15}{15}$
Quarter[5]	1.14	0.04	17.16%	$\frac{4}{8}$	$\frac{4}{8}$	$\frac{7}{15}$	$\frac{7}{15}$
Hexagonal[7]	1.03	0.11	11.07%	$\frac{4}{8}$	$\frac{8}{8}$	$\frac{12}{15}$	$\frac{12}{15}$
4-Queen	1	0		$\frac{8}{8}$	$\frac{8}{8}$	$\frac{10}{15}$	$\frac{10}{15}$
8-Queen	1.32	0.14	28.77%	$\frac{8}{8}$	$\frac{8}{8}$	$\frac{8}{15}$	$\frac{8}{15}$

homogeneity is measured by the average and variance of spatial distances from each skipped pixel to its nearest selected pixel

$$\mu_d = \frac{1}{(N^2 - K)} \sum_{x=1,y=1}^{N} \|(x,y) - S(x,y)\| \quad (1)$$

$$\sigma_d^2 = \frac{1}{(N^2 - K)} \sum_{x=1,y=1}^{N} (\|(x,y) - S(x,y)\| - \mu_d)^2 \quad (2)$$

where N is the dimension of the block, and $S(x,y)$ indicates the coordinates of the selected pixel nearest to the pixel at the position (x,y). K is the number of the selected pixels. Smaller μ_d and σ_d^2 indicate a more spatially homogeneous sampling lattice. An edge is defined as a line passing through the sampling grids in any of 0°, 45°, 90°, and 135° directions as shown in Fig. 1(d). The directional coverage is measured as the percentage of edges that at least one of the selected pixels exists on an edge. Table I shows that the quarter pattern has less spatial homogeneity and lacks half of the coverage in the specified directions. To address the issues of spatial homogeneity and directional coverage, we construct a new N-queen sampling lattice.

To fully represent the spatial information of a $N \times N$ block, it is required that at least one pixel should be selected for each row, column, and diagonal. To satisfy such a constraint, the solution is identical to the problem of placing N queens on a chessboard, which is referred to as N-queen pattern. For a $N \times N$ block, as shown in Fig. 1(c) and (d), every pixel of the N-queen pattern occupies a dominant position, which is located at the center. All the other pixels located on the four lines in the vertical, horizontal and diagonal directions are removed from the list of the selected pixels. With such elimination process, there is exactly one pixel selected for each row, column, and (not necessarily main) diagonal of the block. Thus, the N-queen patterns present a $N:1$ subsampling lattice that can provide N times of speedup improvement.

The N-queen patterns are not unique. For example, there are 92 8-queen patterns for a 8×8 block. The remaining issue is to identify which one provides a better representation. By (1), the average distances of these 92 patterns are distributed between 1.29 and 1.37 pixels. Thus, the variation in average distances is only 0.08 pixels. We find that the 92 8-queen patterns have almost identical performance with varying peak SNR (PSNR) less than 0.1 dB.

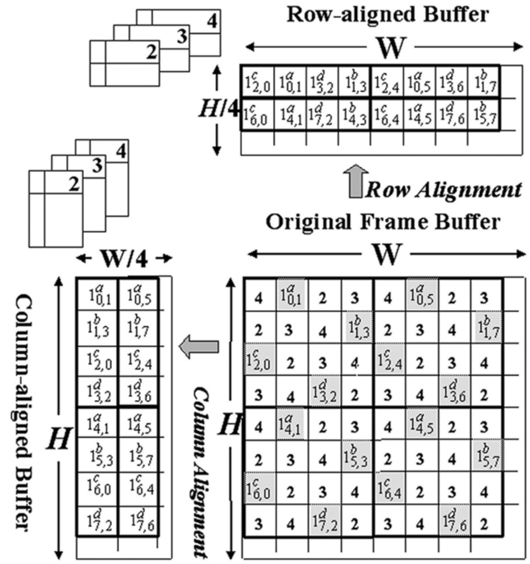

Fig. 2. Row and column alignment approaches for transforming a two-dimensional 4×4 block into a one-dimensional vector of four pixels.

To minimize the memory access bandwidth, a group number (one to four) is used to index each group of pixels that are placed in a separate memory buffer as shown in Fig. 2. There is a separate frame buffer allocated for each of the N groups based on the N-queen lattice. For example, the 4-queen lattice stores the nonoverlapping references pixels in four smaller buffers. One of the special properties of this storage technique is that a macroblock resides in a continuous memory space for easy access. For example, the selected pixels are grouped together to fully exploit the single-instruction multiple-data architecture as proposed by Moschettie *et al.* [8]. If we use a pipelined memory access strategy, a shift of one pixel in each frame buffer represents a spatial shift of $\log(N)$ pixels in the original frame. Thus, this data storage architecture can easily facilitate a $\log(N)$ search strategy. Another interesting observation is that each pixel is sequentially accessible even though the search strategy is hierarchical. This provides an elegant solution to improve both search strategy and memory access.

To compute the full-pixel motion vectors for blocks with sizes 16×16 and 8×8, the motion vectors of 16×16 blocks are computed first. The 8×8 block uses the motion vector of a 16×16 block as an initial position and fine-tunes the search in a window of ± 2. The coding mode is decided based on a tradeoff between the distortion and the required bits for encoding motion vectors. The half-pixel motion vector is found by searching the eight points surrounding the best full-pixel motion vector using the 16×16 and 8×8 modes, respectively. In the fine-tuning process, we classify the eight pixels into two sets. The first set includes the search points on the diagonals, and the other set covers the remainders. For each set, we perform the motion es-

TABLE II
PERFORMANCE OF THE FOUR PIXEL PATTERNS, THE TWO SEARCH STRATEGIES, AND THE VARIOUS VIDEO SEQUENCES ON DIFFERENT TESTING CONDITIONS. FOR EACH METHOD, THE FIRST SYMBOL DENOTES THE SEARCH STRATEGY, AND THE REMAINING SYMBOL DENOTES THE SAMPLING PATTERNS. FOR EXAMPLE, THE NOTATION "PMVFAST_8" REPRESENTS THE MOTION ESTIMATION USING PMVFAST AND EIGHT-QUEEN PATTERN. WHERE THE COLUMN "PSNRY" DENOTES THE AVERAGE PSNR FOR THE LUMINANCE COMPONENT, AND THE COLUMN "CHECKING POINTS" INDICATES THE ACTUAL NUMBER FOR CALCULATING SAD CRITERION OF 16 × 16 PIXELS

Sequence	Format	Bit Rate (bits/sec)	Frame Rate	Search Range	Methods	PSNRY	Δ PSNRY	Checking Points	Ratio over Full Search
Container	QCIF	10k	7.5 (Hz)	16	Full Search	30.11		7802142	1
					Full_Q	29.86	-0.25	1972498	3.96
					Full_H	29.84	-0.27	1972519	3.96
					Full_4	29.91	-0.20	1972505	3.96
					Full_8	29.22	-0.89	1000893	7.80
					PMVFAST_F	29.91	-0.20	141141	55.28
					PMVFAST_Q	29.93	-0.18	57077	136.70
					PMVFAST_H	29.95	-0.16	56926	137.06
					PMVFAST_4	29.97	-0.14	56979	136.93
					PMVFAST_8	29.75	-0.36	43103	181.01
Foreman	CIF	112k	10 (Hz)	16	Full Search	29.49		41722116	1
					Full_Q	29.05	-0.44	10541341	3.96
					Full_H	29.13	-0.36	10541915	3.96
					Full_4	29.19	-0.30	10542251	3.96
					Full_8	28.74	-0.75	5344736	7.81
					PMVFAST_F	29.33	-0.16	965652	43.21
					PMVFAST_Q	29.03	-0.46	343340	121.52
					PMVFAST_H	29.14	-0.35	345490	120.76
					PMVFAST_4	29.21	-0.28	345961	120.60
					PMVFAST_8	28.96	-0.53	244162	170.88
Foreman	CIF	1M	30 (Hz)	16	Full Search	36.15		126095972	1
					Full_Q	35.91	-0.24	31878114	3.96
					Full_H	35.96	-0.19	31878149	3.96
					Full_4	36.03	-0.12	31878296	3.96
					Full_8	35.75	-0.40	16175154	7.80
					PMVFAST_F	36.24	0.09	2476415	50.92
					PMVFAST_Q	36.07	-0.08	973367	129.55
					PMVFAST_H	36.08	-0.07	973062	129.59
					PMVFAST_4	36.15	0.00	973667	129.51
					PMVFAST_8	35.98	-0.17	723427	174.30
Stefan	CCIR 601 (Field image)	4M	60 (Hz)	32	Full Search	30.83		1672669277	1
					Full_Q	30.50	-0.33	419362397	3.99
					Full_H	30.52	-0.31	419359065	3.99
					Full_4	30.75	-0.08	419375634	3.99
					Full_8	30.53	-0.30	210492029	7.95
					PMVFAST_F	30.81	-0.02	9327506	179.33
					PMVFAST_Q	30.55	-0.28	3514014	476.00
					PMVFAST_H	30.57	-0.26	3511029	476.40
					PMVFAST_4	30.69	-0.14	3506592	477.01
					PMVFAST_8	30.51	-0.32	2564783	652.17

timation with various sampling patterns. We then find the best half-pixel motion vectors by comparing the distortion of the two candidates with the minimum distortion of the best full-pixel motion vector.

III. EXPERIMENTAL RESULTS

In our simulation, we use the MPEG-4 reference software, and the distortion measure is sum of absolute difference (SAD), which is computed for a macroblock of size 16×16 and various search ranges. The coding efficiency is analyzed based on the three factors: sampling patterns, search strategies, and testing conditions.

As for the sampling patterns, we use five patterns as described in Fig. 1. The full pattern ("F") selects all of the pixels in the current block. The quarter pattern ("Q") and the hexagonal pattern ("H") are described in [5] and [7], respectively. The 4-queen ("4") pattern is constructed by tiling multiple small 4-queen patterns for each macroblock. The 8-queen ("8") pattern is constructed by tiling similarly to the 4-queen pattern.

As for the search strategies, we tested full search and the fastest approach, predictive motion vector field adaptive search technique (PMVFAST) [2], as recommended by the MPEG-4 committee. We follow the recommended testing conditions as prescribed by the MPEG committee [1]. As shown in Table II and Fig. 3, we reach the following conclusions.

1) The N-queen patterns have negligible video quality degradation. With PMVFAST, the loss in PSNR is less than 0.36 dB for slow motion video such as "Container."

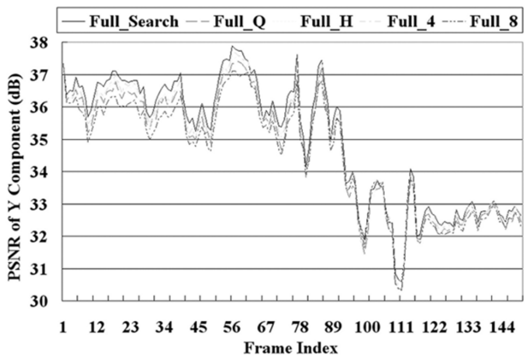

Fig. 3. PSNR comparisons for the motion estimation based on exhaustive search strategy and various subsampling lattices. The Foreman sequence is in common intermediate format and is encoded with 512 kb/s and 15 frames/s. The search range for motion estimation is 16 for encoding.

The loss in PSNR is less than 0.53 dB at worst for fast motion video. The 4-queen pattern is better than the quarter and hexagonal patterns by about 0.05 $\sim$ 0.25 dB for the frame-coded sequences.

2) When we compare algorithms with the same pattern but different search strategies, the degradation using full search is more than that of PMVFAST for all patterns. This may be caused by the predictive nature of PMVFAST.
3) It is advantageous to use the N-queen patterns for the CCIR-601 interlaced sequences because N-queen patterns retain all the horizontal spatial information within a block, while the quarter and hexagonal patterns lose half of the horizontal spatial information.

IV. Conclusion

This letter has presented a novel and simple pixel decimation technique using the N-queen lattice with an application for block-based motion estimation. The complexity and memory bandwidth can be arbitrarily reduced by a factor of N. It is superior in terms of spatial homogeneity and directional coverage. The hierarchical N-queen sampling lattice is flexible when the block size is variable including nonsquare block used in H.26L.

Acknowledgment

The authors wish to thank the anonymous reviewers for their insightful comments to improve the initial draft of this letter.

References

[1] A. M. Tourapis, O. C. Au, and M. L. Liou, "Fast block-matching motion estimation using predictive motion vector field adaptive search technique (PMVFAST)," ISO/IEC, Noordwijkerhout, NL, ISO/IEC JTC1/SC29/WG11 MPEG2000/M5866, 2000.

[2] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Processing*, vol. 9, pp. 287–290, Feb. 2000.

[3] B. Natarajan, V. Bhaskaran, and K. Konstantinides, "Low-complexity block-based motion estimation via one-bit transforms," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, pp. 702–706, Aug. 1997.

[4] X. Song, T. Chiang, X. Lee, and Y.-Q. Zhang, "New fast binary pyramid motion estimation for MPEG2 and HDTV encoding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 1015–1028, Oct. 2000.

[5] M. Bierling, "Displacement estimation by hierarchical block matching," *Proc. SPIE*, vol. 1001, pp. 942–951, 1988.

[6] B. Liu and A. Zaccarin, "New fast algorithms for the estimation of block motion vector," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 3, pp. 148–157, Apr. 1993.

[7] K. T. Choi, S. C. Chan, and T. S. Ng, "A new fast motion estimation algorithm using hexagonal subsampling pattern and multiple candidate search," in *Proc. ICIP*, 1996, pp. 497–500.

[8] F. Moschettie and E. Debes, "About macroblock subsampling for motion estimation," *Proc. ICME*, Aug. 2001.

[9] Y.-L. Chan and W.-C. Siu, "New adaptive pixel decimation for block motion vector estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 113–118, Feb. 1996.

[10] Y. K. Wang, Y. Q. Wang, and H. Kuroda, "A globally adaptive pixel-decimation algorithm for block-motion estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, pp. 1006–1011, Sept. 2000.