American J. of Engineering and Applied Sciences 4 (4): 448-460, 2011
ISSN 1941-7020
© 2014 C.Y.F. Ho *et al*., This open access article is distributed under a Creative Commons Attribution (CC-BY) 3.0 license

Single Step Optimal Block Matched Motion Estimation with Motion Vectors Having Arbitrary Pixel Precisions

[1]Charlotte Yuk-Fan Ho, [2]Bingo Wing-Kuen Ling,
[3]Saverio Giovanni Blasi, [4]Zhi-Wei Chi and [5]Wan-Chi Siu
[1]School of Mathematical Sciences, Queen Mary, University of London,
Mile End Road, E1 4NS, United Kingdom, London
[2]School of Engineering, University of Lincoln, Lincoln,
Lincolnshire, LN6 7TS, United Kingdom
[3]Department of Electronic Engineering, Queen Mary, University of London,
Mile End Road, London, E1 4NS, United Kingdom
[4]Department of Electronic Engineering, Division of Engineering,
King's College London, Strand, London, WC2R 2LS, United Kingdom
[5]Department of Electronic and Information Engineering,
Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong, China

Abstract: Problem statement: This study derives the optimal motion vector with arbitrary pixel precisions in a single step. **Approach:** A non-linear block matched motion model was proposed. Based on the proposed non-linear block matched motion model, the optimal motion vector which minimizes the mean square error was solved analytically in a single step via a gradient approach. **Results:** The mean square error based on the proposed method was guaranteed to be lower than or equal to that based on conventional methods. The computational efforts for the proposed method were lower than that of conventional methods particularly when the required pixel precision is higher than or equal to the quarter pixel precisions. **Conclusion:** As integer pixel locations, half pixel locations and quarter pixel locations are particular locations represented by the proposed model, the mean square error based on the proposed method is guaranteed to be lower than or equal to that based on these conventional methods. Also, as the proposed method does not require searching from coarse pixel locations to fine pixel locations, the computational efforts for the proposed method are lower than that of the conventional methods.

Key words: Motion estimations, tracking applications, respiratory motion, facial motion, block matched, macro blocks, reference frame, current frame, motion vector

INTRODUCTION

Motion estimations play an important role in motion tracking applications, such as in a respiratory motion tracking application (Chun and Fessler, 2009) and in a facial motion tracking application (Lin *et al*., 2002). The most common motion estimation algorithm is the block matched motion estimation algorithm (Saha *et al*., 2008). The current frame is usually partitioned into numbers of macro blocks with fixed or variable sizes. Each macro block in the current frame is compared with a number of macro blocks in the reference frame translated within a search window. Block matching errors are calculated based on a predefined cost function. The macro block in the reference frame that gives the minimum block matching error is considered as the best approximation of the macro block in the current frame. Each macro block in the current frame is represented by the best macro block in the reference frame, the motion vector (the motion vector is the vector representing the translation of the macro block in the reference frame) and the residue (the residue is the difference between the macro block in the current frame and the best translated macro block in the reference frame).

The most common block matched motion estimation algorithm is the full integer pixel search algorithm. The full integer pixel search algorithm is a

Corresponding Author: Bingo Wing-Kuen Ling, School of Engineering, University of Lincoln, Lincoln, Lincolnshire, LN6 7TS, United Kingdom Tel: +44 (0)15 2283 7927

centre based algorithm in which all integer pixel locations in the search window are examined. However, the motion vectors are not necessarily represented by integer pixel precisions and a large portion of macro blocks in the current frame are best approximated by the macro blocks in the reference frame translated within a plus or a minusone pixel range around integer pixel locations. Hence, block matching errors could be further reduced if motion vectors are represented by non-integer pixel precisions. Conventional non-integerpixel search algorithms start searching pixels at half pixel locations. Half pixels are interpolated by nearby pixels at integer pixel locations. Block matching errors at some or all half pixel locations are evaluated. The half pixel location with the minimum block matching error is chosen. Similarly, quarter pixels are interpolated by nearby pixels at half pixel and integer pixel locations. The quarter pixel location with the minimum block matching error is chosen. Finer pixel locations could be evaluated successively. Since the block matching errors at finer pixel locations are evaluated via interpolations from the coarser pixel locations, if motion vectors with very fine pixel precisions are required, then many pixel locations are required to be evaluated. Hence, computational complexities of these algorithms are very high and these algorithms are very inefficient. Also, existing pixel search algorithms could only achieve motion vectors with rational pixel precisions. If the true motion vector is with an irrational pixel precision, then an infinite number of pixel locations have to be evaluated.

Interpolations are implemented via some predefined functions, such as a real valued quadratic function with two variables (Li and Gonzales, 1996), a parabolic function (Du *et al*., 2003) and a straight line (Lee *et al*., 2003). As the block matching error is a highly non-linear and non-convex function of the motion vector, it is very difficult to solve the motion vector that globally minimizes the block matching error. Hence, many pixel locations are still required to be evaluated and the pixel location with the lowest block matching error is chosen. Similar to conventional quarter pixel search algorithms, computational complexities of these algorithms are still very high and these algorithms are still very inefficient. Also, if the true motion vector is with an irrational pixel precision, then an infinite number of pixel locations still have to be evaluated.

In this study, we propose a non-linear block matched motion model and solve the motion vectors with arbitrary pixel precisions in a single step. Our proposed algorithm has the following salient features. (1) The block matching error is evaluated in a single step which globally minimizes the mean square error. As the calculation of the mean square error at a finer pixel location is not derived from the coarser pixel locations, the computational complexity of our proposed algorithm is much lower than that of conventional quarter pixel search algorithms. (2) Our proposed algorithm could achieve the true motion vector even though the true motion vector is with an irrational pixel precision. Numerical computer simulation results show that the mean square errors of various video sequences based on our proposed algorithm are lower than that based on conventional half pixel search algorithms and quarter pixel search algorithms.

Proposed non-linear block matched motion model: Denote the size of a macro block as $N\times N$, where $N\in Z^+$. $\forall k\in Z^+$, let B_{k+1} be a subset of pixels in the k+1th current frame and $B_{k+1}(x, y)$ be the pixel value of B_{k+1} at the pixel location (x, y). Similarly, $\forall k\in Z^+$, let B_k be a subset of pixels in the k^{th} reference frame and $B_k(x, y)$ be the pixel value of B_k at the pixel location (x, y). $\forall k\in Z^+$, denote the motion vector of B_k as $(p_{0,k}+p_k,q_{0,k}+q_k)$, where $(p_{0,k}, q_{0k})\in Z^2$ and $(p_k, q_k)\in S\equiv[0,1]\times[0,1]\backslash\{(0,1),(1,0),(1,1)\}$. $\forall k\in Z^+$, $(p_{0,k}, q_{0k})$ is the best integer pixel location which minimizes the block matching error and can be obtained via existing full integer pixel search algorithms. On the other hand, $\forall k\in Z^+$, (p_k,q_k) is the fine shift within S around $(p_{0,k},q_{0k})$ and the values of p_k and q_k could be either rational or irrational. Motion vectors could be any vectors in one of the four quadrants in $\Re^2$ and the motion vectors in different quadrants are interpolated by different pixels based on different orientations.

$\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$, denote $\tilde{B}^{UL}_{k,p_k,q_k}$ as the translated B_k if the motion vector moves in the upper left direction, $\tilde{B}^{UR}_{k,p_k,q_k}$ as the translated B_k if the motion vector moves in the upper right direction, $\tilde{B}^{LL}_{k,p_k,q_k}$ as the translated B_k if the motion vector moves in the lower left direction and $\tilde{B}^{LR}_{k,p_k,q_k}$ as the translated B_k if the motion vector moves in the lower right direction. $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$, denote $\tilde{B}^{UL}_{k,p_k,q_k}(x,y)$, $\tilde{B}^{UR}_{k,p_k,q_k}(x,y)$, $\tilde{B}^{LL}_{k,p_k,q_k}(x,y)$ and $\tilde{B}^{LR}_{k,p_k,q_k}(x,y)$ be the pixel values of $\tilde{B}^{UL}_{k,p_k,q_k}$, $\tilde{B}^{UR}_{k,p_k,q_k}$, $\tilde{B}^{LL}_{k,p_k,q_k}$ and $\tilde{B}^{LR}_{k,p_k,q_k}$ at the pixel location (x,y), respectively. In this study, $\forall k\in Z^+$, $\forall(p_k,q_k)\in S$, $\forall x\in\{0,\ldots.N-1\}$ and $\forall y\in\{0,\ldots.N-1\}$, $\tilde{B}^{UL}_{k,p_k,q_k}(x,y)$, $\tilde{B}^{UR}_{k,p_k,q_k}(x,y)$, $\tilde{B}^{LL}_{k,p_k,q_k}(x,y)$ and $\tilde{B}^{LR}_{k,p_k,q_k}(x,y)$ are constructed via the following models:

$$\begin{aligned}\tilde{B}^{UR}_{k,p_k,q_k}(x,y) &\equiv (1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ &+(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\ &+q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1)\\ &+p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}+1)\end{aligned}$$

$$\begin{aligned}\tilde{B}^{LR}_{k,p_k,q_k}(x,y) &\equiv (1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ &+(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\ &+q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}-1)\\ &+p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}-1)\end{aligned}$$

$$\begin{aligned}\tilde{B}^{UL}_{k,p_k,q_k}(x,y) &\equiv (1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ &+(1-q_k)p_kB_k(x+p_{0,k}-1,y+q_{0,k})\\ &+q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1)\\ &+p_kq_kB_k(x+p_{0,k}-1,y+q_{0,k}+1)\end{aligned}$$

and:

$$\begin{aligned}\tilde{B}^{LL}_{k,p_k,q_k}(x,y) &\equiv (1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ &+(1-q_k)p_kB_k(x+p_{0,k}-1,y+q_{0,k})\\ &+q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}-1)\\ &+p_kq_kB_k(x+p_{0,k}-1,y+q_{0,k}-1)\end{aligned}$$

respectively. $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$, let the mean square error between the translated B_k and B_{k+1} be MSE_k (p_k, q_k). That is, $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$:

$$MSE_k(p_k,q_k)\equiv\min\left\{\begin{array}{l}\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left|\tilde{B}^{UL}_{k,p_k,q_k}(x,y)-B_{k+1}(x,y)\right|^2,\\ \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left|\tilde{B}^{UR}_{k,p_k,q_k}(x,y)-B_{k+1}(x,y)\right|^2,\\ \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left|\tilde{B}^{LL}_{k,p_k,q_k}(x,y)-B_{k+1}(x,y)\right|^2,\\ \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left|\tilde{B}^{LR}_{k,p_k,q_k}(x,y)-B_{k+1}(x,y)\right|^2\end{array}\right\}$$

It is worth noting that:

$$(1-p_k)(1-q_k)+p_k(1-q_k)+(1-p_k)q_k+p_kq_k=1$$

$\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$. Hence, the average intensity of $\tilde{B}^{UL}_{k,p_k,q_k}$, $\tilde{B}^{UR}_{k,p_k,q_k}$, $\tilde{B}^{LL}_{k,p_k,q_k}$ and $\tilde{B}^{LR}_{k,p_k,q_k}$ will not be boosted up orattenuated down $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$.

If the true motion vector is located at the integer pixel locations, then it is obvious to see that $p_k = q_k = 0$. If the true motion vector is located at the half pixel locations, then it is obvious to see that $p_k = 0$ and $q_k=\frac{1}{2}$, or $p_k=\frac{1}{2}$ and $q_k = 0$, or $p_k=\frac{1}{2}$ and $q_k = 1$, or that $p_k = 1$ and $q_k=\frac{1}{2}$, or $p_k=q_k=\frac{1}{2}$. If the true motion vector is located at the quarter pixel locations, then it is obvious to see that $p_k=q_k=\frac{1}{4}$, or $p_k=\frac{3}{4}$ and $q_k=\frac{1}{4}$, or $p_k=\frac{1}{4}$ and $q_k=\frac{3}{4}$, or $p_k=q_k=\frac{3}{4}$, or $p_k = 0$ and $q_k=\frac{1}{4}$, or $p_k = 0$ and $q_k=\frac{3}{4}$, or $p_k = 1$ and $q_k=\frac{1}{4}$, or $p_k = 1$ and $q_k=\frac{3}{4}$, or $p_k=\frac{1}{4}$ and $q_k = 0$, or $p_k=\frac{3}{4}$ and $q_k = 0$, or $p_k=\frac{1}{4}$ and $q_k = 1$, or $p_k=\frac{3}{4}$ and $q_k = 1$, or $p_k=\frac{1}{2}$ and $q_k=\frac{1}{4}$, or $q_k=\frac{1}{4}$ and $q_k=\frac{1}{2}$, or $p_k=\frac{1}{2}$ and $q_k=\frac{3}{4}$, or $p_k=\frac{3}{4}$ and $q_k=\frac{1}{2}$. Hence, integer pixel locations, half pixel locations and quarter pixel locations are particular locations represented by our proposed model.

Derivation of optimal motion vector: The objective of the block matched motion estimation problem is to find $\forall(p_k,q_k)\in S$ such that MSE_k (p_k, q_k) is minimized $\forall k\in Z^+$. $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$, denote:

$$MSE^{UR}_k(p_k,q_k)\equiv\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ +(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\ +q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1)\\ +p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}+1)\\ -B_{k+1}(x,y)\end{pmatrix}^2$$

$$MSE^{LR}_k(p_k,q_k)\equiv\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ +(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\ +q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}-1)\\ +p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}-1)\\ -B_{k+1}(x,y)\end{pmatrix}^2$$

$$MSE^{UL}_k(p_k,q_k)\equiv\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\ +(1-q_k)p_kB_k(x+p_{0,k}-1,y+q_{0,k})\\ +q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1)\\ +p_kq_kB_k(x+p_{0,k}-1,y+q_{0,k}+1)\\ -B_{k+1}(x,y)\end{pmatrix}^2$$

and:

$$MSE_k^{LL}(p_k,q_k) \equiv \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} (1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k}) \\ +(1-q_k)p_kB_k(x+p_{0,k}-1,y+q_{0,k}) \\ +q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}-1) \\ +p_kq_kB_k(x+p_{0,k}-1,y+q_{0,k}-1) \\ -B_{k+1}(x,y) \end{pmatrix}^2$$

Then $\forall k\in Z^+$ and $\forall(p_k, q_k)\in S$, we have:

$$MSE_k^{UR}(p_k,q_k) = \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} p_kq_k\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix} \\ +q_k\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \\ +p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k})-B_{k+1}(x,y) \end{pmatrix}^2$$

This further implies that $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$:

$$\frac{\partial MSE_k^{UR}(p_k,q_k)}{\partial p_k} = \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left(\begin{pmatrix} p_kq_k\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix} \\ +q_k\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \\ +p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} q_k\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix} \\ +B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}\right)$$

$$= p_kq_k^2\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}^2$$

$$+p_kq_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{4}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}$$

$$+p_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}^2$$

$$+q_k^2\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}$$

$$+q_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} \begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \\ +\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y) \end{pmatrix} \end{pmatrix}$$

$$+\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}.$$

$\forall k\in Z^+$, denote:

$$c_{k,pq^2} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}^2$$

$$c_{k,pq} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{4}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}$$

$$c_{k,p} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}^2$$

$$c_{k,q^2} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k}+1,y+q_{0,k}) \\ +B_k(x+p_{0,k}+1,y+q_{0,k}+1) \end{pmatrix}$$

$$c_{k,q} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\left(\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix} + \begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y) \end{pmatrix}\right)$$

and:

$$c_k \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}$$

Then $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$, we have:

$$\frac{\partial MSE_k^{UR}\left(p_k,q_k\right)}{\partial p_k} = p_k\left(c_{k,pq^2}q_k^2 + c_{k,pq}q_k + c_{k,p}\right) + c_{k,q^2}q_k^2 + c_{k,q}q_k + c_k.$$

Similarly, $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$:

$$\frac{\partial MSE_k^{UR}\left(p_k,q_k\right)}{\partial q_k} = \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left(\left(\begin{matrix} p_kq_k\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix} \\ +q_k\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix} \\ +p_k\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix} \\ +B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y) \end{matrix}\right)\left(\begin{matrix} p_k\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix} \\ +B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{matrix}\right)\right)$$

$$= p_k^2q_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}^2$$

$$+p_kq_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{4}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}$$

$$+q_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}^2$$

$$+p_k^2\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}$$

$$+p_k\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\left(\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix} + \begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y) \end{pmatrix}\right)$$

$$+\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}$$

$\forall k\in Z^+$, denote:

$$z_{k,qp^2} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}^2$$

$$z_{k,qp} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{4}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right) \end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}$$

$$z_{k,q} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right) \end{pmatrix}^2$$

$$z_{k,p^2} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right)\end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right)\end{pmatrix}$$

$$z_{k,p} \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} \begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right)\end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right)\end{pmatrix} \\ +\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k}+1,y+q_{0,k}\right) \\ +B_k\left(x+p_{0,k}+1,y+q_{0,k}+1\right)\end{pmatrix} \\ \begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y)\end{pmatrix}\end{pmatrix}$$

and:

$$z_k \equiv \sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\frac{2}{N^2}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}\right) \\ -B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix} B_k\left(x+p_{0,k},y+q_{0,k}+1\right) \\ -B_k\left(x+p_{0,k},y+q_{0,k}\right)\end{pmatrix}$$

Then $\forall k\in Z^+$ and $\forall(p_k,q_k)\in S$, we have:

$$\frac{\partial MSE_k^{UR}\left(p_k,q_k\right)}{\partial q_k} = q_k\left(z_{k,qp^2}p_k^2 + z_{k,qp}p_k + z_{k,q}\right) + z_{k,p^2}p_k^2 + z_{k,p}p_k + z_k$$

$\forall k\in Z^+$, denote a stationary point of $MSE_k^{UR}(p_k,q_k)$ as $\left(p_k^{UR*},q_k^{UR*}\right)$. Then, $\forall k\in Z^+$, we have

$$\left.\frac{\partial MSE_k^{UR}\left(p_k,q_k\right)}{\partial p_k}\right|_{(p_k,q_k)=\left(p_k^{UR*},q_k^{UR*}\right)} = 0$$

and:

$$\left.\frac{\partial MSE_k^{UR}\left(p_k,q_k\right)}{\partial q_k}\right|_{(p_k,q_k)=\left(p_k^{UR*},q_k^{UR*}\right)} = 0$$

If $c_{k,pq^2}\neq 0$ or $c_{k,pq}\neq 0$ or $c_{k,p}\neq 0$, then we have

$$p_k^{UR*} = -\frac{c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k}{c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}}$$

and:

$$q_k^{UR*}\begin{pmatrix} z_{k,qp^2}\left(-\frac{c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k}{c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}}\right)^2 \\ +z_{k,qp}\left(-\frac{c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k}{c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}}\right) \\ +z_{k,q} \end{pmatrix} + z_{k,p^2}\left(-\frac{c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k}{c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}}\right)^2 + z_{k,p}\left(-\frac{c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k}{c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}}\right) + z_k = 0$$

which further implies that Eq. 1:

$$\begin{aligned} &q_k^{UR*}\begin{pmatrix} z_{k,qp^2}\left(c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k\right)^2 \\ -z_{k,qp}\left(c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k\right)\left(c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}\right) \\ +z_{k,q}\left(c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}\right)^2 \end{pmatrix} \\ &+z_{k,p^2}\left(c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k\right)^2 \\ &-z_{k,p}\left(c_{k,q^2}q_k^{UR*2} + c_{k,q}q_k^{UR*} + c_k\right)\left(c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}\right) \\ &+z_k\left(c_{k,pq^2}q_k^{UR*2} + c_{k,pq}q_k^{UR*} + c_{k,p}\right)^2 = 0 \end{aligned} \quad (1)$$

If $c_{k,pq^2}=0$ and $c_{k,pq}=0$ and $c_{k,p}=0$, but $z_{k,qp^2}\neq 0$ or $z_{k,qp}\neq 0$ or $z_{k,q}\neq 0$, then we have:

$$q_k^{UR*} = -\frac{z_{k,p^2}p_k^{UR*2} + z_{k,p}p_k^{UR*} + z_k}{z_{k,qp^2}p_k^{UR*2} + z_{k,qp}p_k^{UR*} + z_{k,q}}$$

and:

$$c_{k,q^2}\left(-\frac{z_{k,p^2}p_k^{UR*2} + z_{k,p}p_k^{UR*} + z_k}{z_{k,qp^2}p_k^{UR*2} + z_{k,qp}p_k^{UR*} + z_{k,q}}\right)^2 + c_{k,q}\left(-\frac{z_{k,p^2}p_k^{UR*2} + z_{k,p}p_k^{UR*} + z_k}{z_{k,qp^2}p_k^{UR*2} + z_{k,qp}p_k^{UR*} + z_{k,q}}\right) + c_k = 0$$

which further implies that Eq. 2:

$$\begin{aligned} &c_{k,q^2}\left(z_{k,p^2}p_k^{UR*2} + z_{k,p}p_k^{UR*} + z_k\right)^2 \\ &-c_{k,q}\left(z_{k,p^2}p_k^{UR*2} + z_{k,p}p_k^{UR*} + z_k\right)\left(z_{k,qp^2}p_k^{UR*2} + z_{k,qp}p_k^{UR*} + z_{k,q}\right) \\ &+c_k\left(z_{k,qp^2}p_k^{UR*2} + z_{k,qp}p_k^{UR*} + z_{k,q}\right)^2 = 0 \end{aligned} \quad (2)$$

If $c_{k,pq^2}=0$ and $c_{k,pq}=0$ and $c_{k,p}=0$ and $z_{k,qp^2}=0$ and $z_{k,qp}=0$ and $z_{k,q}=0$, then we have Eq. 3:

$$\begin{aligned} &c_{k,q^2}q_k^{UR*2}+c_{k,q}q_k^{UR*}+c_k=0 \text{ and} \\ &z_{k,p^2}p_k^{UR*2}+z_{k,p}p_k^{UR*}+z_k=0 \end{aligned} \tag{3}$$

By solving Eq. 1 or 2 or 3, (p_k^{UR*},q_k^{UR*}) can be found $\forall k\in Z^+$. $\forall k\in Z^+$, denote the total number of vectors (p_k^{UR*},q_k^{UR*}) within S as M_k^{UR}. If $M_k^{UR}\geq 1$, then denote those vectors as $\left(p_{k,m}^{UR*},q_{k,m}^{UR*}\right)$ for $m=1,2,...,M_k^{UR}$ and denote $F_k^{UR}\equiv\left\{\left(p_{k,m}^{UR*},q_{k,m}^{UR*}\right)\right.$ for $m=1,2,...,M_k^{UR}\}\cup\{(0,0)\}$.

However, in general it is not guaranteed that $M_k^{UR}\geq 1$ $\forall k\in Z^+$. If $M_k^{UR}=0$, then there may be no stationary point or the stationary points are not in S. For these two cases, the global minimum of the $MSE_k^{UR}(p_k,q_k)$ could be on the boundaries of S. Hence, it is required to check if there exist some stationary points on the boundaries of S. The following procedures are employed for the checking. $\forall k\in Z^+$ and $\forall q_k\in[0,1]$:

$$\begin{aligned}
&MSE_k^{UR}(0,q_k) \\
&=\left.\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k}) \\ +(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k}) \\ +q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1) \\ +p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_{k+1}(x,y)\end{pmatrix}^2\right|_{p_k=0} \\
&=\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-q_k)B_k(x+p_{0,k},y+q_{0,k}) \\ +q_kB_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_{k+1}(x,y)\end{pmatrix}^2 \\
&=\frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}q_k\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y)\end{pmatrix}^2
\end{aligned}$$

This implies that $\forall k\in Z^+$ and $\forall q_k\in[0,1]$:

$$\begin{aligned}
&\frac{\partial MSE_k^{UR}(0,q_k)}{\partial q_k} \\
&=\frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}q_k\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix} \\
&=\frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}q_k\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix}^2 \\ +\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix}\end{pmatrix}
\end{aligned}$$

$\forall k\in Z^+$, denote:

$$\tilde{c}_{k,0,q}\equiv\frac{2}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix}^2$$

and:

$$\tilde{c}_{k,0}\equiv\frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix}B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix}$$

then $\forall k\in Z^+$ and $\forall q_k\in[0,1]$ we have:

$$\frac{\partial MSE_k^{UR}(0,q_k)}{\partial q_k}=\tilde{c}_{k,0,q}q_k+\tilde{c}_{k,0}.$$

$\forall k\in Z^+$ denote a stationary point of $MSE_k^{UR}(0,q_k)$ as $(0,\tilde{q}_k^{0,UR})$. If $\tilde{c}_{k,0,q}\neq 0$, then

$$\left.\frac{\partial MSE_k^{UR}(0,q_k)}{\partial q_k}\right|_{q_k=\tilde{q}_k^{0,UR}}=0$$

implies that $\tilde{q}_k^{0,UR}=-\frac{\tilde{c}_{k,0}}{\tilde{c}_{k,0,q}}$. If this value is in S, that is if $-\frac{\tilde{c}_{k,0}}{\tilde{c}_{k,0,q}}\in[0,1]$, then this stationary point could be the global minimum. For this case, define $\tilde{F}_{k,0,q}^{UR}\equiv\left\{\left(0,-\frac{\tilde{c}_{k,0}}{\tilde{c}_{k,0,q}}\right)\right\}$.

However, the following three cases could be happened. (Case i) This stationary point may be outside S, that is $\tilde{c}_{k,0,q}\neq 0$ and $-\frac{\tilde{c}_{k,0}}{\tilde{c}_{k,0,q}}\notin[0,1]$. (Case ii) $\tilde{c}_{k,0,q}=0$

and $\tilde{c}_{k,0} = 0$. Then, all the points on the boundary of S are stationary points. (Case iii) $\tilde{c}_{k,0,q} = 0$ and $\tilde{c}_{k,0} \neq 0$. Then, there is no stationary point on the boundary of S. For all these three cases, we do not consider that the global minimum is on the boundary of S. Hence, for these three cases, define $\tilde{F}^{UR}_{k,0,q} \equiv \varphi$, where Ø is denoted as the empty set. Similarly, $\forall k \in Z^+$ and $\forall q_k \in [0,1]$:

$$\mathrm{MSE}_k^{UR}(1,q_k)$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left.\begin{pmatrix}(1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\+(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\+q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1)\\+p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_{k+1}(x,y)\end{pmatrix}^2\right|_{p_k=1}$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-q_k)B_k(x+p_{0,k}+1,y+q_{0,k})\\+q_kB_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_{k+1}(x,y)\end{pmatrix}^2$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}q_k\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}\\+B_k(x+p_{0,k}+1,y+q_{0,k})\\-B_{k+1}(x,y)\end{pmatrix}^2$$

This implies that $\forall k \in Z^+$ and $\forall q_k \in [0,1]$:

$$\frac{\partial \mathrm{MSE}_k^{UR}(1,q_k)}{\partial q_k}$$

$$= \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}q_k\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}\\+B_k(x+p_{0,k}+1,y+q_{0,k})\\-B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}.$$

$$= \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{matrix}q_k\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}^2\\+\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k})\\-B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}\end{matrix}$$

$\forall k \in Z^+$, denote:

$$\tilde{c}_{k,1,q} \equiv \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}^2$$

and:

$$\tilde{c}_{k,1} \equiv \frac{2}{N^2}\sum_{x=0}^{N-1}\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k})\\-B_{k+1}(x,y)\end{pmatrix}\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_k(x+p_{0,k}+1,y+q_{0,k})\end{pmatrix}$$

then $\forall k \in Z^+$ and $\forall q_k \in [0,1]$ we have:

$$\frac{\partial \mathrm{MSE}_k^{UR}(1,q_k)}{\partial q_k} = \tilde{c}_{k,1,q}q_k + \tilde{c}_{k,1}$$

$\forall k \in Z^+$, denote a stationary point of $\mathrm{MSE}_k^{UR}(1,q_k)$ as $(1,\tilde{q}_k^{1,UR})$. If $\tilde{c}_{k,1,q} \neq 0$ and $-\frac{\tilde{c}_{k,1}}{\tilde{c}_{k,1,q}} \in [0,1]$, then this stationary point could be the global minimum. For this case, define $\tilde{F}^{UR}_{k,1,q} \equiv \left\{\left(0,-\frac{\tilde{c}_{k,1}}{\tilde{c}_{k,1,q}}\right)\right\}$.

However, if $\tilde{c}_{k,1,q} \neq 0$ and $-\frac{\tilde{c}_{k,1}}{\tilde{c}_{k,1,q}} \notin [0,1]$, or $\tilde{c}_{k,1,q} = 0$, then we do not consider that the global minimum is on the boundary of S. For these two cases, define $\tilde{F}^{UR}_{k,1,q} \equiv \varphi$. $\forall k \in Z^+$ and $\forall p_k \in [0,1]$:

$$\mathrm{MSE}_k^{UR}(p_k,0)$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left.\begin{pmatrix}(1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k})\\+(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\+q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1)\\+p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}+1)\\-B_{k+1}(x,y)\end{pmatrix}^2\right|_{q_k=0}$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}(1-p_k)B_k(x+p_{0,k},y+q_{0,k})\\+p_kB_k(x+p_{0,k}+1,y+q_{0,k})\\-B_{k+1}(x,y)\end{pmatrix}^2$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix}p_k\begin{pmatrix}B_k(x+p_{0,k}+1,y+q_{0,k})\\-B_k(x+p_{0,k},y+q_{0,k})\end{pmatrix}\\+B_k(x+p_{0,k},y+q_{0,k})\\-B_{k+1}(x,y)\end{pmatrix}^2$$

This implies that $\forall k \in Z^+$ and $\forall p_k \in [0,1]$:

$$\frac{\partial MSE_k^{UR}(p_k,0)}{\partial p_k} = \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}$$

$$= \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}^2 \\ +\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix} \end{pmatrix}$$

$\forall k\in Z^+$, denote:

$$\tilde{z}_{k,0,p} \equiv \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}^2$$

and:

$$\tilde{z}_{k,0} \equiv \frac{2}{N^2}\sum_{x=0}^{N-1}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}) \\ -B_k(x+p_{0,k},y+q_{0,k}) \end{pmatrix}$$

then $\forall k\in Z^+$ and $\forall p_k\in[0,1]$ we have:

$$\frac{\partial MSE_k^{UR}(p_k,0)}{\partial p_k} = \tilde{z}_{k,0,p}p_k + \tilde{z}_{k,0}$$

$\forall k\in Z^+$, denote a stationary point of $MSE_k^{UR}(p_k,0)$ as $\left(\tilde{p}_k^{0,UR},0\right)$. If $\tilde{z}_{k,0,p}\neq 0$ and $-\frac{\tilde{z}_{k,0}}{\tilde{z}_{k,0,p}}\in[0,1]$, then this stationary point could be the global minimum. For this case, define $\tilde{F}_{k,0,p}^{UR} \equiv \left\{\left(-\frac{\tilde{z}_{k,0}}{\tilde{z}_{k,0,p}},0\right)\right\}$.

However, if $\tilde{z}_{k,0,p}\neq 0$ and $-\frac{\tilde{z}_{k,0}}{\tilde{z}_{k,0,p}}\notin[0,1]$, or $\tilde{z}_{k,0,p}=0$, then we do not consider that the global minimum is on the boundary of S. For these two cases, define $\tilde{F}_{k,0,p}^{UR}\equiv\varphi$. Lastly, $\forall k\in Z^+$ and $\forall p_k\in[0,1]$:

$$MSE_k^{UR}(p_k,1) = \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\left.\begin{pmatrix} (1-p_k)(1-q_k)B_k(x+p_{0,k},y+q_{0,k}) \\ +(1-q_k)p_kB_k(x+p_{0,k}+1,y+q_{0,k}) \\ +q_k(1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1) \\ +p_kq_kB_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_{k+1}(x,y) \end{pmatrix}^2\right|_{q_k=1}$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} (1-p_k)B_k(x+p_{0,k},y+q_{0,k}+1) \\ +p_kB_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_{k+1}(x,y) \end{pmatrix}^2$$

$$= \frac{1}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_{k+1}(x,y) \end{pmatrix}^2.$$

This implies that $\forall k\in Z^+$ and $\forall p_k\in[0,1]$:

$$\frac{\partial MSE_k^{UR}(p_k,1)}{\partial p_k} = \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix} \\ +B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix}$$

$$= \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} p_k\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix}^2 \\ +\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix} \end{pmatrix}$$

$\forall k\in Z^+$ denote:

$$\tilde{z}_{k,1,p} \equiv \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix}^2$$

and:

$$\tilde{z}_{k,1} \equiv \frac{2}{N^2}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}\begin{pmatrix} B_k(x+p_{0,k},y+q_{0,k}+1) \\ -B_{k+1}(x,y) \end{pmatrix}\begin{pmatrix} B_k(x+p_{0,k}+1,y+q_{0,k}+1) \\ -B_k(x+p_{0,k},y+q_{0,k}+1) \end{pmatrix}$$

then $\forall k\in Z^+$ and $\forall p_k\in[0,1]$ we have:

$$\frac{\partial MSE_k^{UR}(p_k,1)}{\partial p_k} = \tilde{z}_{k,1,p}p_k + \tilde{z}_{k,1}$$

$\forall k\in Z^+$ denote a stationary point of $MSE_k^{UR}(p_k,1)$ as $(\tilde{p}_k^{1,UR},1)$. If $\tilde{z}_{k,1,p}\neq 0$ and $-\frac{\tilde{z}_{k,1}}{\tilde{z}_{k,1,p}}\in[0,1]$, then this stationary point could be the global minimum. For this case, define $\tilde{F}_{k,1,p}^{UR} \equiv \left\{\left(-\frac{\tilde{z}_{k,1}}{\tilde{z}_{k,1,p}},1\right)\right\}$.

However, if $\tilde{z}_{k,1,p} \neq 0$ and $-\frac{\tilde{z}_{k,1}}{\tilde{z}_{k,1,p}} \notin [0,1]$, or $\tilde{z}_{k,1,p} = 0$, then we do not consider that the global minimum is on the boundary of S. For these two cases, define $\tilde{F}_{k,1,p}^{UR} \equiv \varphi$. $\forall k \in Z^+$, define $F_k^{UR} \equiv \tilde{F}_{k,0,q}^{UR} \cup \tilde{F}_{k,1,q}^{UR} \cup \tilde{F}_{k,0,p}^{UR} \cup \tilde{F}_{k,1,p}^{UR} \cup \{(0,0)\}$. Similarly, $\forall k \in Z^+$, denote the set of motion vectors corresponding to the stationary points of $MSE_k^{UL}(p_k,q_k)$, $MSE_k^{LL}(p_k,q_k)$ and $MSE_k^{LR}(p_k,q_k)$ (including the point $(0,0)$) as F_k^{UL}, F_k^{LL} and F_k^{LR}, respectively. The algorithm for finding the globally optimal motion vector can be summarized as follow:

Algorithm:

Step 1: Implement an existing full integer pixel search algorithm to obtain $(p_{0,k}, q_{0,k})$ $\forall k \in Z^+$.

Step 2: $\forall k \in Z^+$, evaluate F_k^{UL}, F_k^{UR}, F_k^{LL} and F_k^{LR}.

Step3: $\forall k \in Z^+$, evaluate

$$\left(p_k^*, q_k^*\right) \equiv \arg\begin{Bmatrix} \arg\left\{\min\limits_{(p_k,q_k)\in F_k^{UL}} MSE_k^{UL}(p_k,q_k)\right\}, \\ \arg\left\{\min\limits_{(p_k,q_k)\in F_k^{UR}} MSE_k^{UR}(p_k,q_k)\right\}, \\ \arg\left\{\min\limits_{(p_k,q_k)\in F_k^{LL}} MSE_k^{LL}(p_k,q_k)\right\}, \\ \arg\left\{\min\limits_{(p_k,q_k)\in F_k^{LR}} MSE_k^{LR}(p_k,q_k)\right\} \end{Bmatrix}$$

$\forall k \in Z^+$, take (p_k^*, q_k^*) as the globally optimal motion vector of B_k.

Since the global minimum of the mean square error is not necessarily located at rational pixel locations, while the full integer pixel search, full half pixel search and full quarter pixel search algorithms only evaluate at rational pixel locations, the mean square errors based on these conventional methods are very large and these conventional methods are very ineffective. On the other hand, our proposed method guarantee to find the motion vector that globally minimizes the mean square error no matter the motion vector is located at either rational pixel locations or irrational pixel locations. Hence, our proposed method is more effective that conventional methods. Besides, as integer pixel locations, half pixel locations and quarter pixel locations are particular locations represented by our proposed model, the mean square error based on our proposed method is guaranteed to be lower than or equal to that based on these conventional methods.

The computational complexity of our proposed algorithm can be analyzed as follows. As the orders of the polynomials in (1), (2) and (3) are 5, 4 and 2, respectively, $0 \le M_k^{UR} \le 5$ $\forall k \in Z^+$.

Hence, if $M_k^{UR} \ge 1$, then the maximum number of the evaluation points of our proposed method is less than or equal to 21. If $M_k^{UR} = 0$, as the maximum number of the evaluation points in F_k^{UR} is 5, the maximum number of the evaluation points of our proposed method are less than or equal to 17.

For full half pixel search algorithms and full quarter pixel search algorithms, there are 21 and 72 evaluation points, respectively. Hence, the total number of the evaluation points of our proposed method is lower than that of full quarter pixel search algorithms and is lower than or the same as that of the full half pixel search algorithms depending on whether $M_k^{UR} \ge 1$ or not. As conventional block matched motion estimation algorithms evaluate block matching errors from coarse pixel locations to fine pixel locations, the computational complexities grow exponentially as the pixel precisions get finer and finer. From this point of view, the conventional methods are very inefficient. On the other hand, our proposed method does not require searching from the coarse pixel locations to the fine pixel locations. Our proposed method is more efficient than the conventional methods particularly when the required pixel precision is higher than or equal to the quarter pixel precisions.

Optimal motion vectors with arbitrary pixel precisions: For practical motion estimation applications, motion vectors are usually represented by finite pixel precisions. Denote round(z) as the rounding operator that rounds z to the nearest integer and L as the number of bits for the representation of the motion vectors. Then, define:

$$p_{k,L}^* \equiv \frac{\text{round}\left(p_k^* 2^L\right)}{2^L} \text{ and } q_{k,L}^* \equiv \frac{\text{round}\left(q_k^* 2^L\right)}{2^L}$$

Obviously, $p_{k,L}^*$ and $q_{k,L}^*$ are the L bits representation of p_k^* and q_k^*, respectively. It is worth noting that $p_{k,L}^*$ and $q_{k,L}^*$ is the suboptimal solution only. This is because an error may be introduced when applying the rounding operator to p_k^* and q_k^*. Although the globally optimal solution could be found by solving the corresponding integer programming problem, solving the corresponding integer programming problem requires a numerical optimizer and the computational complexities are very high. In fact, the difference between the obtained suboptimal solution and the globally optimal solution is very small. Hence, it is more practical to solve the problem via our proposed method. Also, it is worth noting that the computational complexity of our proposed method is independent of the required pixel precisions. Hence, the computational complexity of our proposed method is lower than that of conventional methods when the required pixel precision is high.

MATERIALS AND METHODS

In order to have complete investigations, video sequences with fast motion, medium motion and slow motion are studied. The video sequences, Foreman, Coastguard and Container, are, respectively, the most common fast motion, medium motion and slow motion video sequences. Hence, motion estimations are performed to these video sequences. Except the first frame of these video sequences, the mean square errors of all the frames of these video sequences are evaluated. Each current frame takes its immediate predecessor as the reference frame. The sizes of the marco blocks are chosen as 8×8 and 16×16 and the sizes of the search windows are chosen as 32×32 and 40×40, which are the most common block sizes and window sizes used in international standards. The comparisons are made with the full integer pixel search algorithm, the full half pixel search algorithm and the full quarter pixel search algorithm.

The mean square error performances of our proposed method with the motion vectors having 1-4 bits representations, the full integer pixel search algorithm, the full half pixel search algorithm and the full quarter pixel search algorithm with the size of the macro blocks 8×8 and the size of the search windows 32×32 applied to the video sequences Coastguard, Container and Foreman are shown in Fig. 1a-c, respectively.

RESULTS

It can be seen from the Fig. 1 that the improvements on the average mean square errors of the full half pixel search algorithm, the full quarter pixel search algorithm, our proposed method with the motion vectors having 1 bit representation, our proposed method with the motion vectors having 2 bits representation, our proposed method with the motion vectors having 3 bit representation and our proposed method with the motion vectors having 4 bits representation over the full integer search algorithm for the video sequences Coastguard are 1.4894×10^{-4}, 2.2242×10^{-4}, 1.4892×10^{-4}, 2.2163×10^{-4}, 2.5293×10^{-4} and 2.6433×10^{-4}, respectively, which correspond to 17.8531%, 28.8039%, 17.8526%, 28.7715%, 34.1366% and 36.2830%, respectively, that for the video sequences Container are 1.4406×10^{-6}, 3.6476×10^{-6}, 1.5171×10^{-6}, 3.7159×10^{-6}, 1.7249×10^{-5} and 1.9126×10^{-5}, respectively, which correspond to 1.0115%, 4.4170%, 1.0432%, 4.4460%, 27.0415% and 30.1629%, respectively and that for the video sequences Foreman are 1.5788×10^{-4}, 2.2863×10^{-4}, 1.5927×10^{-4}, 2.2883×10^{-4}, 2.4908×10^{-4} and 2.5469×10^{-4}, respectively, which correspond to 24.7674%, 39.1977%, 25.0038%, 39.3369%, 44.2051% and 45.5749%, respectively.

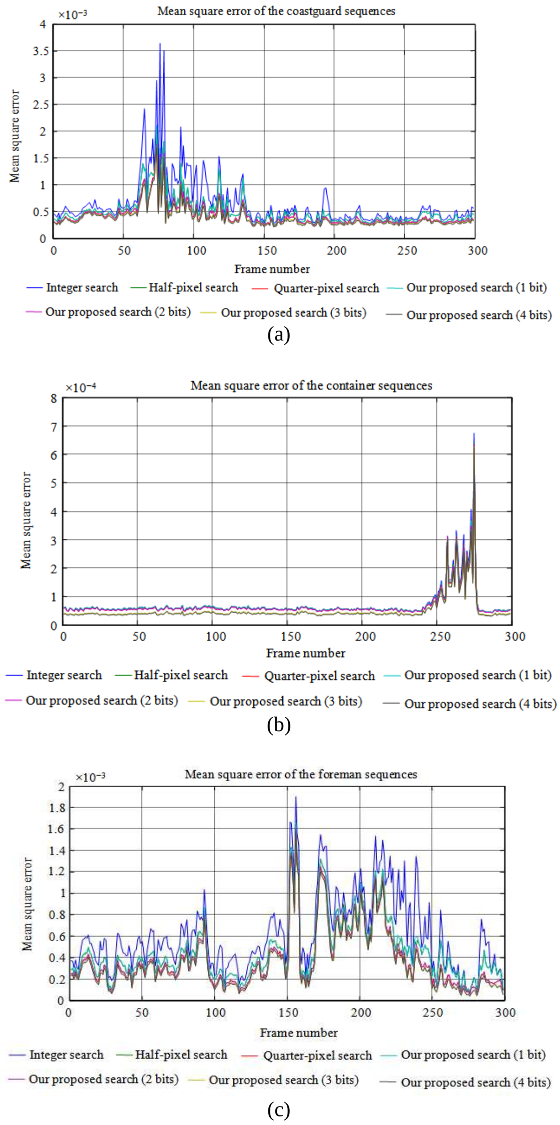
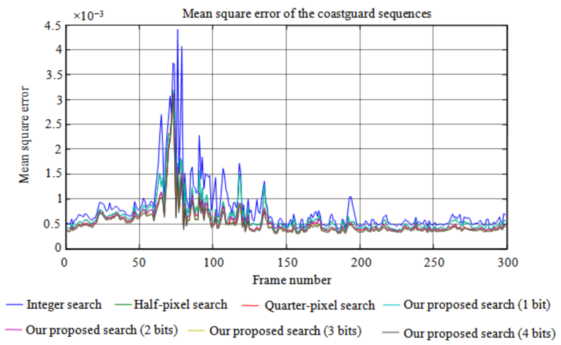
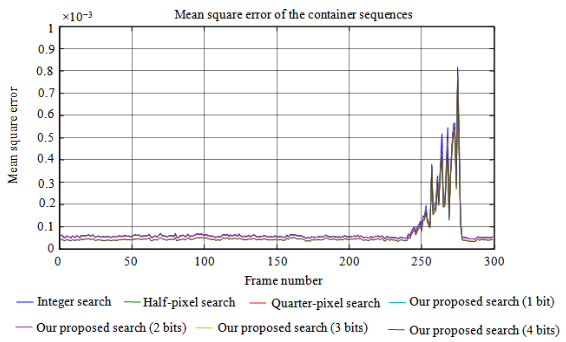


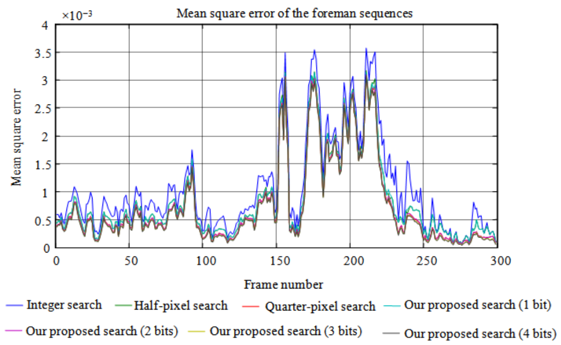
Fig. 1: The mean square error performances of our proposed method with the motion vectors having 1 to 4 bits representations, the full integer pixel search algorithm, the full half pixel search algorithm and the full quarter pixel search algorithm with the size of the macro blocks 8×8 and the size of the search windows 32×32 applied to the video sequences Coastguard, Container and Foreman



(a)



(b)



(c)

Fig. 2: The mean square error performances of our proposed method with the motion vectors having 1 to 4 bits representations, the full integer pixel search algorithm, the full half pixel search algorithm and the full quarter pixel search algorithm with the size of the macro blocks 16×16 and the size of the search windows 40×40 applied to the video sequences Coastguard, Container and Foreman

Similar results are obtained for different size of macro blocks and different size of the search windows. Figure 2 shows the improvements on the average mean square errors of various algorithms with the size of the marco blocks 16×16 and the size of the search windows 40×40 applied to the same set of video sequences. The improvements on the average mean square errors of the full half pixel search algorithm, the full quarter pixel search algorithm and our proposed method with the motion vectors having 1 bit representation, our proposed method with the motion vectors having 2 bits representation, our proposed method with the motion vectors having 3 bit representation and our proposed method with the motion vectors having 4 bits representation for the video sequences Coastguard are 1.7838×10^{-4}, 2.5650×10^{-4}, 1.7828×10^{-4}, 2.5511×10^{-4}, 2.8711×10^{-4} and 2.9943×10^{-4}, respectively, which correspond to 18.4666%, 27.6579%, 18.4517%, 27.5624%, 31.7472% and 33.5302%, respectively, that for the video sequences Container are 1.8757×10^{-6}, 2.5444×10^{-6}, 2.0329×10^{-6}, 2.6633×10^{-6}, 1.4967×10^{-5} and 1.6661×10^{-5}, respectively, which correspond to 0.7710%, 1.5106%, 0.7993%, 1.5294%, 21.7783% and 24.5069%, respectively and that for the video sequences Foreman are 2.1073×10^{-4} , 2.9528×10^{-4} 2.1438×10^{-4}, 2.9723×10^{-4}, 3.2154×10^{-4} and 3.2816×10^{-4}, respectively, which correspond to 21.6021%, 34.2148%, 21.7420%, 34.2884%, 38.6100% and 39.8738%, respectively.

DISCUSSION

From the above numerical computer simulation results, it can be concluded that the mean square error performances of our proposed method with the motion vectors having 1 bit representation is very close to that of the full half pixel search algorithm and that of our proposed method with the motion vectors having 2 bit representation is very close to that of the full quarter half pixel search algorithm. For our proposed method with the motion vectors having more than 2 bits representations, the mean square error performances of our proposed method are always better than that of the full half pixel search algorithm and the full quarter pixel search algorithm for all of the above three video sequences. In particular, for slow motion video sequences, such as the video sequence Container, our proposed method significantly outperforms the full integer pixel search algorithm, the full half pixel search algorithm and the full quarter pixel search algorithm. This is because the globally optimal motion vectors for these slow motion video sequences are very close to the origin and far from the half pixel locations and the quarter pixel locations. In this case, the full half pixel search algorithm and the full quarter pixel search algorithm would not yield very significant improvements over the full integer pixel search algorithm. On the other hand, our proposed method could give a better solution by introducing one more bit for the representation of the motion vectors and hence yields very significant improvements.

CONCLUSION

A nonlinear block matched motion model is proposed in this study. The motion vector with arbitrary pixel precisions which globally minimizes the mean square error is solved analytically in a single step. As integer pixel locations, half pixel locations and quarter pixel locations are particular locations represented by our proposed model, the mean square error based on our proposed method is guaranteed to be lower than or equal to that based on these conventional methods. Also, as our proposed method does not require searching from coarse pixel locations to fine pixel locations, our proposed method is more efficient than conventional methods particularly when the required pixel precision is higher than or equal to the quarter pixel precisions.

ACKNOWLEDGEMENT

Study obtained in this study was supported by a research grant from the Center for Multimedia Signal Processing (under project BB9D), The Hong Kong Polytechnic University.

REFERENCES

Chun, S.Y. and J.A. Fessler, 2009. A simple regularizer for B-spline nonrigid image registration that encourages local invertibility. IEEE J. Selec. Topics Signal Proc., 3: 159-169. DOI: 10.1109/JSTSP.2008.2011116

Du, C., Y. He and J. Zheng, 2003. PPHPS: A parabolic prediction-based, fast half-pixel search algorithm for very low bit-rate moving-picture coding. IEEE Trans. Circ. Syst. Video Technol., 13: 514-518. DOI: 10.1109/TCSVT.2003.813416

Lee, Y.G., J.H. Lee and J.B. Ra, 2003. Fast half-pixel motion estimation based on directional search and a linear model. Proc. SPIE Visual Commun. Imag. Proc., 5150: 1513-1520.

Li, X. and C. Gonzales, 1996. A locally quadratic model of the motion estimation error criterion function and its application to subpixel interpolations. IEEE Trans. Circ. Syst. Video Technol., 6: 118-122. DOI: 10.1109/76.486427

Lin, I.C., J.S. Yeh and M. Ouhyoung, 2002. Extracting 3D facial animation parameters from multiview video clips. IEEE Comput. Graph. Appli, 22: 72-80. DOI: 10.1109/MCG.2002.1046631

Saha, A., J. Mukherjee and S. Sural, 2008. New pixel-decimation patterns for block matching in motion estimation. Signal Proc. Imag. Commun., 23: 725-738. DOI: 10.1016/j.image.2008.08.004