TECHNICAL REPORT

ISO/IEC TR 14496-7

Second edition
2004-10-15

Information technology — Coding of audio-visual objects —

Part 7: Optimized reference software for coding of audio-visual objects

Technologies de l'information — Codage des objets audiovisuels —

Partie 7: Logiciel de référence optimisé pour le codage des objets audiovisuels

Reference number
ISO/IEC TR 14496-7:2004(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

In exceptional circumstances, the joint technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when the joint technical committee has collected data of a different kind from that which is normally published as an International Standard ("state of the art", for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 14496-7, which is a Technical Report of type 3, was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This second edition cancels and replaces the first edition (ISO/IEC 14496-7:2002) which has been technically revised.

ISO/IEC TR 14496 consists of the following parts, under the general title *Information technology — Coding of audio-visual objects*:

— *Part 1: Systems*

— *Part 2: Visual*

— *Part 3: Audio*

— *Part 4: Conformance testing*

— *Part 5: Reference software*

— *Part 6: Delivery Multimedia Integration Framework (DMIF)*

— *Part 7: Optimized reference software for coding of audio-visual objects [Technical Report]*

— *Part 8: Carriage of ISO/IEC 14496 contents over IP networks*

— *Part 9: Reference hardware description [Technical Report]*

— *Part 10: Advanced Video Coding*

— *Part 11: Scene description and application engine*

— *Part 12: ISO base media file format*

— *Part 13: Intellectual Property Management and Protection (IPMP) extensions*

— *Part 14: MP4 file format*

— *Part 15: Advanced Video Coding (AVC) file format*

— *Part 16: Animation Framework eXtension (AFX)*

— *Part 17: Streaming text format*

— *Part 18: Font compression and streaming*

— *Part 19: Synthesized texture stream*

Introduction

Purpose

This part of ISO/IEC 14496 was developed in response to the growing need for optimized reference software that provides both improved visual quality and faster execution while compliance is preserved. The goal is to provide non-normative tools that are essential for implementations of the normative parts of the ISO/IEC 14496 specifications. For example, Part 5 of the ISO/IEC 14496 specifications uses a full search motion estimation which is theoretical optimum in coding efficiency but impractical for commercial implementation. In the past, the industry needs to create its own encoding tools for its target products. In this part, we provide a well-tested set of encoding tools that can enhance the performance but should not be standardized. The following recommended tools would be up to the individual organization to decide if it wishes to adopt or adapt these tools for its specific needs. This part provides significant reduction in the time-to-market and provides a reference benchmark for commercial ISO/IEC 14496 compliant products.

Information technology — Coding of audio-visual objects —

Part 7:
Optimized reference software for coding of audio-visual objects

1 Scope

This part of ISO/IEC 14496 specifies the encoding tools that enhance both the execution and quality for the coding of visual objects as defined in ISO/IEC 14496-2. The tool set is not limited to visual objects but at this point all the recommended tools are visual encoding tools. There are four tools that have been described in this technical report.

- Fast Motion Estimation
- Fast Global Motion Estimation
- Fast and Robust Sprite Generation
- Fast Variable Length Decoder Using Hierarchical Table Lookup

These tools have been demonstrated as robust tools with source codes for both MoMusys and Microsoft implementations. In the current implementations, there is single software that includes all tools existed in the ISO/IEC 14496-2. This is obviously inefficient in terms of code size and execution speed. To address this issue, the optimized reference software has compilation switches such that only selected tools as defined by the profiles and levels are included. Such level of optimization is performed at high level programming language. The platform specific optimization is currently not addressed by this part.

2 Fast Motion Estimation

2.1 Introduction to Motion Adaptive Fast Motion Estimation

The optimization of fast motion estimation is essentially a multi-dimensional problem. The key dimensions concerned in this problem are: Rate, Quality (PSNR), Speed-up (or Computational Gain), Algorithmic Complexity, Memory Size and Memory Bandwidth (see Figure 1). There always exists a trade-off among all these five key dimensions. Therefore, it is highly desirable to have an adaptive fast motion estimation core algorithm with scalable structure, which can be adaptively optimized with respect to all or selected aspects for various coding environment and requirements. Since the rate control is used to fix the bit-rate, the optimization problem is reduced by one dimension to four dimensions.

Motion Vector Field Adaptive Search Technique (MVFAST) [1] is a generic algorithm of the family of *motion-adaptive* fast search techniques, originally proposed by Kai-Kuang Ma and Prabhudev Irappa Hosur from Nanyang Technological University (NTU), Singapore. The MVFAST offers high performance both in quality and speed and does not require memory to store the searched points and motion vectors. The MVFAST has been adopted by MPEG-4 Part 7 in the Noordwijkerhout MPEG meeting (March 2000) as the *core technology* for fast motion estimation.

A derivative of MVFAST, called *Predictive* MVFAST (PMVFAST) [2], is considered as an *optional approach* that might benefit in special coding situations. PMVFAST incorporates a set of thresholds into MVFAST to trade higher speed-up at the cost of memory size, memory bandwidth and additional algorithmic complexity. In PMVFAST, the threshold values are adjusted based on the 54 test cases specified by MPEG-4. However, the coding performance and sensitivity of PMVFAST using these thresholds for the video sequences and encoding conditions outside the MPEG-4 test set has *not* been studied and verified.

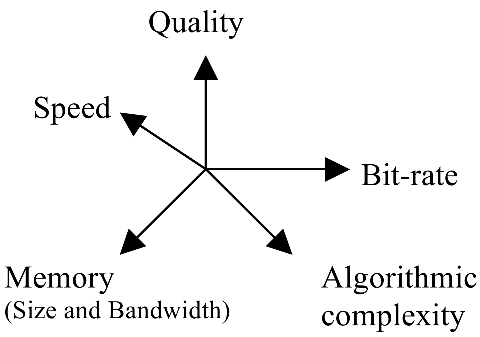


Figure 1 — Five dimensional optimization problem of fast motion estimation

2.2 Technical Description of Core Technology — MVFAST

2.2.1 Detection of stationary blocks

A large number of MBs in the video sequences (e.g., "talking head" video sequences) with low-motion content tend to have motion vectors equal to (0,0). Such MBs in the regions of no-motion activity can be detected simply based on the sum of absolute difference (SAD) at the origin. Therefore, we exploit an optional phase, called *early elimination of search*, as the first step in MVFAST as follows. The search for a MB will be terminated immediately, if its SAD value obtained at (0,0) is less than a threshold T, and the motion vector is assigned as (0,0). Through extensive simulations, we found that among those zero-motion blocks identified, about 98% of them have their SAD at position (0,0) less than 512. Hence, we choose T = 512 to enable the mechanism of early elimination of search. Since this early elimination of search phase is optional, it can be turned off or disabled by imposing T = 0.

2.2.2 Determination of local motion activity

The *local motion vector field* at a macroblock (MB) position is defined as the set of motion vectors in the *region of support* (ROS) of that MB. The ROS of a MB includes the n neighborhood MBs. In MVFAST, the ROS with n = 3 is shown in Figure 2. Let $V=\{V_0, V_1, \ldots V_n\}$, where $V_0 = (0,0)$, and V_i (and $i \neq 0$) is the motion vector of MB_i in the ROS (see Figure 2). The cityblock length of $V_i=(x_i, y_i)$ is defined as $l_{vi} = |x_i| + |y_i|$. Let $L = MAX\{l_{vi}\}$ for all V_i. The motion activity at the current MB position is defined as follows.

$$\text{Motion Activity} = \begin{cases} \text{Low, if } L \leq L1; \\ \text{Medium, if } L1 < L \leq L2; \\ \text{High, if } L > L2; \end{cases} \quad (1)$$

where L_1 and L_2 are integer constants. We choose L_1 and L_2 as the cityblock distance from the center point of the pattern to any other point on the small and large search patterns (see Figure 3), respectively. Thus, L_1 =1 and L_2 =2.

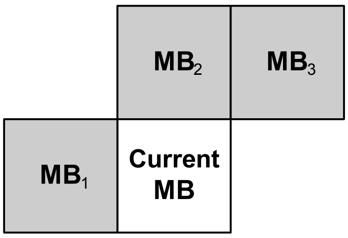


Figure 2 — Region of support (ROS) for the current MB consists of MB1, MB2 and MB3

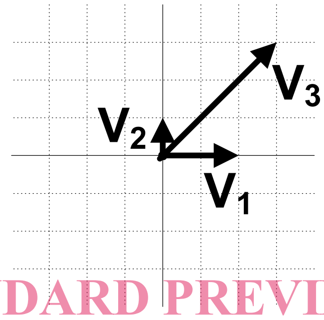
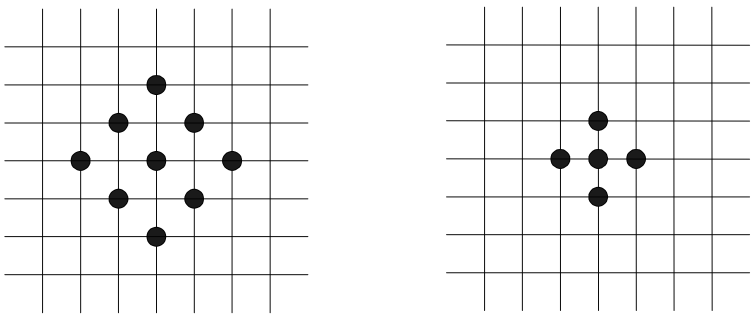


Figure 3 — Example of distribution of motion vectors belonging to set V. In this case, lv1 = 2, lv2 = 1, lv3 = 6; thus L = MAX{lv1, lv2, lv3} = 6

2.2.3 Search Center

The choice of the search center depends on the local motion activity at the current MB position. If the motion activity is low or medium, the search center is the origin. Otherwise, the vector belonging to set V that yields the minimum sum of absolute difference (SAD) is chosen as the search center.



(a) (b)

Figure 4 — (a) Large Diamond Search Pattern (LDSP) and (b) Small Diamond Search Pattern (SDSP)

2.2.4 Search Strategy

A local search is performed around the search center to obtain the motion vector for the current MB. The search patterns employed for the local search are shown in Fig. 4. Two strategies are proposed for the local search and their choice depends on the motion activity identified. If the motion activity is low or high, we employ small diamond search (SDS). Otherwise, we choose large diamond search (LDS).

i) *Small Diamond Search* (SDS)

Step 1: Small diamond search pattern (SDSP) is centered at the search center, and all the checking points of SDSP are tested. If the center position yields the minimum SAD (i.e., no motion), then the center represents the motion vector; otherwise, go to Step 2.

Step 2: The center of SDSP moves to the point where the minimum SAD was obtained in the previous step, and all the points on SDSP are tested. If the center position yields the minimum SAD, then the center represents the motion vector; otherwise, recursively repeat this step.

ii) *Large Diamond Search* (LDS)

Step 1: Large diamond search pattern (LDSP) is centered at the search center, and all the checking points of LDSP are tested. If the center position gives the minimum SAD, go to Step 3; otherwise, go to Step 2.

Step 2: The center of LDSP moves to the point where the minimum SAD was obtained in the previous step, and all the points on LDSP are tested. If the center position gives the minimum SAD, go to Step 3; otherwise, recursively repeat this step.

Step 3: Switch the search pattern from LDSP to SDSP. The point that yields the minimum SAD, is the final solution of the motion vector.

Table 1 summarizes the methodology for selection of search center and search strategy depending on the motion activity at the current MB position.

Table 1 — The search modes for MVFAST

Motion Activity	Search Center	Search Strategy
Low	Origin	SDS
Medium	Origin	LDS
High	The position of the vector in set *V* that yields minimum SAD	SDS

2.2.5 Perspectives on implementing MVFAST

The MVFAST algorithm can be structured in terms of *profiles*. The MVFAST itself as described above can be viewed as the **main profile**. The low, medium and high motion activity cases in Table 1 can be considered individually as three other different profiles of MVFAST. Depending on the video coding applications, any one of these individual profiles can be turned "ON" simply by adjusting the two parameters, L_1 and L_2, in Equation (1). If we set $L_1 = L_2 =$ Search Range, we obtain "low motion activity" profile. The "medium motion activity" profile (which is the same as Diamond Search, as described in VM Version 14) can be obtained, if we set $L_1 = -1$ and $L_2 =$ Search Range. For "high motion activity" profile, we can set $L_1 = L_2 = -1$. Note that in this case, Search Range = 2*N, if the search in either coordinate is in the range [–N, N-1].

Although MVFAST is implemented in an intelligent way such that the overlap of search points is minimized when the search pattern moves, few search points are visited more than once. This overlap can be avoided by keeping the record of all the search points visited and testing if the current search point is visited earlier. Thus further improvement over speed-up can be achieved.

The search point (0,0) is always tested in MVFAST. However, some improvement in computational gain is obtained by testing (0,0) point only, if any of the motion vectors in the ROS has motion vector = (0,0).

Through extensive experiments using MVFAST, it is found that further improvement in objective quality can be achieved when interlaced CCIR sequences with high global motion are coded in progressive mode, by including the motion vector of collocated block on the previously coded non-intra frame in the set *V*. During the motion estimation of interlaced pictures, each frame prediction of macroblock motion is performed before field motion estimation. Therefore, for field motion estimation of current macroblock, its frame motion vector is included in set *V*.

From hardware implementation viewpoint, to restrict the total number of search points for a block in the worst case to be *N*, an additional stopping criterion — "stop the search when the number of search points visited so far is equal to *N*", can be included in SDS and LDS given in subclause 2.4.

2.2.6 Special Acknowledgements

Kai-Kuang Ma and Prabhudev Irappa Hosur would like to sincerely acknowledge tremendous support from Professor Meng Hwa Er, Dean, School of Electrical and Electronic Engineering, and Deputy President of Nanyang Technological University, Singapore, who plays a vital role on promoting and directing all Singapore MPEG activities. For independent verification efforts, the following individuals are greatly acknowledged: Dr. Weisi Lin, Mr. Chengyu Xiong, Dr. Ee Ping Ong, all from Institute of Microelectronics (IME), Singapore.

CONTACT PERSON:

Dr. Kai-Kuang Ma, School of Electrical and Electronic Engineering, Nanyang Technological University, Block S2, Nanyang Avenue, Singapore 639798. Tel: +65-790-6366; Fax: +65-792-0415; Emails: ekkma@ntu.edu.sg and kaikuang@hotmail.com.

2.3 Technical Description of PMVFAST

2.3.1 Introduction

This section provides the technical description of the *Predictive Motion Vector Field Adaptive Search Technique* (PMVFAST) which adds some techniques from the *Advance Predictive Diamond Zonal Search* (APDZS) [2] proposed by the Hong Kong University of Science and Technology (HKUST) to the MVFAST core mentioned above to achieve larger speed up. The PMVFAST was contributed by Prof. Ming L. Liou, Dr. Oscar C. Au, and Alexis Tourapis of HKUST. PMVFAST is faster than MVFAST at the expense of higher hardware complexity

Several independent parties, Optivision Inc., Sarnoff Co., Mitsubishi Electric Information Technology Center America, National Technical University of Athens (NTUA), and Beijing University of Aeronautics and Astronautics (BUAA), conducted evaluation throughout the entire adoption process. For independent verification efforts, the following individuals are greatly acknowledged: Dr. Weiping Li (from Optivision), Dr. Hung-Ju Lee and Dr. Tihao Chiang (from Sarnoff), Mr. Anthony Vetro and Dr. Huifan Sun (from Mitsubishi), Mr. Gabriel Tsechpenakis, Mr. Yannis Avtithis and Prof. Stefanos Kollias(from NTUA), and Prof. Bo Li, Yaming Tu (from BUAA).

2.3.2 Technical Description of PMVFAST

PMVFAST combines the ‘stop when good enough’ spirit, the thresholding stopping criteria and the spatial and temporal motion vector prediction of APDZS and the efficient large and small diamond search patterns of MVFAST. Let the *refBlock* be the block in the reference frame at the same spatial location as the current block. Without loss of generality, the distortion criterion is assumed to be the Sum-of-Absolute-Difference (SAD), though it can be other measures. The predicted motion vector in PMVFAST is the median of the motion vectors of three blocks spatially adjacent to the current block (left, top and top right), as in MPEG motion vector predictive coding.

Firstly, the PMVFAST computes the SAD of the predicted motion vector (PMV), and stops if any one of two stopping criteria is satisfied. The first criterion is that the PMV is equal to the motion vector of refBlock and the SAD of PMV is less than that of refBlock. The second criterion is that the SAD of PMV is less than a threshold.

Secondly, the PMVFAST computes the SAD of some highly-probable motion vectors (MV of left, top and top right spatially neighboring blocks, MV of (0,0) and MV of refBlock) and stops if any one of two stopping criteria is satisfied. The first criterion is that the best motion vector so far is equal to the MV of refBlock and the minimum SAD so far (MinSAD) is less than that of refBlock. The second criterion is that the MinSAD is less than a threshold.

Thirdly, the PMVFAST selects the MV associated with minSAD and performs a local search using techniques of MVFAST. If PMV is equal to (0,0) and the motion vectors of the three spatially adjacent blocks are identical with large associated SAD, the large diamond search of MVFAST is applied. Otherwise, if the motion vectors of the three spatially adjacent blocks are identical and are the same as the MV of refBlock, small diamond search is applied with the simplication that only one small diamond pattern is examined. Otherwise, the small diamond search of MVFAST is applied.

Here is the step-by-step algorithm of PMVFAST. The variables *thresa*, *thresb* are integers used as thresholds in the stopping criteria.

(Initialization)

Step 1: Set thresholding parameters (*thresa* & *thresb*). These are set as follows:
If first row and column, *thresa* = 512, *thresb* = 1024
Else *thresa* = minimum value of the sad of left, top and top-right blocks. *thresb* = *thresa* + 256;
If *thresa<512, thresa = 512.* If *thresa* > 1024, *thresa* = 1024.
If *thresb* > 1792, *thresb* = 1792.
Set *Found*=0 and *PredEq*=0
Compute the predicted MV according to the Median rule.
Select previous MV, above, and above-right and calculate median.
If block is an edge block, depending to the position, do the following:
If block is on the first column, assume previous MV to be equal to (0,0).
If block is on the first row, select previous MV as the prediction.
If block is on the last column, assume above right MV to be equal to (0,0).
If left MV = top MV = top-right MV then set *PredEq*=1;

(Initial prediction calculation)

Step 2: Calculate *Distance*= $|MedianMV_X|$ + $|MedianMV_Y|$ where MedianMV is the motion vector of the median.
If *PredEq*=1 and $MV_{predicted}$ = Previous Frame MV, set Found=2
Step 3: If *Distance*>0 or *thresb*<1536 or *PredEq*=1.
Select small Diamond Search. Otherwise select large Diamond Search.
Step 4: Calculate SAD around the Median prediction. MinSAD=SAD
If Motion Vector equal to Previous frame motion vector and MinSAD<PrevFrmSAD goto Step 10.
If SAD<=256 goto Step 10.
Step 5: Calculate SAD for motion vectors taken from left block, top, top-right, and Previous frame block. Also calculate (0,0) but do not subtract offset.
Let MinSAD be the smallest SAD up to this point. If MV is (0,0) subtract offset.

Step 6: If MinSAD <= *thresa* goto Step 10.
If Motion Vector equal to Previous frame motion vector and MinSAD<PrevFrmSAD goto Step 10.

(Diamond Search)

Step 7: Perform Diamond search, with either the small or large diamond. If *Found*=2 only examine one Diamond pattern, and afterwards goto step 10

Step 8: If small diamond, iterate small diamond search pattern until motion vector lies in the center of the diamond. If center then goto step 10.

Step 9: If large diamond, iterate large diamond search pattern until motion vector lies in the center. Refine by using small diamond and goto step 10.

(Final step. Use best MV found.)

Step 10: The motion vector is chosen according to the block corresponding to MinSAD.
By performing an optional local half-pixel search, we can refine this result even further.

2.3.3 Special Acknowledgement

M.L. Liou, O.C. Au, A.M. Tourapis would like to sincerely acknowledge the tremendous support from Prof. Ishfaq Ahmad, the Department of Electrical and Electronic Engineering of the Hong Kong University of Science and Technology (HKUST), the Hong Kong Telecom Institute of Information Technology (HKTIIT), the Hong Kong Reseach Grants Council (RGC).

CONTACT PERSON: Dr. Oscar C. Au, Department of Electrical and Electronic Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong, China. Tel: +852 2358-7053; Fax: +852 2358-1485; Emails: eeau@ust.hk.

2.4 Conclusions

The comparison of MVFAST vis-à-vis PMVFAST is given in Table 2.