Volume 2, Issue 6, June 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

The role of Block matching algorithm in Stereo image Compression

T.Ramaprabha MSc MPhil
Lecturer, Dept of Computer Science
Sarah Tucker College (Autonomous)
Tirunelveli– 627007 ,South India

Dr.M.Mohamed Sathik
Principal
Sadakkathella Appa College(Autonomous)
Tirunelveli-627011,South India

Abstract : 3D stereo images are acquired by simulating human's eyesight effect upon observing objects through two horizontally separated perspectives. Correspondingly, two frames are resulted for one 3-D image, labelled as left frame and right frame. If these two frames are to be transmitted with the idea of reconstructing the 3D image at the receiver end, we would need double the bandwidth required for monocular image transmission. Therefore, data compression is necessary. The left and right views that comprise a 3D-stereoscopic image pair or motion stream pair are obviously very similar. They are often described as "highly redundant", in that most of the information contained in either is repeated in the other, or as "highly correlated" in that either is for the most part easily predicted from the other by application of some external information about the relationship between them. We can thus synthesize a reasonable approximation to either view given the other view and a little additional information that describes the relationship between the two views. A useful form for the additional information is a disparity map: a two dimensional vector field that encodes how to displace blocks of pixels in one view to approximate the other view. In this paper I wish to illustrate how the block matching algorithm works and helpful for the effective stereo image compression.

Keywords : Stereo Image, Motion estimation ,Block matching algorithm Simple and 1Dfull search algorithm.

I. Introduction

Stereoscopic image pairs represent a view of the same scene from two slightly different positions. When the images are presented to the respective eye the human observer perceives the depth in the scene as in 3 dimensions. One can obtain stereo pairs by taking pictures with two cameras that are placed in parallel 2-3 inches apart. the same point in the object will be mapped to different coordinates in the left and right images. block-based motion-compensation used to reduce the memory requirements of video files. This method takes advantage of temporal redundancy between two or more frames

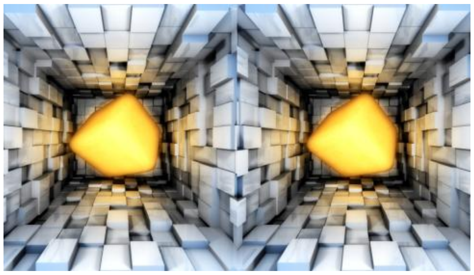


Fig 1.Stereoscopic image

II. Motion estimation

When using motion estimation for compressing video data, this compression technique uses the fact that usually not the whole image content changes from one frame of a video to the next, but only regions. Often these regions do not disappear from the image, they just change their location within the image[4] . The object itself also changes its appearance only slightly on its way through the scene. The idea of motion estimation is to detect all objects within an image, compute their motion and represent it by motion vectors.

..

Fig 1.Original

The advantage of this technique is, that the image data for the background and for the objects is stored only in one frame - the following frames contain the motion vectors. A static background then is represented by a zero vector because it has not moved whereas the object movement is stored with non-zero motion vectors, pointing to the new locations of the objects. If the camera position is not fixed, the static background also turns into a moving object. As mentioned , the motion of semantic objects in videos should be represented by motion vectors. The problem is to detect the objects and to find their exact boundaries. The motion of semantic objects in videos should be represented by motion vectors.

Fig 2. Difference between origional and next frame

III. Basics of Block matching algorithm

In block matching, an image on the screen is divided into a grid of blocks. One block of the image is compared a pixel at a time with a block in the same place in the next image. If there is no motion, there is a high correlation between the two blocks. If something has shifted, the same place position in the next field will not provide good correlation, and it will be necessary to search for the best correlation in the next image. The position that gives the best correlation is assumed to be the new location of a moving object. Block matchers work at the pixel level, so they are poor at tracking larger objects and high-motion speeds.

To overcome this, hierarchical techniques are used, where block matching is initially carried out with large blocks. Then the process is repeated at subdivided blocks all the way down to pixel resolution. Such hierarchical techniques try to balance the ability to track large objects versus small objects. In practice, however, they fall short on accuracy when tracking small, fast-moving objects, such credit roll text. They also tend to introduce spurious motion vectors on cuts, fades, sources with noise and other irregular material.

MPEG encoders use block matching algorithms to relocate an object in another frame. Therefore the image is segmented into a no of rectangular blocks of 8 by 8 pixels.[5] Depending on the algorithm used for motion estimation, a block within a search range is compared with the source block. Block matching uses a value called “block distortion measure” - BDM - to rate the similarity between two blocks. The basic idea is to sum up the differences of the pixel luminance’s of pixels located at the same position in the two blocks. There are different algorithms to perform this calculation:

Let Ci(x,y) detect the luminance of pixel (x,y) in block i. The „ Mean Squared Error“ (MSE) uses the sum of squared differences between the two luminance values. The sum is divided by the quantity of the compared pixels to normalize the result:

$$D=\frac{1}{(n*m)}*\sum_{y=0}^{m-1}\sum_{x=0}^{n-1}(c1(x,y)-c2(x,y))^2$$

The „Sum of Absolute Differences “ (SAD) differs only slightly by using absolute differences instead of squared differences:

$$D=\frac{1}{(n*m)}*\sum_{y=0}^{m-1}\sum_{x=0}^{n-1}\left|(c1(x,y)-c2(x,y))\right|$$

It is not necessary to normalize the result in the end, since we only need the minimum cost block position and not the absolute cost itself. In fact, it costs an extra mathematical operation per comparison. We used normalized costs to make the results independent of a specific block size. Usually, the motion estimation algorithms do not search a whole frame for the best matching block.

A search range defines a search area around the coordinates of the source block[6]. It is necessary to understand that the source block is from a frame A, and the search area located in a frame B. Usually these are successive frames in a video[1]. The most simple search algorithm, full search , to find the block with the lowest BDM value within a specified search area, is to match the source block to every existing block in this area. The search-area is a rectangular area around the source block[2]. Its size, and therefore the number contained search coordinates, depends on the horizontal search range r_x and the vertical search range r_y

Size of search area is $(2*r_x+1)*(2*r_y+1)$

Consider r_x and r_y are of the same size, therefore only use the value r[7]. Matching all those blocks in the search area to the source block is the computationally most intensive algorithm: For a search range of 10 pixels and also a block size of 8 by 8 pixels, there are (2∗ 10+1) 2 = 441 blocks to be matched with 64 operations per block. k.

IV. Block Matching Algorithm

A Block Matching Algorithm (BMA) is a way of locating matching blocks in a sequence of digital video frames for the purposes of motion estimation. The purpose of a block matching algorithm is to find a matching block from a frame I in some other frame, j which may appear before or after i . This can be used to discover temporal redundancy in the video sequence, increasing the effectiveness of interframe video compression and television standards conversion. Successive video frames may contain the same objects.

Motion estimation examines the movement of objects in an image sequence to try to obtain vectors representing the estimated motion. Motion compensation uses the knowledge of object motion so obtained to achieve data compression. In interframe coding, motion estimation and compensation have become powerful techniques to eliminate the temporal redundancy due to high correlation between consecutive frames.

V. FULL Search Algorithm

A grid is used to present a schematic overview of the search area, as displayed in Fig. 4. The centre of the grid is the coordinate (0,0). Red dots highlight the search coordinates, the respective MEA will compare with the source block. Each grid coordinate represents a distance of one pixel. If new search coordinates are added to the grid, the newer ones are displayed larger than the older ones.

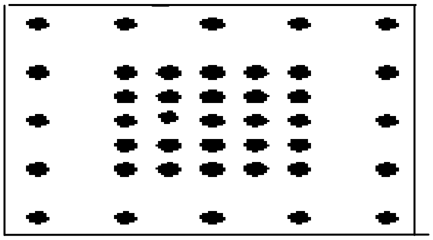


Fig 3.Grid-Search area

Full Search Algorithm is a simple MEA that compares the source block with the blocks at every position within the search area. As mentioned above, the calculation costs of this algorithm are very high, but it find the optimal block position within the search range. The number of comparisons increases quadratically (n2) with the search range[2]:

The number of comparisons is (2r + 1)2

The steps for 1DFSA the algorithm is,

1. Start with a search pattern of all coordinates (x,0) within the search area and find the
Coordinate with the lowest BDM value.

2. Continue by creating a new search pattern, consisting of all coordinates that are vertically lined up with the current best matching position. Again search the block with the lowest BDM value.

3. Halve the search range. Proceed like in step 1, this time with the row of the current best matching block.

4. Step 2 is repeated with the reduced search range

The no of comparison is calculated by

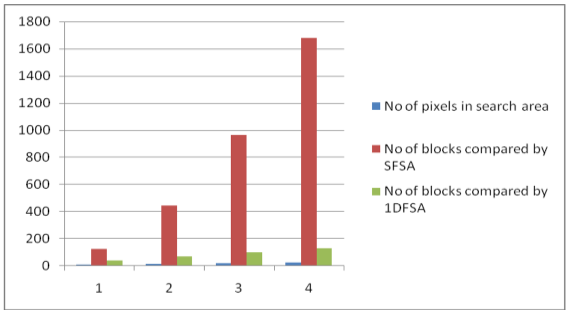
2 x (2r + 1) + 2 (1/2r +1)

One-dimensional full search performs faster than simple full search. The number of compared blocks just increases linearly when increasing the search range.

Table 1. Performance of full search algorithm

No of pixels in search area	No of blocks compared by SFSA	No of blocks compared by 1DFSA
5	121	36
10	441	64
15	961	96
20	1681	124

Chart 1.Comparision of performance of Simple full search algorithm
Vs ID full search algorithm.



VI. Conclusion

The idea of motion estimation is to detect all objects within an image, compute their motion and represent it by motion vectors. The advantage of this technique is, that the image data for the background and for the objects is stored only in one frame - the following frames contain the motion vectors. A static background then is represented by a zero vector because it has not moved whereas the object movement is stored with non-zero motion vectors, pointing to the new locations of the objects. If the camera position is not fixed, the static background also turns into a moving object. This study presents a survey of simple full search motion estimation algorithms and 1Dfull search algorithm and they are illustrates the described algorithms with a step-by-step explanation. Additionally, the matching accuracy are measured, which enables a quantitative comparison. of both the algorithms.

Reference

[1] Mei-Juan Chen, Liang-Gee Chen, Tzi-Dar Chiueh: "One-Dimensional Full Search Motion Estimation Algorithm For Video Coding" IEEE Transaction On Circuits Ans Systems for Video Technology, Vol. 4 No. 5, page 504-509, October 1994

[2]. Th. Zahariadis, D. Kalivas "A Spiral Search Algorithm For Fast Estimation Of Block Motion Vectors" Signal Processing VIII, theories and applications.

Proceedings of the EUSIPCO 96. Eighth European Signal Processing Conference p.3 vol. lxiii +2144,1079-82,vol.2.

[3]. M.Ghanbari "The cross search algorithm for Motion Estimation" IEEE Transactions of Communications, vol. 38, no. 7 pp 950-3, July1990.

[4]. Garnham, N. W., Motion Compensated Video Coding, University of Nottingham PhD Thesis, October 1995, ISBN X763409714

[5]. Bhaskaran, V, Konstantinides, K, "Image and video compression standards - algorithms and architectures", Kluwer academic publishers, 1996

[6]. Netravali, A N and Haskell, B G, "Digital Pictures: Representation, Compression and Standards", 2nd Edition, Plenum Press, 1995.

[7]..Ghanbari, "Video coding: an introduction to standard codecs", IEE Press, 1997

[8]. A. Puri, H. Hang and D. Schilling, "An efficient block-matching algorithm for motion compensated coding", Proceedings IEEE ICASSP, pp. 25.4.1 - 24.4.4, 1987.