

A modified 2-d logarithmic search technique for video coding

[Technology](#)



A significant step in video coding is searching for similar segments in previous frames and use only the difference information for reconstruction thus reducing space requirement. Different search techniques including Full search and 2-D logarithmic search etc. are used in the current literature. Full search restricts its application because of its computational load. 2D logarithmic search is computationally less expensive although there are some spaces for improvement. In this paper we propose a new search technique by modifying the 2-D logarithmic search that requires less search points with insignificant loss in visual quality.

Experimental results demonstrate the effectiveness of the proposed technique. Keywords: video coding, 2-D logarithmic search.

INTRODUCTION

Video is a sequence of still images representing scenes in motion. A video is created by capturing a numbers of still images in a short time interval. When these still images are displayed very quickly, it represents the motion of the object in the images. Video represent the huge amount of data. In order to transfer video data from one place to another efficiently it is required to compress the size of video data.

One way to compress the size of video data is video coding [[1]] [[2]]. The principal goal in the design of a video-coding system is to reduce the transmission rate subject to some picture quality constraint. In transmission side, the first frame (normally called the reference frame) is transmitted as it is and the remaining frames are sent as a function of the reference frame. The frame to be sent is divided into a number of blocks and the best match

for the block is looked for in the search window of the reference frame. This processing is called the search technique in video coding literature.

There exist a number of video coding techniques including MPEG-1/2/4 [[2]] [[7]], H. 26X [[8]] etc. uses search techniques like Full search [[1]], 2-D logarithmic search [[3]], Coarse-Fine-Three-Step search [[4]], Conjugate Direction search [[5]], and Pyramid search [[6]]. Each of these search techniques has merits and demerits in their favor. Full search finds the best match for a block as it searches all the candidate positions in the search window. Full search however is computationally expensive and renders difficulty for real time implementation.

Some variants exist that applies some heuristics to reduce the candidate search points and reduce the computational complexity although compromising the image quality a bit. 2-D logarithmic search is one such search technique that reduces the search points to a subset of the search window (to be detailed in literature review) and finds the near-optimal best match with reduced computational complexity. Although computationally inexpensive it contains some redundancy in the search space. We aim to reduce this redundancy and aim to find a modified 2-D logarithmic search technique with even reduced computational load.

Experimental results demonstrate that the proposed technique reduces the number of search points and thus reduces search time with insignificant sacrifice of image quality. The paper is organized as follows. In Section II we elaborate some related works. In Section III we present our proposed search approach. Some experimental results to demonstrate the effective of the proposed approach is presented in Section IV. Finally Section V concludes <https://assignbuster.com/a-modified-2-d-logarithmic-search-technique-for-video-coding/>

the paper. II. Related works In this section we present full search technique and the logarithmic search technique.

In both cases the frame to be coded is divided into a number of non-overlapping equal size blocks of size $p \times q$. The best match is looked for in a search window of size $(2d+1) \times (2d+1)$ in the reference frame. Fig 1: Block matching process in video coding that uses search techniques. A. Full Search In Full search [[1]] finds the best match by inspecting all the $(2d+1) \times (2d+1)$ candidate positions within the search window. Full search procedure is brute force in nature. The advantage of Full Search is that it delivers good accuracy in searching for the best match.

The disadvantage is that it involves a large amount of computation. B. 2-D Logarithmic Search Jain and Jain [[3]] developed a 2-D logarithmic search technique that successively reduces the search area, thus reducing the computational burden. The first step computes the similarity for five points in the search window. These five points are as follows: the central point of the search window and the four points surrounding it, with each being a midpoint between the central point and one of the four boundaries of the window. Among these five points, the one corresponding to the minimum dissimilarity is picked as the winner.

In the next step, surrounding this winner, another set of five points are selected in a similar fashion to that in the first step, with the distances between the five points remaining unchanged. The exception takes place when either a central point of a set of five points or a boundary point of the search window gives a minimum dissimilarity. In these circumstances, the distances between the five points need to be reduced. The procedure <https://assignbuster.com/a-modified-2-d-logarithmic-search-technique-for-video-coding/>

continues until the final step, in which a set of candidate points are located in a 3x3 2-D grid.

The steps in a 2-D logarithmic search technique are presented in Fig 2. Fig 2: The 2-D logarithmic search technique. The circle numbered n is searched at the n -th step. The arrows indicate the points selected as the center of the search for the next pass. The 2-D logarithmic search hits a maximum of 18 points and a minimum of 13 search points. The advantage of this technique is that it successively reduces the search area, thus reducing the computational burden. One of the disadvantages is that some points are searched more than once thus leave some space for improvement.

Moreover, it follows a greedy approach by selecting the minimum dissimilar point at each step thus posing a threat to follow a local minimum trend. Considering these facts we propose to modify the 2-D logarithmic search to overcome the local minimum problem and also eliminate the redundant computing as described in the following section. iii. proposed search technique We mainly modified the 2-D logarithmic search technique to eliminate the redundancy and local minimum problem associated with it. The search technique is elaborated next under the light of 2-D logarithmic search technique.

Our proposed search technique starts with the five points in the search window where the one is at the center and other four surrounds center point (Fig 3(a)). Unlike 2-D logarithmic search, our proposed technique selects two points $min1$ and $min2$ (Fig 3(b)) that has dissimilarity scores lower than the other three points. We then select a point as the center of search for the next pass that lies on the line in between $min1$ and $min2$. This selection

<https://assignbuster.com/a-modified-2-d-logarithmic-search-technique-for-video-coding/>

reduces the local minimum effect as it simply does not follow the minimum point.

Moreover, the five points selected in the next pass does not match with any of the previous points thus eliminates the redundancy that exists in 2-D logarithmic search. Centered at the point selected at the next pass the search continues (Fig 3(d)-Fig 3(f)). The steps of the search are portrayed in Fig 3.

- Following are some of the merits of our proposed technique:
- Successively reduces the search area with no point searched twice
- Maximum search points are 12 and minimum search points are 5 – an improvement over 2-D logarithmic search.

Results and Discussion

We have conducted a comparative analysis of Full Search, 2-D logarithmic Search and our proposed search technique as presented next. All the experiments were conducted on MPEG sequences using MATLAB. We used sequences like garden, Akiyo, Table Tennis, Car, and coastguard. Full search, 2-D logarithmic search and our proposed technique applied in these standard MPEG file and we computed the ASNR (Average Signal to Noise Ratio) and Computational load (i. e. number of search points). The results on different sequences are presented next. Akiyo Sequence: Each frame of the Akiyo sequence is of 352? 88 pixels, recorded at 25 frames per second and there are a total of 398 video frames. Fig 4 shows the reconstructed 20th frame of Akiyo sequence coded using Full search, 2D-logarithmic search and proposed search technique. In this video only face portion is moving. Search point

comparison for these three search techniques is presented in Fig 5 and ASNR is reported in Fig 6. ASNR achieved using the proposed search technique is almost equal 2D logarithmic search but at reduced number of search points (Fig 5). Number of search points remains almost similar over the different frames.

Fig 3: The different steps of our proposed 2-D logarithmic search technique.

(a) five points of search window, (b) the direction of the search in between the direction offered by the two points min1 and min2. (c) Search at step 2, (d) min1 and min2 at step 2, (e) Search points at step 3, and (f) Search ends at the blue point. (a)| (b)| (c)|

Fig 4: Reconstructed 20th frame of the Akiyo sequence using (a) Full search, (b) 2-D logarithmic search, and (c) Our proposed search technique.

Fig 5: Comparison of # of search points for Akiyo sequence. Fig 6:

Comparison of ASNR for Akiyo sequence. Table 1: ASNR value of different

search for Akiyo sequence. Car Sequence: Each frame of the Car sequence is of 320x240 pixels and recorded at 25 frames per second and there are a

total of 398 video frames. The reconstructed 20th frame of Car sequence

using the three search techniques is presented in Fig 7. In this video

sequence the car moves but background is still. Here each repeated two

times. Average no of search point is almost 10.46 for repeated frames and

11.50 for new frames. Here number of search points vary significantly

compared to Akiyo sequence. Overall the proposed technique has reduced

search points (Fig 8) although the ASNR is bit low (Fig 9). ASNR value of

some frames shown in Table 2. Fig 7: Reconstructed 20th frame of the Car

sequence using (a) Full search, (b) 2-D logarithmic search, and (c) Our

<https://assignbuster.com/a-modified-2-d-logarithmic-search-technique-for-video-coding/>

proposed search technique. Fig 8: Comparison of # of search points for Car sequence. Fig 9: Comparison of ASNR for Car sequence. Garden Sequence: Each frame of the Garden sequence is of 352? 240 pixels and recorded at 30 frames per second and there are a total of 59 video frames. Fig 10 represents the reconstructed 20th frame of this sequence coded using the three search techniques. In this video the motion is due to camera movement. Fig 11 and Fig 12 reveals that the new search technique reduces the number of search points with minor loss in ASNR. ASNR value of some frames shown in Table 3. Here Average no of search point for each frames required almost same.

In frame 20th average no of search point is 11. 6053 and ASNR is 18. 22931.

Fig 10: Reconstructed 20th frame of the Garden sequence using (a) Full search, (b) 2-D logarithmic search, and (c) Our proposed search technique.

Fig 11: Comparison of # of search points for Garden sequence. Fig 12:

Comparison of ASNR for Garden sequence. Coastguard Sequence: Each frame of the Coastguard sequence is of 320? 240 pixels and recorded at 25 frames per second and there are a total of 378 video frames. Here the boat and the camera are moving. Fig 13 represents a reconstructed frame of this sequence coded using the three search techniques. Fig 14 represents the search point required by the three techniques. Our proposed technique shows periodic nature in terms of search points.

This is due to the repetitive nature of motion in the video. Fig 15 represents a comparison of ASNR obtained using different techniques. Table 4 shown ASNR of some frames. Fig 13: Reconstructed frame of the Coastguard sequence using (a) Full search, (b) 2-D logarithmic search, and (c) Our <https://assignbuster.com/a-modified-2-d-logarithmic-search-technique-for-video-coding/>

proposed search technique. Fig 14: Comparison of # of search points for Coastguard seq. Fig 15: Comparison of ASNR for Coastguard sequence. Table tennis Sequence: Each frame of the Table tennis sequence is of 352x 240 pixels and recorded at 30 frames per second and there are a total of 9 video frames. Here ball is moving fast. The reconstructed frames, number of search points, and ASNR of the three search techniques are presented in Fig 16, Fig 17, and Fig 18. Some ASNR of Table tennis sequence shown in table 5. Fig 16: Reconstructed frame of the Table tennis sequence using (a) Full search, (b) 2-D logarithmic search, and (c) Our proposed search technique. Fig 17: Comparison of # of search points for Table tennis sequence. Overall the result of ASNR for Full Search is best in all cases but number of search point is so high. The result of ASNR for 2-D logarithmic and our proposed search is almost same but the number of search point of our proposed search is smaller than the 2-D logarithmic search and thus an improvement over the existing technique.

Fig 18: Comparison of ASNR for Table tennis sequence. Table 5: ASNR value of different search for Table tennis seq

Conclusion

In this paper we have presented a new search technique for video coding that is a modification of the existing 2-D logarithmic search. The proposed technique reduces the search time of 2-D logarithmic search by reducing the redundant search points.

Although ASNR is sacrificed to some extent it had insignificant visual impact as observed from the experimental results.

References

- Shi and H. Sun, “ Image and Video Compression for Multimedia Engineering”, Fundamentals, Algorithms and Standards, 2nd Edition.
- P. N. Tudor, “ MPEG-2 Video Compression”, IEEE J Langham Thomson Prize, Electronics and Communication Engineering journal, December 1995.
- J. R. Jain and A. K. Jain, “ Displacement Measurement and Its Application in Interframe Image Coding”, IEEE Transactions on Communications, vol. com-29, no. 12, December 1981.
- T. Koga, K. Linuma, A. Hirano, Y. Iijima, and T.
- Ishiguro, “ Motion-compensated interframe coding for video conferencing,” Proc. NTC’81, G5. 3. 1-G5. 3. 5, New Orleans, LA, Dec. 1981.
- R. Srinivasan and K. R. Rao, “ Predictive coding based on efficient motion estimation,” Proc. of ICC, 521-526, May 1984.
- D. Tzovaras, M. G. Strintzis, and H. Sahinolu, “ Evaluation of multiresolution block matching techniques for motion and disparity estimation,” Signal Process. Image Commun. , 6, 56-67, 1994.
- MPEG-4, <http://en.wikipedia.org/wiki/MPEG-4>, last accessed in December 2008.
- H. 264, <http://en.wikipedia.org/wiki/H.264>, last accessed in December 2008. *