Journal of Advances and Scholarly Researches in Allied Education
Vol. 19, Issue No. 6, December-2022, ISSN 2230-7540

Examining the Adaptive Elasticity Model in H.264 for Real-Time Video Encryption System Design

Maya Chowksey[1]*, Dr. Ravindra Tiwari[2]

[1] Research Scholar, LNCT University, Bhopal

[2] Associate Prof, LNCT University, Bhopal

Abstract- The term "video compression" refers to the process of encoding a video file in a way that results in a smaller file size. It can be difficult to record a lengthy video sequence due to the size of the resulting video files. An effective motion estimation method in the H.264 encoder has been proposed, together with an optimal dual encryption methodology. In this case, motion estimation is accomplished by a combination of a full search and a diamond search technique. The RSA-based encryption algorithm is encrypted using ECC & Rivest-Shamir-Adleman (RSA). Both efficient motion estimation & H.264 video compression are provided by the suggested approach. Compression ratio, PSNR, correlation coefficient, mean absolute error, and pixel change rate are some of the metrics used to evaluate the video compression technique's efficacy.

Keywords- video compression, Elliptic Curve Cryptography, Adaptive Elastic Motion Model H.264

- x -

INTRODUCTION

The term "multimedia" refers to a presentation that incorporates multiple media types. Multimedia information can be presented visually, audibly, textually, photographically, or through moving images & sound. Each one is a highly effective means of conveying ideas & information. The most recent update to the global video coding standard is H.264/MPEG-4 AVC. The ITU-T's Video Coding Experts Group (VCEG) & ISO/IEC's Moving Picture Experts Group (MPEG) collaborated on its creation. In order to save space & time, video compression technology eliminates unnecessary data. Spatial reduction physically compresses video by omitting or downsampling a quarter or more of each frame's original data. The most popular methods for video compression adhere to a number of standards, including those established by the Moving Picture Experts Group (MPEG), as well as H.261, H.263, & H.264. The primary goal of developing encryption systems is to furnish a trustworthy means of exchanging data. However, concerns about the safety of video chats have only just begun to be addressed. Goyal et al. (2014) point out that there is currently no requirement for encryption in existing video coding standards. First, the video to be compressed is split into individual frames in the suggested method. H.264 encoder is used to carry out the procedure. H.264 uses a tree-based, square-based allocation scheme, with a minimum component size of 4x4. The rate-distortion optimised performance has been developed by allowing the macro blocks to be partitioned into many levels of size up to 32x32, with the minimum block size being 8x8. Over a wide range of output bit rates, the elasticity-based compression method produces superior overall compression efficiency. As a result, the suggested method reduces the required number of bits for movement vectors, the required computation time, and the number of connected parts. The APSO method is used to determine the best motion vector. As a result, we now have revised optimal motion vectors. Finally, the revised optimal motion vector is encrypted using ECC. A pair of keys is used to provide security, hence this method is known as public key cryptography. There are two types of keys: public & private. Therefore, the video's original content has been maintained through the use of cryptographic methods. After the video has been encrypted, it is compressed. ECC is notable for its low resource requirements, including low memory, power, & bandwidth requirements. When it comes to encrypting motion vectors, ECC is the way to go. As can be seen in Figure 1, the general diagram of the suggested approach is given.

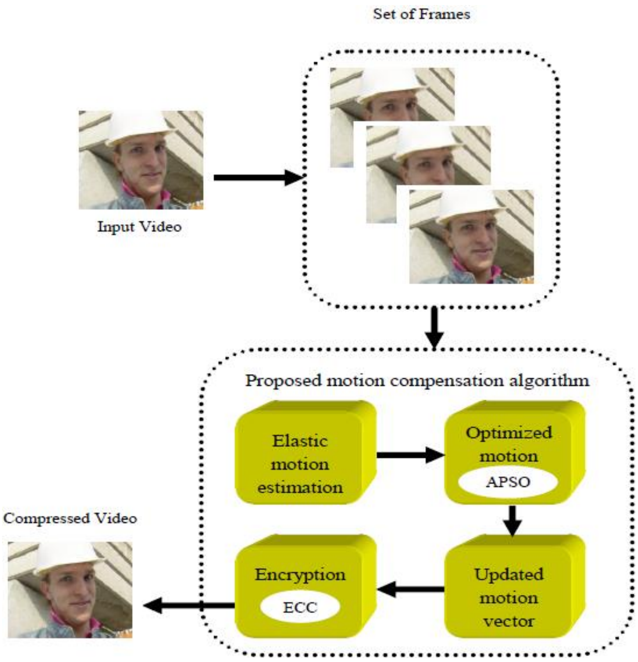

Figure 1: Video encryption system based on an adaptive elastic motion Model

ELASTIC MOTION FORM

Elastic or non-rigid image registration techniques have been extensively used in medical imaging, object tracking, image stabilization and motion analysis applications (Crum et al. 2004). To monitor small changes, it aligns the two images into a similar coordinate system. In the suggested work, this elastic motion shape for estimating the movement parameters from the video frames are utilized. Reflect on the block matching framework with two blocks (*Z* a_i ,b_i) and (*Z'* a'_i ,b'_i), which are in the form of co-ordinate transformation and is specified below,

$$a_i^{'} = a_i + \sum_{n=1}^{N/2} M_n \lambda_n (a_i, b_i) \qquad (1)$$

$$b_i^{'} = b_i + \sum_{n=N/2+1}^{N} M_n \lambda_n (a_i, b_i) \qquad (2)$$

Where, *N* = indicates the total number of motion parameters

M_n =indicates motion parameters

λ_n ()= indicates Basis functions which indicates complex mappings between *Z* and *Z'* .

The basic functions for the co-ordinate transform are,

$$\lambda_n(a_i, b_i) = \lambda_{n+N/2}(a_i, b_i)|$$

$$= \cos\left(\frac{(2a_i+1)\pi q}{2C}\right)\cos\left(\frac{(2b_i+1)\pi r}{2D}\right) \qquad (3)$$

Where,

$$n = sq + r + 1;\, q, r = 0, 1, \ldots, s-1;$$

The elastic motion is obtained by an arrangement of independent cosines, which have the ability to indicate smooth movement fields with a base number of co-productive. The larger part parameterization of the nontranslational movement is expanded by these different cosines. In this part, motion vector calculation is also used to the description.

MOTION VECTORS COMPUTATION

By the computation of motion parameters, the best block is forecasted from its reference frame. By minimizing the Sum of Squared Differences (SSD) among the blocks *Z* and *Z'*, the best-matched block is found out. The SSD is identified as,

$$F = \sum_{i=1}^{P} \left[Z\left(a_i^{'}, b_i^{'}\right) - Z\left(a_i, b_i\right) \right]^2 = \sum_{i=1}^{P} f_i^2 \qquad (4)$$

In equation 4, *P* symbolizes the number of pixels in the area, where the two images overlie and *fi* points out the difference in intensity values of a pixel *i* .

The motion parameters are revised by Δ*M* based on the first-order Taylor approximation of the SSD. The functional dependence of *Z* and ' *Z* on (ai,bi) and (a'_i,b'_i) are excluded, and the equation is specified below,

$$F^{'} = \sum_{i=1}^{P} \left[\left(Z^{'} + \frac{\partial Z^{'}}{\partial M} \Delta M \right) - Z \right]^2$$

$$= \sum_{i=1}^{P} \left[\frac{\partial Z^{'}}{\partial M} \Delta M + f_i \right]^2 \qquad (5)$$

The incomplete derivatives of *F'* regarding motion parameter updates are specified in equation,

$$\frac{\partial F^{'}}{\partial \Delta M} = 2 \sum_{i=1}^{P} \frac{\partial Z^{'}}{\partial M} \left[\frac{\partial Z^{'}}{\partial M} \Delta M + f_i \right] \qquad (6)$$

Let

$$\frac{\partial F'}{\partial \Delta M} = 2\sum_{i=1}^{P}\frac{\partial Z'}{\partial M}\left[\frac{\partial Z'}{\partial M}\Delta M + f_i\right] = 0 \qquad (7)$$

While comparing eqn. (6) as in eqn. (7), the error gets minimized after the revising of motion parameters. Thus, the accompanying condition for the reconsidered motion parameter is obtained.

$$\sum_{i=1}^{P}\left[\frac{\partial Z'}{\partial M}\right]^2 \Delta M = -\sum_{i=1}^{P}\frac{\partial Z'}{\partial M} f_i \qquad (8)$$

The above equation. (8) can be rewritten by applying matrix notation. Multiple motion parameters are comprised in transforms.

$$H\,\Delta M = w \qquad (9)$$

where, H = Hessian matrix and $H_{k,l} = \sum_{i=1}^{P}\frac{\partial Z'}{\partial M_k}\frac{\partial Z'}{\partial M_l}$ is the element of H.

w = weighted gradient vector and $b_k = -\sum_{i=1}^{P}\frac{\partial Z'}{\partial M_k} f_i$ is the element of w.

The above-mentioned elements of *H* and *w*are attained by the chain rule as in equation. (10).

$$\frac{\partial Z'}{\partial M_k} = \frac{\partial Z'}{\partial a_i'}\frac{\partial a_i'}{\partial M_k} + \frac{\partial Z'}{\partial b_i'}\frac{\partial b_i'}{\partial M_k} \qquad (10)$$

The expressions $\frac{\partial Z'}{\partial a_i'}$ and $\frac{\partial Z'}{\partial b_i'}$ 'symbolizes the horizontal and vertical gradients of ' *Z* . In elastic registration, $\frac{\partial a_i'}{\partial M_k}$ and $\frac{\partial b_i'}{\partial M_k}$ are the same to the basic functions of the warping function (see Eqn. (11)).

$$\frac{\partial a_i'}{\partial M_k} = \lambda(a_i, b_i) \text{ and } \frac{\partial b_i'}{\partial M_k} = \lambda(a_i, b_i) \qquad (11)$$

The updating of motion parameters is iterated at each stage with the iteration number *t* based on the equation (12).

$$M^{t+1} = M^t + \Delta M$$
$$= M^t + H^{-1}w \qquad (12)$$

It is necessary to re-assess *a'i, b'i, Z;, H and w* with the assistance of modified motion parameters for every one of the cycles. The motion parameters are iteratively revised till a base SSD value is acquired. From this point forward, the updates are placed into the motion parameters and utilizing novel warping function, a roughly warped version of *Z* can be figured, which is next abused in the accompanying cycle. The whole procedure is delayed, till the least variety in SSD or greatest cycle is acquired. Taken after by this the squares are separated utilizing the tree development and its profound representation is as per the following,

MOTION VECTOR OPTIMIZATION

First, the motion parameters to be encrypted are optimized to obtain the best matched block. In the recommended video compression work, APSO algorithm is employed for the process of motion parameter optimization. It is clearly detailed in the following section.

ADAPTIVE PARTICLE SWARM OPTIMIZATION (APSO)

Particle swarm optimisation is a concept for optimisation algorithms that is inspired by the cooperative nature of flocks of birds. After a collection of subjective particles is assembled, PSO's algorithm continues its search for optimal solutions by cycling through new generations. Each particle is guided through a search space centred on its own best position, with that position adjusted for its distance from the best particle in the swarm. A fitness function based on the optimisation issue is used to determine how far away from the global optimum each particle is. Two distinct forms of adaptation are used, as per PSO. The first refers to a person's personal best, whereas the second is the finest in the world. It's an algorithm for determining the particle's optimal location, or "best position," or "pbest," by comparing all possible starting points. This pbest does not include any related data on the other particles. It's the finest particle selection algorithm in the world since it learns where the best particle is located across the entire swarm. Furthermore, every molecule uses its knowledge of future events in a way that benefits itself the most. There are several steps involved in the PSO computation, all of which are repeated indefinitely until the stopping condition is met.

(1) Evaluate the fitness of each particle.
(2) Update individual & global best functions.
(3) Update velocity & position of each particle.

The APSO has two primary phases. As a first step, we execute a real-time evolutionary state estimate approach to determine whether or not the population is in a state of exploration, exploitation, convergence, or jumping out at the end of each generation. It allows for real-time optimisation of search efficiency & convergence rate by adjusting computational parameters such as inertia weight & acceleration coefficients. If the evolutionary state is determined to be a convergence state, then an elitist learning method is implemented. In order to escape the anticipated local optimum, the strategy will take action on the particle with the best global score. In this case, every p-body travels through an n-dimensional space Rn. Each p-body encompasses the following three vectors on its own. Elastic motion representation motion parameters of cosine warping functions are derived as particles in the search space.

- cp_p **-vector:** It represents where the p-th particle is at this moment in the search.
- lb_p **-vector:** It indicates the position in the search space of the p-th best solution found so far.
- cv_p **-vector:** It points out the direction for which the particle *p* will travel (the current velocity).

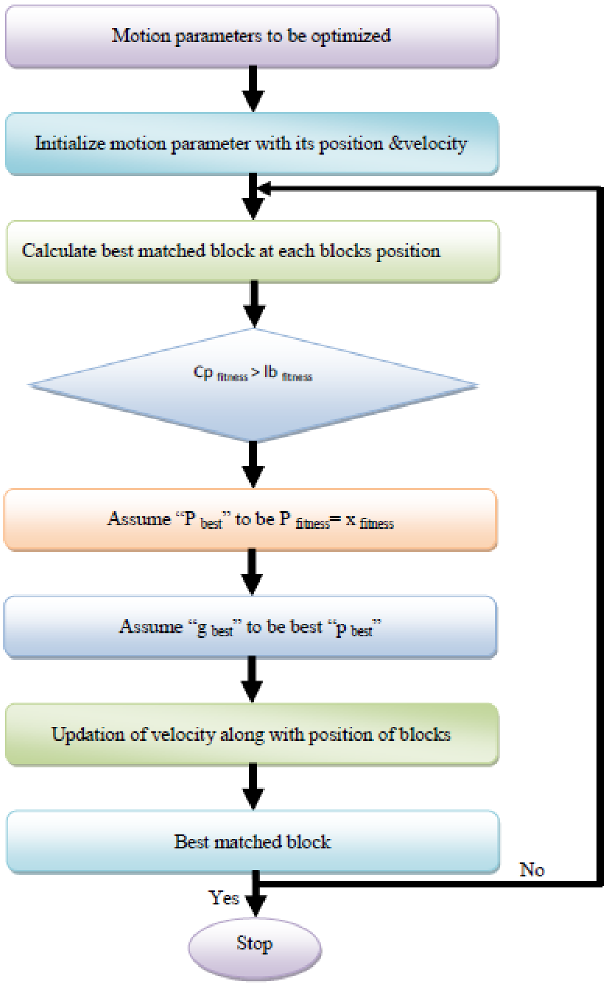
Motion parameters to be optimized
Initialize motion parameter with its position &velocity
Calculate best matched block at each blocks position
Cp fitness > lb fitness
Assume "P best" to be P fitness= x fitness
Assume "g best" to be best "p best"
Updation of velocity along with position of blocks
Best matched block
No
Yes
Stop

Figure 2 Steps for APSO

APSO Algorithm Steps

Step 1: Initialize a population of *i* particles with each particle's position cp_i and velocity *i cv* on a problem space *Rn* of dimension *n*.

Step 2: Calculate the fitness function for each particle *i* in *d* variables.

Step 3: Make comparison among the particle's fitness value, *cp fitness* and the particle's *pbest* fitness value, *lb fitness*. If the current fitness value of a particle is better than the particle's *pbest* fitness value, then set the *pbest* value into the current position in the *d*th dimension.

Step 4: Check the gbest value of the lb fitness & pbest value of all the particles. Change the particle's array index & value to reflect the gbest value if the present value, pbest, is better than the gbest value's meaning.

Step 5: Revise the velocity and position of the particles specified as in equations (13) and (14).

$$\begin{aligned} cv_{id} &= \omega \times cv_{id} + \varphi_{1a} \times r_1 \times (pbest_{id} - cp_{id}) \times pbest_{id} + \\ &\quad \varphi_{1b} \times r_2 \times (cp_{id} - pworst_{id}) \times pworst_{id} + \varphi_2 \times r_3 \times (gbest_{id} - cp_{id}) \end{aligned} \quad (13)$$

$$cp_{id} = cp_{id} + cv_{id} \quad (14)$$

where,

i - Particle.

ω - Inertia weight.

φ_{1a} - Learning rates governing the particle towards to its best position.

φ_{1b} - Learning rates governing the particle away from its worst position.

$\varphi 2$ - Learning rates governing the social components.

r_1, r_2, r_3 - Random numbers that are uniformly distributed in the range [0,1].

cv_{id} must be in the range of $[cv_{max}, cv_{min}]$, cv_{max} indicates the maximum velocity.

Step 6: Repeat step 2, until a better fitness or maximum number of iterations are met.

The optimized motion vectors attained from this APSO algorithm is next exploited to get revised motion vector in the elastic motion estimation representation. As a result, it is possible to increase the revised form of a motion vector with an efficient optimization method (Aghababa et al. 2010).

ECC BASED ENCRYPTION

ECC is a public-key encryption method based on elliptic curve algebra over finite fields (Karki & Ajit 2014). When compared to non-ECC cryptography (based on plain Galois fields), the keys needed for ECC are much lower while yet providing the same level of security. The lack of a known sub-exponential technique for solving the discrete logarithm problem on a suitably chosen elliptic curve is the primary reason for the field of ECC appeal. Compared to competing methods like DSA & RSA, ECC can employ significantly lower parameters while maintaining the same level of security. Advantages of smaller key sizes include less need for storage, less use of bandwidth, and faster computations. In the proposed approach, the updated motion parameters are encrypted using an ECC using a custom, public key. The ECC method generates active, unique, & public keys, making the updated motion parameters more secure for compacting. The primary time period and structure are described in more detail below.

KEY GENERATION BY ECC

Elliptic Curve Cryptography (ECC) is as well-known as public key cryptography, which normally has a pair of keys, a public key and a private key, and a set of actions associated with the keys to complete the cryptographic operations. The most important advantage of ECC is the small key size. The operations of elliptic curve cryptography are explained over two predetermined fields: Prime field and Binary field. For cryptographic operations, the suitable field is selected with a finitely massive number of points. The prime field operations choose a prime number *P R* and finitely large numbers of basic points are produced on the elliptic curve, such that the generated points are between 0 to *Z* .

PROPOSED MOTION COMPENSATION ALGORITHM

The recommended method, which exploits an elastic motion form and a tree construction of multi-level larger block divisions to get better compression competence for the video coding process. The best block of pixels in the current frame is forecasted from its reference frame which has formerly been passed on to the decoder by the inter-frame coding (Ayele et al. 2013). The function

(*J*mode), by which a Lagrangian cost function gets minimized is chosen and it is specified as,

$$J_{\text{mode}} = D_{\text{mode}} + \mu_{\text{mode}} \; R_{\text{mode}} \qquad (15)$$

Where,

*D*mode- The value of SSD between the original and reconstructed blocks.

µmode - Lagrangian multiplier

*R*mode - The requisite bit-rate that assists to pass on the motion vectors, transform coefficients residuals (after prediction) and macro block type information.

Consider the elastic motion vectors for the selection of a mode in the proposed method. The motion vector (*J motion*) is chosen for a particular block division with the help of Lagrangian cost function and it is specified as,

$$J_{\text{motion}} = D_{\text{motion}} + \mu_{\text{motion}} \; R_{\text{motion}} \qquad (16)$$

Where,

D motion - The value of the Sum of Absolute Differences (SAD) between the current division and the division in the reference frame denoted by the candidate motion vector.

µ *motion* - Lagrangian multiplier

R motion - The requisite bit-rate that helps to transmit the motion vectors

CONCLUSION

The proposed APSO based elastic motion model achieves a better block portioning ability which increases the accuracy of motion vectors. These motion vectors are encrypted with the encryption of the ECC algorithm. The performance of the system was examined by the assessment metrics, PSNR, CR, CC, MAE, UACI and NPCR. The metrics PSNR and CR were used to assess the compression efficiency and the results of experimentations have proved that the suggested system attains good compression ratio with better PSNR values for both the Akiyo and Foreman videos. Then the encryption of motion parameters by means of ECC algorithm is introduced in this research. This encryption procedure allows only the authorized person to decode the original video properly. This encryption capacity was as well examined by the metrics CC, NPCR, MAE, and UACI. The CC of both video datasets has the capacity to oppose the statistical attacks by offering good MAE values. The comparison was done between the existing H.264 codec with the suggested codec and obtained result shows that the suggested method provides better compression.

REFERENCES

1. Abomhara, M., Zakaria, O., Khalifa, O. O., Zaidan, A. A., & Zaidan, B. B. (2022). Enhancing selective encryption for H. 264/AVC using advanced encryption standard. arXiv preprint arXiv:2201.03391.

2. Agi, I & Gong, L 1996, ‘An empirical study of secure MPEG video transmissions’, Proceedings of the Symposium on Network and Distributed System Security, pp. 137-144

3. Ahmad Kholaif, M., Terence Todd, D., Polychronis Koutsakis and Aggelos Lazaris “Energy Efficient H.263 Video Transmission in Power Saving Wireless LAN Infrastructure”, IEEE Transactions on Multimedia, Vol. 12, No. 2, pp.142-153, 2010.

4. Ahmad Kholaif, M., Terence Todd, D., Polychronis Koutsakis and Aggelos Lazaris "Energy Efficient H.263 Video Transmission in Power Saving Wireless LAN Infrastructure", IEEE Transactions on Multimedia, Vol. 12, No. 2, pp.142-153, 2010.

5. Ahmadi, A. and Azadfar, M. M. "Implementation of fast Motion Estimation algorithms and comparison with full search method in H.264", Computer Science & Network Security Journal., Vol. 8, pp. 139-143, 2008.

6. Amit Rameshchandra Ukalkar and Narendra G. Bawane, "Analysis of Low Complexity Motion Estimation Algorithms for H.264 Video Compression Standard", IMECS,Vol. 1, March 2009.

7. Ben Atitallah, A., Arous, S., Loukil, H. and Masmoudi, N. "Hardware Implementation and Validation of the Fast Variable Block Size Motion Estimation Architecture for H.264/AVC", International Journal of Electronics and Communications, Vol. 66, pp. 701-710, 2012.

8. Bharat Bhargava, Changgui Shi and Sheng-Yih Wang, "MPEG Video Encryption Algorithms", Multimedia Tools and Applications, Vol. 24, No. 1, pp. 57-79, 2004.

9. Dalal, M., & Juneja, M. (2021). A secure and robust video steganography scheme for covert communication in H. 264/AVC. Multimedia Tools and Applications, 80(9), 14383-14407.

10. Hua-Zhen Yao & Ya-Tao Jing 2010, „The Design of Video-Conference Encryption system based on H.264", In Proceedings of IEEE International Conference on Multimedia Technology, vol. 1, pp. 191-194.

Corresponding Author

Maya Chowksey*

Research Scholar, LNCT University, Bhopal