Relocated I-Frames Detection in H.264 Double Compressed Videos Based on Genetic-CNN

Qiang Xu [*], Xinghao Jiang[*], Tanfeng Sun[*], Peisong He[*],Shilin Wang[*], Bin Li[†]
[*]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China
[†] Shenzhen Key Laboratory of Media Security, Shenzhen, China
E-mail: {xuqiangwhu, xhjiang, tfsun,gokeyhps, wsl}@sjtu.edu.cn; libin@szu.edu.cn

***Abstract*—Analyzing the appearance of Relocated I-frames is a vital step in double compression detection in Group of Pictures (GOP) non-aligned videos. In this work, a frame-wise relocated I-frames detection method in H.264 double compressed videos based on Genetic-CNN is proposed. Video clips which contain three adjacent frames are used as the input of network to separate image content from noise. A preprocessing operation is adopted by extracting the noise residual. The genetic algorithm is applied to verify the possibility of automatically designing deep network structures. The network optimization operation mainly includes CNN encoding, initialization, the construction of fitness function and genetic operations, e.g. selection, mutation and crossover. By testing on a data set composed of published YUV sequences, the results clearly demonstrate the efficacy of the proposed approach and show that the generated CNN can achieve better performance than previous method investigated.**

Keyword- Double Compressed Video, Relocated I-frames, Genetic-CNN, Noise Residual.

I. INTRODUCTION

The popular usage of video editing software makes it easy for people to edit a video. This poses a great threat to the authenticity and reliability of video information. Therefore video forensics has become an importance issue in information security. As video recompression appears in most of the tampered videos [1], video double compression detection is of vital importance in video forensics [2].

Video double compression detection can be divided into two categories according to whether the GOP structures of first and second compression are aligned or not. For GOP aligned double compression detection, the existing method include: DCT coefficient distribution based methods, first digit law based methods, Markov statistics based methods, etc. [3-5]. For GOP non-aligned video double compression, due to the shift of GOP structure, part of the original I-frames is re-encoded as P-frames (relocated I-frames). Because of the big different between intra frame prediction and inter frame prediction, the change introduces temporally distributed fingerprint. Existing approaches are mainly based on temporal analysis to realize the GOP non-aligned double compression detection. For example, in [2,6], prediction residual in each video frames are extracted to reveal double compression fingerprint. Detection methods based on block artifact are proposed by Luo et.al. [7] and He et.al. [8]. In addition, abnormal changes in macroblock types are also used in double compression detection [9]. All the methods mentioned above are based on the fixed-length GOP structure. When it comes to adaptive GOP structure, these methods become invalid. What's more, the video types targeted by these algorithms are specific, when it faces to different video types, the performance may be downgraded significantly. So in order to fully reveal the footprint of double compression and improve the flexibility of detection methods, it is necessary to adopt frame-wise identification of relocated I-frames.

On the other hand, in the past few years deep learning algorithms have made significant breakthroughs, especially in the image classification domain, the state-of-the-art algorithms on visual recognition are mostly based on the deep Convolutional Neural Network (CNN) [10], however, the existing deep neural network structures (e.g., Alexnet [11], Densenet [12]) are manually designed, not learned, and little research has been devoted to the automatic design of deep neural network. Due to the different application scenarios, we cannot directly utilize the existing network for RI-frames detection, therefore, it motivates us to develop a new type of deep learning network model which can automatically design network structures.

In this paper, Genetic-CNN is developed to detect relocated I-frames by adapting self-designed network. Specifically, the contributions of this paper are detailed in the following. (1) To prevent neutral network from learning the diverse content of video, a preprocessing operation is adopted by extracting the noise residual. (2) Genetic algorithm is applied to design the structure of deep neural networks automatically. Each network structure is represented by a fixed-length binary string. After that, several popular genetic operations are used to explore the search space efficiently. (3) As the granularity of the detection target has been refined, the detected information becomes more abundant which brings researchers more comprehensive compression historical data. The experimental results clearly demonstrate with the genetic operations that the average accuracies generally get higher from generation to generation. After successive iterations, a high-quality network structure which achieves high recognition accuracy in RI frames detection is generated.

The remainder of this paper is organized as follows. Section 2 briefly introduces the model of H.264 non-aligned double compression. Section 3 illustrates the way of using genetic algorithm to design network structures. Experiments

are shown in Section 4 and conclusion is drawn in section 5.

II. THEORETICAL MODEL OF NON-ALIGNED DOUBLE H.264 COMPRESSION AND RI-FRAMES

In a motion sequence of H.264 video, individual frames of pictures are grouped together (called a group of pictures, or GOP). The typical Group of Pictures (GOP) structure is IBBPBBP... and I-frame is used to predict the first P-frame and these two frames are also used to predict the first and the second B-frames. In video double compression scenario, assuming the GOP of double compressed video and the single compressed are not equal in length, then the relocated I-frame will occur periodically. In baseline profile for simplicity, when the second compression process adopts a different GOP structure from the first compression, part of the original I-frames are re-encoded as P-frames at the second compression (this is called Relocated I-frame or RI frame) due to the change of the GOP structure, with the example shown in Fig.1. As the encoding procedure of I-P frame is not the same as I-frame, it brings double compression traces.

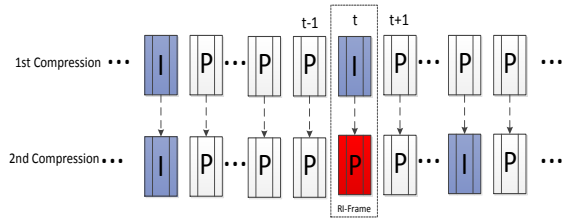

Fig. 1: Example of Video double compression in baseline profile

As shown in Fig. 1, in baseline profile, each GOP starts with an I-frames followed by P-frames. For single compression sequence, the GOP size G_1=t-1. Assuming the sequence undergoes double compression with GOP size G_2=i (i≠kt-k; k∈Z), the original I-frame at frame t is re-encoded as a P-frame (RI frame). Considering the difference between intra and inter-frame coding and the theory of motion prediction and compensation, the compression of macroblock in the t[th] and (t-1)[th] frame can be expressed as follows:

$$PM_{t-1}^{1st} = MCP\left(\widetilde{PM}_{t-2}^{1st}, PM_{t-1}\right) + R_{t-1}^{1st}, \quad (1)$$

$$IM_{t}^{1st} = ITP\left(\widetilde{IM}_{t}^{1st}, IM_{t}\right) + R_{t}^{1st} \quad (2)$$

where $PM_{t-1}^{1st}, IM_t^{1st}$ respectively denotes the macroblock of the (t-1)[th], t[th] frame in frame sequence $\{F_N\}$, IM_t, PM_{t-1} represents the macroblock in t[th],(t-1)[th] raw frame, and $\widetilde{IM}_t^{1st}$, $\widetilde{PM}_{t-2}^{1st}$ are their reference macroblock in the 1[st] compression. $MCP(\cdot)$, $ITP(\cdot)$ indicate H.264 motion prediction operator, intra prediction operator, and the prediction residuals are expressed by R_{t-1}^{1st}, R_t^{1st} respectively. For intra coding process, the prediction block is generated by matching the reconstructed blocks, and the prediction modes include Intra_4*4, Intra_16*16.

If the sequence undergoes double compression, the 2[nd] compression of frames in position (t-1) and t can be denoted as:

$$PPM_{t-1}^{2nd} = MCP\left(\widetilde{PM}_{t-2}^{2nd}, PM_{t-1}^{1st}\right) + R_{t-1}^{2nd} \quad (3)$$

$$IPM_{t}^{2nd} = MCP\left(\widetilde{PM}_{t-1}^{2nd}, IM_{t}^{1st}\right) + R_{t}^{2nd} \quad (4)$$

where PPM_{t-1}^{2nd}, IPM_t^{2nd} respectively denote the macroblock of the (t-1)[th], t[th] frame in the 2[nd] compression, and $\widetilde{PM}_{t-2}^{2nd}$, $\widetilde{PM}_{t-1}^{2nd}$ are their reference macroblock. R_{i-1}^{2nd} ,R_i^{2nd} denote the prediction residuals. Substituting the equation (1)(2) into the equation (3)(4) gives:

$$PPM_{t-1}^{2nd} = MCP\left(\widetilde{PM}_{t-2}^{2nd}, MCP\left(\widetilde{PM}_{t-2}^{1st}, PM_{t-1}\right) + R_{t-1}^{1st}\right) + R_{t-1}^{2nd} \quad (5)$$

$$IPM_{t}^{2nd} = MCP\left(\widetilde{PM}_{t-1}^{2nd}, ITP\left(\widetilde{IM}_{t}^{1st}, IM_{t}\right) + R_{t}^{1st}\right) + R_{t}^{2nd} \quad (6)$$

By applying Eqs (5)(6) to each video frames, the non-aligned double H.264 compression video frames can be expressed as:

$$F_{IP} = \{\ldots IPM_{(k-1)t}^{2nd}, IPM_{kt}^{2nd}, IPM_{(k+1)t}^{2nd} \ldots\}, k \in N^* \quad (7)$$

$$F_{PP} = \{\ldots PPM_{u-1}^{2nd}, PPM_{u}^{2nd}, PPM_{u+1}^{2nd} \ldots\}, u \in N^*, u \neq t \quad (8)$$

Where N^* represents the set of positive integers, F_{IP} and F_{PP}respectively denote RI and PP frames in H.264 double compression, and are composed of series of IPM^{2nd} and PPM^{2nd}, As the big different between intra frame prediction and inter frame prediction, Intra-frame prediction exploits spatial redundancy, i.e. correlation among pixels within one frame. By calculating prediction values through extrapolation from already coded pixels for effective delta coding, inter frame prediction tries to take advantage from temporal redundancy between neighboring frames. Besides, the t[th] frame and the (t-1)[th] frames in the first compression belong to different GOPs, which lead to weaker correlation between them. Thus the prediction residual between RI frame and PP frame differ obviously, and the difference is used as the clue to distinguish relocated I-frames from other frames. The proposed method use genetic operations to find competitive network structures which can learn multiple levels of representations of footprint left by 2[nd] compression.

III. PROPOSED METHOD

As deep neural network has been successfully applied in many areas, various fixed network structures have been designed manually. However, on different scientific issues, the applicability of fixed networks always varies widely. Therefore it becomes increasingly meaningful to develop a new type of deep learning network model which can automatically design network structures. In this section, Genetic-CNN is proposed for the RI-frame detection. Firstly, the main framework is introduced, then several contents include input data preprocessing. Network structure design is analyzed in detail. In the part of network structure design,

genetic algorithm is adopted to find the optimal structure, each individual structure is first represented by a fixed-length binary string, then several genetic operations are defined by combining the double compression scenario, including selection, crossover and mutation.

A. Overall Framework

The proposed method follows a pipeline depicted in Fig. 2, and the main steps are as follows:

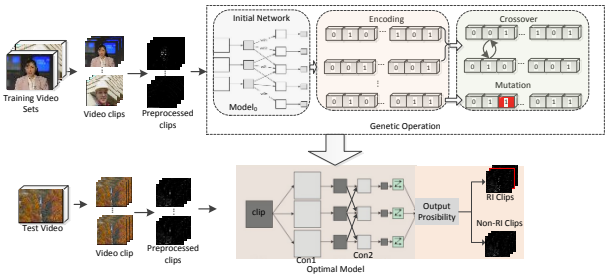

Fig. 2: Framework of the proposed RI frame detection method

Step1.Video preprocessing

In the first step, videos are cut into video clips to form datasets, each video clip is labeled first, and then preprocessing operation is adopted by extract the noise residual.

Step2.Network Initialization

Individual networks are initialized in this step, the initialization of individual networks in Genetic-CNN should follow several restrictions like: each network has a limited number of layers, and each layer is designed as a set of predefined building blocks such as convolution and pooling, each model can be represented by a binary string.

Step3. Optimal model generation

Initialized networks are fed with processed training clips, and the fitness of individual is determined by its recognition accuracy. After several genetic operation like selection, crossover and mutation, the genetic process comes to an end when the generation exceeds the maximum number of generations and the training process outputs the optimal learned model which achieves the highest accuracy and contains all CNN parameters.

Step4.Data testing using optimal model

In the test section, test data are fed to the optimal CNN model and the network yields the classification probability of the input data. This probability is converted to the estimated label (RI clips or NON-RI clips). Clearly, if pre-processing is applied during training, it must be applied during testing.

B. Data Preprocessing

Decompressed frames are stacked in every 3 continuous frames to form video clips $D_i = \{D_i(t), t = 1,2,3\}$ to take advantage of the temporal correlation in a video. Each video clip can be regarded as a 4D tensor with RGB channels, spatial channel and temporal channel. In order to expose video double compression traces, input video clips $\mathrm{D_i}$ are first converted to grayscale $\mathrm{DGray_i}$, with a denoising pre-processing operator adapted afterwards. The process can be expressed as follows:

$$\check{\mathrm{D}}_i(t) = \left|DGray_i(t) - MF\left(DGray_i(t)\right)\right|, t = 1,2,3 \qquad (9)$$

$MF(\cdot)$ indicates the median filter operator, which relies on a spatially adaptive statistical model for the Discrete Wavelet Transform. This algorithm is widely used in the field of forensics for its great capability of separating image content from noise [13].

C. Network Structure Design

In this paper, a novel network design method is proposed. We try to explore the possibility of automatically learning the structure of deep neural networks. To design a neural network, several restrictions should take into consideration, in which the network has a limited number of layers, and each layer is designed as a set of predefined building blocks such as convolution and pooling. However, it is important to note that even under these limitations, the total number of possible network structures grows exponentially with the number of layers. Therefore, considering the computation complexity, it is impractical to enumerate all the candidates and find the optimal one. This problem is formulated as optimization in a large search space, and the genetic algorithm is applied to traversing the space efficiently. The flow chart of the proposed Genetic-CNN is depicted in Fig.3.

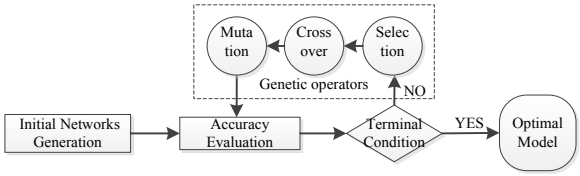

Fig. 3: Flow chart of the proposed Genetic- CNN

As Fig.3 shows, the main processes include binary encoding, initialization, selection, crossover, mutation and evaluation.

a) Binary Encoding

In our work, network structure is represented by binary string, and the binary string is directly manipulated as genotype in genetic algorithm. By analyzing the state-of-art network structure, we note that each network can be divided into several stages, with each stage separated by pooling layer. Inspired by this strategy, we define a binary encoding area in each stage. The nodes within each stage indicate convolutional operation and they are so strictly arranged that connections from a higher-numbered node to a lower-numbered node (e.g. connection from C2 to C1) are not allowed. After convolution, batch normalization and ReLU are followed. Assuming that the network consists of N stages, the n-th stage, n = 1, 2, ⋯, N, contains $K_n$nodes, denoted by $C_n k_i$, k_i = *1*, *2*, ⋯, K_n, then we use $L = (K_n - 1) * K_n/2$ bits to encode the net. Each bit represents whether the current node is connected with its

previous nodes. If there is a connection from node C1 and C2, the first bit of the binary string is 1. It is important to note that we do not encode the fully-connected part of a network and the convolutional operations in one stage share the same parameters (number of filters, stride, width, height, etc.). On this basis, it's easily to know that the number of possible network structures is 2^L. In the case of large L value, the computation complexity of exhaustive search algorithm is unbearable.

As shown in Fig.4, stages are separated by pooling layers. The number of nodes in stage1 and stage2 are 3 and 4 respectively, then we encode them in 3 and 6 bits and the binary code of stage1 is 100 as there's only one connection from $C_1$1 to $C_1$2. Similarly, the code of stage2 is 000101.

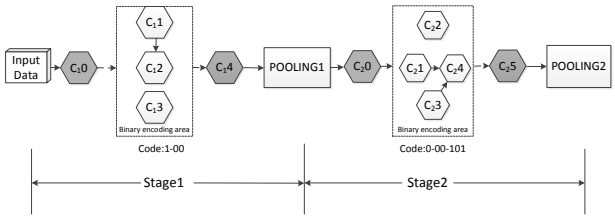

Fig.4 Example of a two-stage network structure binary encoding

b) Genetic Operation

Initialization: The initial population M_0 consists P randomized models $\{Model_0(i)\}(1 \le i \le P)$ which form the searching space of possible network. All models are encoded to a L length binary string and each bit is independently sampled from a Bernoulli distribution: *b~B(0.5).*

Selection: Selection is an indispensable procedure to generate new individuals. In our work, Roulette wheel selection is performed to select potential network, individuals with higher fitness level are more likely to be selected. The steps of Roulette wheel selection in the *t–th* generation are as follows:

Step1: Obtain the fitness value $f_t(i)$ of individual in a P size population $\{Model_t(i)\}(1 \le i \le P)$.

Step2: Calculate the probability of individual $\{Model_t(k)\}$ being selected is $p(k)$:

$$p(k) = f_t(k) / \sum_{i=1}^{P} f_t(i) \; ; k = 1,2, \dots P \qquad (10)$$

Step3:Suppose $q(0)=0, q(k) = p(1) + p(2) + \cdots p(k); \; k = 1,2, \dots P$

Step4: Generate a random number r $(0 \le r < 1)$, if $(k\text{-}1) < r < q(k)$, then individual $\{Model_t(k)\}$ is selected.

Crossover and Mutation: In genetic algorithm, crossover and mutation operators are used to vary the programming of a chromosome or chromosomes from one generation to the next. For crossover operation in network design, the basic unit is a stage, individuals in the population are paired randomly and some genes in two individuals are exchanged simultaneously at the crossover point with probability P_C. Different from crossover, mutation process alters one or more genes in an individual $\{Model_t(i)\}$ with a small probability P_m. For a binary string, if the genome bit is 1, it is changed to 0 after mutation process and vice versa.

Evaluation: Fitness function is the evaluation function to guide the search in genetic algorithm. In the issue of genetic algorithm-based RI frame detection, each individual network $\{Model_t(i)\}$ is evaluated by calculate the detection accuracy, then the fitness function is denoted as ACC(), which indicates network with higher detection accuracy are more likely to be chosen .

c) Process of Network Structure Design Based on Genetic Algorithm

Step1: Determine the number of generations (G), size of population (P), crossover rate (P_C) and mutation rate (P_m) according to the actual situation.

Step2: Generate the initial population $\{Model_0(i)\}(1 \le i \le P)$, and calculate each accuracy.

Step3: After the fitness value of individual network and the chosen probabilities are calculated, select P individuals from $\{Model_t(i)\}$ $(0 \le t \le G, 1 \le i \le P)$ by roulette wheel selection to form a new population.

Step5: For each pair $\{Model_t(2i-1), Model_t(2i-1)\}$ $(0 \le t \le G, 1 \le i \le P/2)$, perform crossover operator on the selected two individuals with crossover rate P_C.

Step6: Perform mutation operator on each $\{Model_t(i)\}$ $(0 \le t \le G, 1 \le i \le P)$ with mutation rate P_m.

Step7: If the generation does not exceed G, return Step3 and continue the evolution, otherwise terminate the iteration and find the individual with highest fitness, the individual is regard as the optimal network.

IV. EXPERIMENTS

In this section, experiments are carried out to evaluate the performance of Genetic-CNN, the proposed method is compared with method based on prediction residual in [2] and Alexnet [11]. The performance of the optimization operation and different input frame number are analyzed.

A. Datasets Setup

In order to verify the validity of the proposed method, datasets are built by using 19 published YUV sequences which were obtained from the online video databases: http://media.xiph.org/video/derf/. These sequences contain diverse contents in order to train a generalized network. YUV sequences involved in the training phase includes: akiyo, paris, bridge-far, container, deadline, flower, silent, hall, highway, intros, mobile, mother-daughter, news, and sequences used for the testing includes: bridge-close, sign-irene, galleon, students, washdc, waterfall.

For each sequence, only the first 250 frames are used and the parameters setting for 1st Encoding and 2nd Encoding are listed in Table1. All these sequences are encoded in H.264 format and in baseline profile with constant bitrate(CBR) mode. The x264 is used as the H.264 codec and the GOP size is 5 in single compressed videos and 10 in double compressed videos. Finally, 304 double compressed videos and 76 single compressed videos are generated.

Table 1 Dataset and parameters setting for the experiment

YUV Sequence involved		
Training phase: *akiyo, paris, bridge−far, container, deadline, flower, silent, hall, highway, intros, mobile, mother−daughter, news;* Testing phase: *bridge−close, sign−irene, galleon, students, washdc, waterfall.*		
Parameters	**1st Encoding**	**2nd Encoding**
Bitrate	{500, 600, 700, 800}kbps	{500, 600, 700, 800}kbps
GOP size	5	10
Sequence Number	76 (=19× 4)	304 (=19× 4× 4)

As introduced in previous section, in order to take advantage of the temporal information in videos, the input samples in our work are video clips instead of single frames or videos. That is, before preprocessing, every three consecutive frames in each video are stacked as video clips. To build the datasets, samples are generated as follow: For each single compressed video belongs to the same YUV, 200 video clips stacked as PPP are extracted; For each double compressed video belongs to the same YUV, 400 video clips whose middle frame is RI frame and 200 video clips whose middle frame is PP frame are extracted. Video clips containing RI frame are defined as positive samples and the rest are negative samples. Finally, we can obtain 10400 =13*400+13*400 training samples and 4800=6*400+6*400 testing samples in total.

B. Parameter Configuration

In the Genetic-CNN architecture design, we set stage number N=2, the nodes number in stage1 K_1 is 3, and K_2=5 in stage2. The number of possible network structures can be easily figured out, that is $2^L = 2^{13}$=8192. The number is so great that the computation complexity is unbearable to the system. In the experiment, the size of kernels in the first convolutional layer in each stage and average-pooling layer are 5×5 with spatial stride 1, 2 respectively, and the kernel size of other convolutional layers is set to 1 × 1. We apply 30 training epochs with learning rate 0.001, followed by 20 epochs with learning rate 0.0001, momentum was set to 0.9, Mini-batch gradient descent is applied to train the individual models and batch size is set to 13. For genetic operation, the size of initial population P=20, with maximum generations G 50. The crossover rate P_C and the mutation rate P_m were 0.5 and 0.3 respectively, the relatively high value of P_C $P_m$facilitate the generation of new structures.

C. Performance Comparison

In this section, the performance of the proposed method is evaluated by comparing with Chen et.al.'s method in [2] and the well know neural network AlexNet, TPR(True positive rate),TNR(True negative rate) and ACC(Accuracy) are used to evaluate the detection performance, their formulas are shown below:

$$TPR = \frac{TP}{(TP + FN)} * 100\% \quad (11)$$

$$TNR = \frac{TN}{(TN + FP)} * 100\% \quad (12)$$

$$ACC = \frac{TP + TN}{P + N} * 100\% \quad (13)$$

TP,TN,FP,FN,P,N respectively denotes the number of correctly identified, correctly rejected, incorrectly identified, incorrectly rejected, positive samples and negative samples.

a) Comparison with Prediction Residual Method

In this paper, the proposed method is compared with Chen et al.'s method [2], as their work is a prior one. Chen et al.'s method [2] propose a prediction residual based feature named PRED feature, and the theory in [2] indicate the PRED feature has the ability to identify the difference between RI frames and PP frames. However, different from our proposed frame-wise detection method, the detection method in [2] is based on a single complete video unit and PRED feature is extracted to detect double H.264 compression with nonaligned GOP structures. By measuring the difference of PRED characteristics between adjacent frames in the complete video the result achieves AUC of 0.9306 which is quite high. Considering there are few studies on frame-wise RI-frames detection and PRED feature is feasible to differentiate the RI frames from the PP frames, we thus compared with Chen et al.'s method.

For fair comparison, the prediction residual based experiments are redone by using the datasets and parameters mentioned above. The average prediction residual for each non-overlapping 4x4 block in the second frames of input video clips are extracted from the bit stream directly during the decoding process, and each residual value is rounded to its nearest integer. video clips containing RI-frames are defined as positive samples and video clips where the middle frame is single compressed P-frames or PP frames are set as negative samples. A binary classifier is used for the classification procession. TNR, TPR, ACC are calculated and results are shown in Table 2.

Table 2 Results comparison with prediction residual method（%）

Evaluation criteria	TNR	TPR	ACC
Prediction Residual Method[2]	91.71	85.83	88.77
Proposed	**93.67**	**91.42**	**92.54**

Data in Table 2 show that the true positive rate and true negative rate of Genetic-CNN reach 91.42% and 93.67% respectively, while the result of Chen et al.'s method achieve 85.83% and 91.71%. The data suggested that considering the accuracy, proposed Genetic-CNN is 3.77% higher than Chen et.al.'s method. The reason for the worse performance of the

prediction residual method is the difference of video content and the varied motion strength in different videos. Also, the distribution of the prediction residuals of a complete video has a strong regularity. However, this regularity becomes weaker in a set of video clips cut from multiple videos. It is hard for prediction residual based method to find a proper threshold value to discriminate RI-frames and other frames. On the other hand, when the bit rate of the second compression is lower than that of the first compression, the effect of the PRED feature is suppressed. A lower bit rate means an increase in the quantization step size and higher information loss degree. The prediction residual of each frame of the video is greatly reduced, which suppresses the effect of the PRED feature. In contrast, the data-driven based Genetic-CNN has a strong feature learning ability, and the preprocessed feature data has a more substantial representation of the original data, which will greatly facilitate the classification of relocated I-frames and other frames. In summary, Genetic-CNN performs better than prediction residual method in RI-frames detection.

b) Comparison with AlexNet

AlexNet is a well know neural network which achieved a top-5 error of 15.3%, more than 10.8 percentage points ahead of the runner up in the ImageNet Large Scale Visual Recognition Challenge in 2012. The network contains only eight layers: the first five are convolutional layers, and the last three are fully connected layers. To match the input of the network, samples are resized to 224*224, and other parameters e.g. kernel size, filters number, remains the same. In order to compare the performance fairly, the input data of AlexNet must undergo the same preprocessing operation as in proposed framework, that is to say, noise residual and temporal information are also extracted in AlexNet.

Data in Table 3 show the true positive rate and true negative rate of AlexNet just achieve 75.6% and 87.62%. And the ACC of Genetic-CNN is 10.93% higher than AlexNet.

Table 3 Results comparison with AlexNet（%）

Evaluation criteria	TNR	TPR	ACC
AlexNet[11]	87.62	75.60	81. 61
Proposed	**93.67**	**91.42**	**92.54**

The results indicate that on the issue of RI-frames detection, the detection results of artificially designed AlexNet are not as expected, and the proposed Genetic-CNN yield considerably high performance, one reason for this difference could be the using of different pooling operation in neural network, i.e., max-pooling in AlexNet will cause a certain degree of information loss, while average-pooling in Genetic-CNN can synthesize all residual information and achieve better generalization ability. In addition, the use of smaller convolution kernels in deeper layers prevents overfitting of the network and improves the ability to learn global statistics. Actually, the AlexNet can be regarded as a local solution in network space, and the proposed Genetic-CNN can find the optimal network structure which outperforms AlexNet in RI-frames detection by the unique genetic operation and the network.

D. Performance of the Optimization Operation

To verify the feasibility of genetic algorithm in designing neural network, experiments have been carried out. We firstly generate the initial population $\{Model_0(i)\}(1 \le i \le 20)$, each accuracies of all individuals are calculated by using preprocessed input samples, then roulette wheel selection is adapted to form a new generation. The size of population remains unchanged at 20. After that, crossover operator and mutation operator are performed with rate 0.5 and 0.3 respectively. Finally, the evolution procession terminates at the 50^{th} generation and output the optimal network. In order to observe the optimization performance of genetic algorithm in different periods, the detection accuracy (including the max/min/avg/med accuracy) of local optimum network is recorded by every 2 generation in the preceding 10 generations and every 10 in the following 40 generations. The results are as shown in Fig.5:

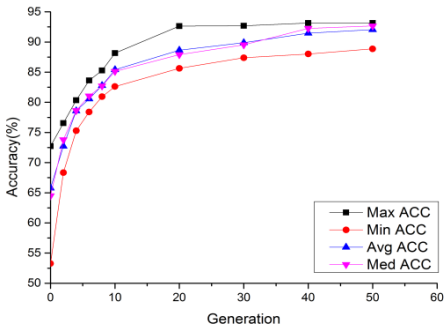

Fig. 5: Detection Accuracy over all individuals with respect to the generation

Fig.5 schematically reveals the relationship between different type of detection accuracy and the generation. It appears that with the increase of generation, the average detection accuracy is improved. Optimization results are evolved toward the optimal solutions, in addition, from the figure, we can find that the accuracy increment is more obvious in the first 10 generations, after 50 generation. The maximum accuracy reached 93.15% and the average accuracy reached 92.06%. The results are consistent with earlier findings showing that the genetic algorithm can easily find solutions which performance better than parent solutions in the early stages. With the genetic operations, a competitive network structures can be found which achieve high recognition accuracy. It is worth noting that although the maximum recognition rate of the individual is not improved from the 40^{th} generation to 50^{th} generation, the average and medium accuracies generally get higher from generation to generation. This result validates that the genetic processes ultimately result in the next generation population of chromosomes which is different from the initial generation. The genetic algorithm improves the overall quality of the

individuals. After 50 generations, the recognition error rate of the best individual drops from 27.28% to 6.85% and the binary code of the optimal individual is 1010011110001.

E. Performance Analysis for Different Input Frame Number

In this section, the influence of different input frame number is researched for RI-frames detection in H.264 double compressed video. Results are shown in Table 5.

Table5 Comparison for different input frame number（%）

Evaluation criteria	TNR	TPR	ACC
One Frame as Input	48.26	51.30	49.78
Three adjacent Frames as Input	**93.67**	**91.42**	**92.54**

As the results in Table5 show, the TNR and TPR achieved 93.67% and 91.42% respectively if three adjacent frames are stacked as video clip, and the average detection rate achieved 92.54%. Compared with input only containing one frame, network with three adjacent frames as input gets a significantly higher detection value. The average detection rate of network containing only one frame as input is 49.78%, which is lower than random guessing. These findings support the hypothesis that the temporal inconsistency among RI-frames and its adjacent frames is an important clue for RI-frames detection in H.264 double compressed videos.

V. CONCLUSIONS

In this paper, a relocated-I frame detection algorithm in H.264 GOP non-aligned videos is proposed. The overall framework contains a preprocessing module, which aims to prevent neutral network from learning the diverse content of video by extract the noise residual. Then, data are fed into Genetic-CNN in a stack of continuous frames instead of individual frames. Different from common manual designed network, Genetic-CNN applied genetic algorithm to design neural network in an automatic way. The optimization operation mainly includes binary encoding, initialization and several genetic operations. Compared with the method based on prediction residual and the famous neutral network alexnet, the proposed method can achieve superior results.

Our future work includes transforming the basis idea to video-wise analysis of video inter-frame tamper detection. and working on other types of video manipulation detection.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (No. 61572320, 61572321). The corresponding author is Dr. Tanfeng Sun.

REFERENCES

[1] S. Milani, M. Fontani, P. Bestagini, M. Barni, A. Piva, M. Tagliasacchi, S. Tubaro, “An overview on video forensics,” *APSIPA Trans. Signal Inf. Process.* vol.1, 2012, pp.1229-1233

[2] S. Chen, T. F. Sun, X. H. Jiang, P. S. He, Y. Q. Shi. “Detecting double h.264 compression based on analyzing prediction residual distribution,” *International Workshop on Digital Watermarking. Springer*,vol.10082, 2016, pp.61-74.

[3]W. Wang, H. Farid, “Exposing digital forgeries in video by detecting double quantization,” in: *Proceedings of the 11th ACM Workshop on Multimedia and Security, ACM,* 2009, pp.39-48.

[4] Y. Su, J. Xu, “Detection of double-compression in MPEG-2 videos,” in: *2010 2nd International Workshop on Intelligent Systems and Applications (ISA), IEEE*, vol32(32),2010, pp.1-4.

[5] J. Xu, Y. Su, Q. Liu, “Detection of double MPEG-2 compression based on distributions of dct coefficients,” *Int. J. Pattern Recogn. Artif. Intell*. vol.27(01), 2013, pp.35-40.

[6]W. Wang, H. Farid, “Exposing digital forgeries in video by detecting double MPEG compression,” in: *Proceedings of the 8th Workshop on Multimedia and Security, ACM*, 2007, pp.35-42.

[7] W. Luo, M. Wu, J. Huang, “MPEG recompression detection based on block artifacts,” *International Society for Optics and Photonics,* vol.12(02),2008, pp.681-689.

[8] P. S. He, T. F. Sun, X. H. Jiang, “Double compression detection in MPEG-4 videos based on block artifact measurement with variation of prediction footprint,” *International Conference on Intelligent Computing,* vol.9927,2015, pp.787-793.

[9] D.Vazquez-Padin, M. Fontani, T. Bianchi, “Detection of video double encoding with GOP size estimation,” *Information Forensics and Security (WIFS), 2012 IEEE International Workshop on. IEEE,* 2012, pp.151-156.

[10]L. Xie, A. Yuille, “Genetic CNN,” *IEEE International Conference on Computer Vision. IEEE Computer Society*, 2017, pp.1388-1397.

[11]A. Krizhevsky, I. Sutskever, G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in: *Advances in neural information processing systems*, vol.60(02),2012,pp. 1097-1105.

[12]G. Huang, Z. Liu, L. V. D. Maaten, K. Q. Weinberger, “Densely Connected Convolutional Networks,” in: *Computer Vision And Pattern Recognition,* 2016, pp.43-49.

[13] M. Barni, L. Bondi, N. Bonettini, P. Bestagini, A. Costanzo, “Aligned and non-aligned double JPEG detection using convolutional neural networks,” *Journal of Visual Communication & Image Representation*, vol.49(04),2017, pp.153-163.