Article

Time Delay Optimization of Compressing Shipborne Radar Digital Video Based on Deep Learning

Hongrui Lu , Yingjun Zhang * and Zhuolin Wang

Navigation College, Dalian Maritime University, Dalian 116026, China; lu450619968@dlmu.edu.cn (H.L.); zl_w98@dlmu.edu.cn (Z.W.)

* Correspondence: zhangyj@dlmu.edu.cn

Abstract: The High Efficiency Video Coding Standard (HEVC) is one of the most advanced coding schemes at present, and its excellent coding performance is highly suitable for application in the navigation field with limited bandwidth. In recent years, the development of emerging technologies such as screen sharing and remote control has promoted the process of realizing the virtual driving of unmanned ships. In order to improve the transmission and coding efficiency during screen sharing, HEVC proposes a new extension scheme for screen content coding (HEVC-SCC), which is based on the original coding framework. SCC has improved the performance of compressing computer graphics content and video by adding new coding tools, but the complexity of the algorithm has also increased. At present, there is no delay in the compression optimization method designed for radar digital video in the field of navigation. Therefore, our paper starts from the perspective of increasing the speed of encoded radar video, and takes reducing the computational complexity of the rate distortion cost (RD-cost) as the goal of optimization. By analyzing the characteristics of shipborne radar digital video, a fast encoding algorithm for shipborne radar digital video based on deep learning is proposed. Firstly, a coding tree unit (CTU) division depth interval dataset of shipborne radar images was established. Secondly, in order to avoid erroneously skipping of the intra block copy (IBC)/palette mode (PLT) in the coding unit (CU) division search process, we designed a method to divide the depth interval by predicting the CTU in advance and limiting the CU rate distortion cost to be outside the traversal calculation depth interval, which effectively reduced the compression time. The effect of radar transmission and display shows that, within the acceptable range of Bjøntegaard Delta Bit Rate (BD-BR) and Bjøntegaard Delta Peak Signal to Noise Rate (BD-PSNR) attenuation, the algorithm proposed in this paper reduces the coding time by about 39.84%, on average, compared to SCM8.7.

Citation: Lu, H.; Zhang, Y.; Wang, Z. Time Delay Optimization of Compressing Shipborne Radar Digital Video Based on Deep Learning. *J. Mar. Sci. Eng.* **2021**, *9*, 1279. <https://doi.org/10.3390/jmse9111279>

Academic Editor: Marco Cococcioni

Received: 11 October 2021

Accepted: 13 November 2021

Published: 17 November 2021

Keywords: shipborne radar digital video; unmanned ship; HEVC; SCC; deep learning; delay of compression

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In recent years, various automation technologies centered on unmanned ships have developed rapidly. The observation of shipborne radar digital video is the most intuitive and efficient way to understand the navigation state of the target ship. Optimizing the transmission and storage of radar data has become a popular topic in the field of unmanned ship research. Data compression technology, as the basis of ship-to-shore intelligent information interaction, is also a key technology of unmanned ship remote control. The International Convention for the Safety of Life at Sea (SOLAS) requires that ships of 500 gross tonnage (GT) and above should be equipped with radar devices, and ships of 10,000 GT and above should be fitted with two independent radar devices; furthermore, the radar devices should be able to operate at 9 GHz frequency [?]. The main data to be used in the future, such as driving video and radar digital images of unmanned ships, need to be compressed and transmitted to intelligent equipment for processing

and storing, under the condition of limited maritime bandwidth. Thus, it is necessary to improve the efficiency of compression of the algorithm to reduce delay. As shown in Figure ??, for the purpose of realizing remote virtual driving of an unmanned ship, the data generated by various devices on ships need to be shared in real time with other devices and shore-based consoles to monitor the current performance status of the ship, and simulate and create a real ship bridge environment. As the most important piece of equipment to assist ships in safe navigation and collision avoidance, shipborne radar is mainly responsible for indicating the position between the ship and objects, strengthening the awareness in real-time traffic situations, and assisting the ship in judging encounters. The International Maritime Organization (IMO) requires that when a ship is sailing along a coast or in a port, the shipborne radar digital video needs to show clear shorelines and the location of other fixed dangerous objects. In collision avoidance operations, shipborne radar needs to provide the pilot with the detection of nearby hazardous targets and the position information of obstacles relative to the ship. This is obtained according to a report to help the ship make decisions and verify the effectiveness of collision avoidance action [?]. IMO also stipulates that shipborne radar must be able to detect small buoys, including fixed navigation beacons and hazards, for safe navigation and collision avoidance [?]. Therefore, the speed and code rate of compressing shipborne radar digital video not only determines the transmission efficiency, but also affects the safety of the ship under remote control.

Figure 1. Schematic diagram of remote transmission of radar video.

Since the 1980s, in order to solve the problem of large-scale intercommunication and transmission of video images, the International Telecommunication Union for Telecommunication Standardization Sector (ITU-T) and the International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) jointly issued the H.26x series of video coding standards [?]. In 2013, the Joint Collaborative Team on Video Coding (JCT-VC) issued the high efficiency video coding standard (H.265/HEVC), which is mainly applied to ultra-high-definition digital video, and released the expansion scheme of screen content coding (HEVC-SCC) [?]. In 2016, Intra Block Copy (IBC) [?], Palette Mode (PLT) [?], Adaptive Color Transform (ACT) [?], Adaptive Motion Vector Resolution (AMVR) [?], etc., were added. The above technology has further improved the performance of encoding digital screen video. Although the introduction of new technologies is dedicated to improving coding quality and coding efficiency, it has also led to increasing computational complexity. In the optimal dividing process of determining the Coding Tree Unit (CTU), HEVC introduces a new segmentation structure, which increases its code rate

saving by about 50% compared with Advanced Video Coding (AVC). However, the coding complexity simultaneously increases by 253% [?]. The SCC coding scheme is proposed based on the HEVC standard, but is different from the Coding Unit (CU) division of HEVC. According to the texture complexity of the Quantization Parameter (QP) and CTU, HEVC recursively divides the CU downward and calculates the RD-cost to obtain the best division method. However, due to the introduction of IBC and PLT modes in SCC, the RD-cost of various modes of CU are calculated by traversal. Even in CUs with high texture complexity, the downward division can be stopped. This method of sacrificing computing power to improve coding performance is obviously not suitable for an environment with limited computing resources.

In recent years, numerous research has been conducted using machine learning methods to reduce the complexity of HEVC coding. In order to solve the problem that the quad-tree-based recursive CU division search is too complicated in the CTU dividing process, the method in [?] established a large-scale coding test general sequence CU dividing dataset. The study proposed a Hierarchical CU Partition Map (HCPM), and designed an end-to-end ETH-CNN intra-frame division decision-making network and an ETH-CNN-LSTM inter-frame division decision-making network. By predicting the intra-frame and inter-frame CU division results, deep learning methods were used instead of search decision for CU division in HEVC. Literature [?] proposed a fast decision-making algorithm for CU based on deep learning, which uses the neural network structure in the Inception module to decide the CTU division in advance, in order to achieve stable transmission in an environment with high network fluctuations. Method [?] regarded coding efficiency and bit error rate as the main factors affecting CU division, and proposed a Low-Complexity Mode Switching-based Error Resilient Encoding (LC-MSRE) based on a neural network. The approach used SMIF-intra and STMIF-inter to compare the intra-frame and inter-frame CU distortion to determine the best way to divide the CTU. With the process of the PU angle prediction link in the intra-prediction process, HEVC increases the PU prediction modes to 35, including 33 angle prediction modes and Planar and DC prediction modes. Method [? ? ?] proposed a CNN-based PU angle prediction model (AP-CNN), which replaced the traditional PU angle prediction mode in the HEVC lossless coding mode. Ref. [?] was based on the original AP-CNN in [?], improving the loss function and training strategy to enhance the prediction accuracy. Ref. [?] proposed an optimized architecture LAP-CNN based on AP-CNN, which reduced the complexity of the model and increased the coding speed. In [?], the author designed a CNN optimization model based on LeNet-5, and adopted an early termination of the CU division strategy, which reduced the computational complexity of the rate distortion cost (RD-cost) in intra-prediction. In order to reduce the coding complexity of SCC, the method in [?] adopted a network structure (Deep-SCC) for predicting the CU division mode, in which Deep-SCC-I is responsible for quickly predicting the original sample value, and Deep-SCC-II is responsible for reducing the optimal computational complexity of the prediction mode.

In addition to the above-mentioned methods of reducing the computational complexity of HEVC with the CU division decision as the core, some researchers have considered the relationship between the depth of CTU division and the computational complexity of HEVC. The method in [?] sets two CTU depth ranges. According to the texture complexity of the currently coded CTU, it determines the depth range in which the CTU recursively calculates the RD-cost and determines the best segmentation result of the CTU. The method in [?] distinguished between the CU division methods in HEVC and SCC. By counting the coding results of a large number of screen content videos, four kinds of CTU depth ranges based on screen content coding were set, and CTU depth prediction based on deep learning was proposed. The model improves the coding efficiency by skipping and terminating the calculation of the RD-cost outside the prediction depth range. The article [?] transformed the decision-making problem of the division mode into a classification problem, and proposed a CU fast classification algorithm based on a convolutional neural network, which can perform fast coding by learning image texture, shape, and other features. The article [?

] proposed two fast CU dividing algorithms based on machine learning; one uses an online Support Vector Machine (SVM) for dividing CU, and the other is based on Deep-CNN to predict the CU dividing algorithm. The horizontal comparison of these algorithms proves that the deep learning method is more suitable for reducing the computational complexity of HEVC.

Although these schemes for optimizing coding complexity can reduce the computational complexity and coding delay in the CTU division to a certain extent, they are not completely applicable to the ship domain. Due to the nature of the ship's work and the particularity of the geographical environment, it is difficult to frequently maintain and upgrade the equipment and instruments, and the main video data, such as shipborne radar and thermal imaging, are quite different from the traditional screen content video. To date, an accelerated coding scheme for compressing shipborne radar digital video in maritime field has not been developed. Therefore, optimizing the coding complexity by analyzing the characteristics of the digital video of the shipborne radar is a critical step to realizing the remote control of the ship.

1. Based on the characteristics and application scenarios of shipborne radar digital video images, this study developed an algorithm to rapidly encode shipborne radar digital video by combining the algorithm with the deep learning method to predict the CTU in advance without affecting the quality of the encoding.
2. By establishing a sufficient CTU division depth dataset based on shipborne radar digital video, the proposed CTU division depth interval prediction model is trained, which improves the efficiency of predicting the CTU division depth interval in the process of encoding shipborne radar digital video.
3. When encoding shipborne radar digital video, the encoder can predict the search depth interval of the CU in advance by calling the trained prediction model, thus effectively reducing the computational redundancy caused by the full RD-cost of CU traversal, while reducing the computational complexity of encoding shipborne radar digital video.

This paper is divided into four parts, and the specific organization is as follows. Section ?? introduces the characteristics of shipborne radar digital video in detail and presents the design of a CTU depth interval model based on radar images. Section ?? presents the optimization of the CTU division process in the SCC encoding process, and proposes a time delay optimization algorithm of compressing shipborne radar digital video based on deep learning. According to previous related research work, a CNN-based fast convolutional network structure is proposed to accurately predict the search depth interval of CU. Section ?? introduces the experimental process and analyzes the experimental results. Finally, we summarize the work in Section ??.

2. Radar Digital Video Coding Depth Division Algorithm Based on Deep Learning

The SCC encoding process is similar to the HEVC standard. The difference is that HEVC is mainly used to encode natural content video captured by cameras, whereas SCC is chiefly for screen content video generated by computers. Shipborne radar digital video is a form of mixed screen content video in the field of computer vision. Therefore, the SCC coding scheme was taken as the basic platform for compressing the shipborne radar digital video to optimize the coding process and reduce the delay in compression.

2.1. Characteristics of Radar Images

In the field of navigation, radar images are formed by radar transmitters transmitting radio waves to obstructions, other ships, coastlines, and other major targets, and the receivers receiving radio echoes. The Maritime Safety Committee (MSC) in MSC.192 (79) stipulates that the radar image, target tracking information, positional data derived from the ship's own position (EPFS), and geo-referenced data (geospatial coordinates and attributes) should be integrated and displayed on the shipboard radar screen. The integration and display of the ship's Automatic Identification System (AIS) information

should be provided to complement radar. Meanwhile, selected part of the electronic navigation chart and other vector chart information can be provided in accordance with needs of the ship so that the driver can monitor and navigate the position, resulting a safer navigation of the ship [?], thus improving the pilot's recognition of the target in the radar image and the perception of the ship's surrounding environment [?]. The size of the object mark in the radar image is inversely proportional to the size of the set range. Under normal navigation, the larger the range, the smaller the volume of the object. On the shipboard radar screen, the display color of the background area usually occupies most of the display area of the entire radar image. In addition, in order to facilitate the pilot's view of the contents of the radar screen under different lighting conditions, the brightness value and color of the set background are usually different from the intensity of the target echo display. In contrast to the content refresh of ordinary video, the content refresh of radar digital video changes with the rotation period of antenna scanning. In addition, the image must be updated continuously and smoothly, so that the content of adjacent frames does not produce obvious variation. A large number of repetitive patterns or signs are also an obvious feature in radar videos, and the edges of the patterns are sharper.

In the display of target information, the shipborne radar digital video shall be displayed in accordance with the specified performance standards, using the relevant symbols specified by SN/Circ.243 [?]. For small-size ships of less than 500 GT, the diameter of the radar operation display area should not be less than 180 mm, and the total area of the display area is to be no less than 195 mm × 195 mm. For medium and large ships of above 500 GT and below 10,000 GT, the diameter of the radar operation display area is required to be no less than 250 mm, and the total area of the display area is to be no less than 270 mm × 270 mm. For large-size ships of 10,000 GT and above, the diameter of the radar operation display area shall not be less than 320 mm, and the total area of the display area is required to be no less than 340 mm × 340 mm. The resolution of the shipborne radar image is also quite different from that of ordinary video image. The common video image resolutions are 4:3, 16:9, and other mainstream display ratios, whereas radar video image resolutions are mostly 1:1 and 5:4, and other non-mainstream display ratios. The screen size and resolution of shipborne radar are very important parameters to determine the encoding speed, which decides the total number of CTUs in the radar image. Generally speaking, the higher the resolution and the larger the screen size, the more CTUs in the image, and the slower the encoding speed.

For the transmission or storage of radar video images, lower bit rate and higher image quality are important indicators for evaluating the performance of compression algorithms. For the distortion of small objects, in particular, encoding radar images is more stringent than encoding common screen images. Before the SCC coding scheme was proposed, AVC and HEVC standards were used to encode and transmit real-time radar video in the maritime field. The continuous updating of the coding scheme enables the radar video with a lower bit rate while maintaining the same definition. SCC has further improved the performance of coded radar video by introducing new IBC and PLT modes. Although SCC was proposed on the basis of HEVC, there are obvious differences in CTU division when using SCC and HEVC to encode the same video image, and SCC coding is of higher complexity. This is shown in Figure ??a, which is an example of CTU division of a frame of a radar image on the Y component using HEVC encoding. We mark the ordinary CU division mode with a dark blue solid line. As shown in Figure ??b, which is an example of CTU division on the Y component using SCC to encode the same frame of the radar image, in order to intuitively judge the difference between the two coding schemes in the CU segmentation results, we mark the case of CU using IBC mode with a red solid line, and the case of CU using PLT mode with a yellow solid line. It can be seen that the two coding schemes differ greatly in the selection of CTU division mode when encoding shipborne radar digital video.

Figure 2. CTU division of radar digital video: (a) the CTU partition results of HEVC; (b) the CTU partition results of SCC.

As can be seen from Figure ??, when coding a CTU with a more complex texture, in order to ensure the quality of the details of the image, HEVC will divide the CTU into smaller CUs, whereas SCC will divide the CTU into larger CUs using IBC and PLT modes. This is also one of the reasons why SCC reduces the bit rate by more than HEVC. The intra-coding processes of HEVC and SCC are shown in ????, in which it can be noted that the SCC reduces the code rate by increasing the RD-cost of the IBC/PLT mode in the process of recursively calculating the RD-cost of the CU. Nonetheless, this further division method also results in more calculations. At present, HEVC fast coding algorithms based on deep learning have mainly been proposed for the texture features of the CTU itself, ignoring the relationship with the adjacent coded CUs. Therefore, a fast CTU division algorithm suitable for HEVC cannot accurately match the CTU segmentation result of SCC, which not only increases the code rate, but also affects the encoding effect.

Figure 3. Intra-CTU partition of HEVC.

Figure 4. Intra-CTU partition of SCC.

2.2. Modeling of CTU Division Depth Interval

The quad-tree dividing rules formulated by HEVC are also applicable to SCC. HEVC specifies a CTU of 64×64 sized as the root node, and each leaf node of the CTU is called a CU. Each CTU is divided into four non-overlapping CUs; in turn, the size of the CU may be

64×64 , 32×32 , 16×16 , or 8×8 . Each encoded CTU will have its own division method. A CTU may contain several CUs of different sizes, CUs of the same size, or only contain one CU; that is, the CTU is not divided (CU size = CTU size). The size of the set QP value also determines the degree of detail of the CU division. The smaller the QP value, the less the image distortion, and the more precise the CU division. The larger the QP value, the more the details of the image are lost, and the rougher the CU division.

At present, SCC mainly relies on traversing all 64×64 to 8×8 CU divisions—a total of 85 CUs—and calculating the RD-cost to determine the best CTU division method. However, this exhaustive search mode generates a large number of redundant calculations. In order to reduce the coding delay caused by an exhaustive search, a CTU depth interval model based on radar images was established, as shown in Figure ???. In the CTU depth interval model designed by us, the CU of 64×64 is defined as the CU of depth 0, 32×32 is defined as the CU of depth 1, 16×16 is defined as the CU of depth 2, and 8×8 is defined as the CU of depth 3. Thus, there are four types of depths in the CTU.

Figure 5. CTU depth interval model.

Each CTU can be regarded as a random permutation of CUs of different sizes, with a total of 85 permutations. In this paper, we use depth (x, y, z) to represent the depth intervals of different combination modes, where x, y, z are the depths of CUs of different sizes, and if the two are equal, then the same depth is taken. For example, Depth (0) indicates that the current CTU only contains CUs with a depth of 0, and Depth (3) indicates that the current CTU only includes CUs with a depth of 3. Depth (1,2) represents that the current CTU contains CUs with the depth of 1,2. Depth (1,2,3) indicates the current CTU has a maximum depth of 3 and the minimum depth is 1, including CUs with depths of 1, 2, and 3. The CTU dividing results of many coded radar videos show that the CTU dividing depth directly affects the selection of IBC and PLT modes by CU. Therefore, in order to train the neural network model more conveniently and avoid erroneously skipping the IBC/PLT mode when encoding the radar image, thus causing the loss of coding performance, we classify the depth interval of CTU, as shown in Equations (1) and (2). In addition, we designed the corresponding CU search depth interval D_x , as shown in Table ??, with five

total labels. For example, when D_0 is the maximum value of prediction, it indicates that the prediction depth of the current CTU is 0, and the downward division is no longer continued. If the value of D_2 is maximum, it means that there is a CU with a depth of 2 in the current CTU, and CU with a depth of 1 may exist, so the search depth interval of the CUs is limited to $[1, 2]$. If D_3 is the maximum value of prediction, it shows that the prediction depth interval of the current CTU is $[1, 3]$, and the search mode is limited to CUs with depths of 1, 2, and 3. By using the neural network to predict the current CTU, the probability that the CU belongs to these five search depth intervals is obtained. According to the predicted CU search depth interval, the RD-cost outside the calculation interval is skipped or terminated, which limits the exhaustive search times during CTU division and reduces the computational redundancy caused by the traversal RD-cost to some extent.

$$\begin{cases} Range_{min} = Depth_{min}(x, y, z) \\ Range_{max} = Depth_{max}(x, y, z) \end{cases} \quad (1)$$

$$Range_{Dep} = [Range_{min}, Range_{max}] \quad (2)$$

where $x, y, z \in (0, 1, 2, 3)$, $Range_{Dep}$ represents the division depth interval of the current CTU, $Depth_{min}(x, y, z)$ denotes the minimum depth of the interval, and $Depth_{max}(x, y, z)$ indicates the maximum depth of the interval. If the two are equal, the same value is taken.

Table 1. Depth range classification of CTU.

The Depth Range of CTU ($Range_{Dep}$)	Category Label (D_x)	Depth Interval of CU Search
$Range_{Dep} = [0]$	D_0	None
$Range_{Dep} = [1]$	D_1	$[0, 1]$
$Range_{Dep} = [2], Range_{Dep} = [1, 2]$	D_2	$[1, 2]$
$Range_{Dep} = [1, 3]$	D_3	$[1, 2, 3]$
$Range_{Dep} = [3], Range_{Dep} = [2, 3]$	D_4	$[2, 3]$

2.3. Data Set Establishment

In order to train the neural network model with sufficient samples, we collected a large number of high-definition shipborne radar digital video sequences having 30 fps at 1280×1024 (5:4) and 1024×1024 (1:1) resolutions. We regard the Y component of each complete CTU and its corresponding label as sample data, and establish a CTU division depth interval dataset based on the digital video of the shipborne radar. In order to speed up the production of datasets, we processed the shipborne radar digital video into 298 video sequences with a length of 10 s under different sailing environments. The specific parameter information is shown in Table ?? . SCC official test platform SCM8.7 was used to encode the processed shipborne radar digital video sequence. In the configuration of the encoder, QP is set to the common values of 22, 27, 32, and 37 for the encoding test, and the default configuration of all-intra coding is adopted to save and record all Y components of CTU and division depth information, and the corresponding search depth interval label is set.

After the dataset is constructed, three seed datasets are obtained by random sampling. The number of samples of the sub-datasets are shown in Table ??, in which the training set accounts for 80% of the total dataset, the verification set accounts for 10%, and the test set accounts for 10%. We consider that the content of the radar video is refreshed with the rotation period of antenna scanning, the period is long, and the inter-frame redundancy is large. Therefore, in order to avoid overfitting in the training stage due to the high repetitiveness of the data, frame level interval sampling (FLIS) was adopted in the data processing stage to enhance the difference between data. In the shipborne radar digital video with a frame of 30 fps used in this paper, the rotation period of the radar antenna scanning is about 2.5 s, and the scan line rotation in the echo area is about

144° per second. In the two adjacent radar images, the angle of the scan line differs by about 4.8°. Accordingly, during the process of making the training dataset, we chose to encode 10 frames of shipborne radar digital video and save the CTU division depth data of 1 frame of the radar image, and set the corresponding label D_x for training the network structure. For example, when the resolution of the shipborne radar digital video is 1280×1024 , each radar image will contain a total of 320 CTU to be encoded, which means that 9600 sample data will be generated each second. In addition, the CTU of the adjacent frame in the same situation may not change before the radar scan line is refreshed. Therefore, the data similarity of the adjacent radar image is fully decreased using FLIS.

Table 2. Shipborne radar digital video parameters.

Radar Range	Background	Number of Sequences	Frames	Samples in Frame	Raw Samples Amount	Samples Amount after FLIS
1.5 NM (1:1)	Anchor	12	3600	16×16	921,600	92,160
	Under way	16	4800		1,228,800	122,880
	Port	16	4800		1,228,800	122,880
1.5 NM (5:4)	Anchor	8	2400	20×16	768,000	76,800
	Under way	12	3600		1,152,000	115,200
	Port	16	4800		1,536,000	153,600
3 NM (1:1)	Anchor	20	6000	16×16	1,536,000	153,600
	Under way	12	3600		921,600	92,160
	Port	12	3600		921,600	92,160
3 NM (5:4)	Anchor	20	6000	20×16	1,920,000	192,000
	Under way	10	3000		960,000	96,000
	Port	18	5400		1,728,000	172,800
6 NM (1:1)	Anchor	14	4200	16×16	1,075,200	107,520
	Under way	8	2400		614,400	61,440
	Port	12	3600		921,600	92,160
6 NM (5:4)	Anchor	12	3600	20×16	1,152,000	115,200
	Under way	12	3600		1,152,000	115,200
	Port	8	2400		768,000	76,800
12 NM (1:1)	Anchor	4	1200	16×16	307,200	30,720
	Under way	16	4800		1,228,800	122,880
	Port	10	3000		768,000	76,800
12 NM (5:4)	Anchor	4	1200	20×16	384,000	38,400
	Under way	16	4800		1,536,000	153,600
	Port	10	3000		960,000	96,000

Table 3. The number of samples in the dataset.

Radar Range	Train Set	Validation Set	Test Set	Total
1.5 NM	546,816	68,352	68,352	683,520
3 NM	638,976	79,872	79,872	798,720
6 NM	454,656	56,832	56,832	568,320
12 NM	414,720	51,840	51,840	518,400
Total	2,055,168	256,896	256,896	2,568,960

3. Time Delay Optimization of Compressing Shipborne Radar Digital Video Based on Deep Learning

3.1. Flow Chart of Compressing Delay Optimization Algorithm

The time delay optimization algorithm of compressing shipborne radar digital video based on deep learning proposed in this paper is shown in Figure ??.

Figure 6. Encoding delay optimization algorithm.

Considering the particularity of radar video applications, in order to take into account the speed and efficiency of encoding radar video, we did not choose to directly predict the CTU division result of the radar image when designing the algorithm. Instead, we divided the depth interval by predicting the CTU, allowing the encoder to independently decide the best division result within the set CU search depth interval, which reduced the error between the predicted division result and the actual division result. The specific process of the time delay optimization algorithm of compressing radar digital video proposed is as follows:

1. Input the shipborne radar digital video signal to be compressed into the encoder. If the radar video format is RGB, it will be preprocessed to the YUV format. Before officially starting encoding, the encoder will divide a frame of the radar image into N CTUs to be encoded.
2. The Y component of the CTU is input into the trained prediction model—the network structure of the model is shown in Figure 7 (we introduce the structure and training process in detail in Section ??)—and the pixel matrix is normalized to accelerate the convergence rate. The output of the model is the probability of D_x .
3. Compare the probability of each classification label output by the model to determine the search depth interval of CU in the current CTU. If the probability of D_0 is the maximum, the current CTU is not divided directly; that is, the depth is 0, and the RD-cost is no longer traversed downward. If not, determine whether D_1 is the maximum. If the probability of D_1 is maximum, the CU search depth interval is set to $[0, 1]$, otherwise, the minimum value of the CU search depth interval is set to 1.

4. Determine whether D_2 is the maximum value; if so, set the maximum value of the CU search depth interval to 2, and calculate the RD-cost of each CU mode within the set CU search depth interval [1, 2] to determine the final CTU division result. Otherwise, judge whether D_3 is the maximum value; if so, the maximum value of the CU search depth interval is set to 3, and the set CU search depth interval [1, 3] is traversed to calculate the RD-cost of each mode of the CU to determine the final CTU division result. If not, the minimum value of the CU search depth interval is set to 2.
5. Check whether D_4 is the maximum value. If D_4 is the maximum value, set the maximum value of the CU search depth interval to 3, and calculate the RD-cost of each CU mode within the set CU search depth interval [2, 3] to determine the final CTU division result. If not, initialize the CU search depth interval to [0, 3].
6. In the process of predicting the CTU division depth interval, there is a very low probability that the prediction will be inaccurate; that is, if the probability is low or zero, it is not likely to happen. In order to avoid the influence of this situation on the encoded image, the CU search depth interval is initialized to [0, 3] after the fifth step, and the RD-cost of all CUs in each mode is calculated through traversal to determine the final CTU division result.
7. In the process of compressing shipborne radar digital video, by limiting the CU search depth interval, a large amount of computational redundancy due to the cost of ergodic rate distortion is reduced. For example, when the prediction result is D_2 , the encoder stops traversing the RD-cost with a CU depth of 3 and skips the calculation of RD-cost with a CU depth of 0. This speeds up the current CTU encoding rate and reduces the compressed delay to a certain extent.

3.2. CTU Partition Deep Prediction Model Based on Deep Learning

Based on the characteristics of shipborne radar digital video, we proposed a CTU division depth interval prediction model on account of deep learning, and optimize the complexity of the neural network model according to previous related work [? ?]. The improved fast convolutional network structure is shown in Figure ??.

Figure 7. Design of the deep learning model.

The structure of a neural network is composed of two preprocessing layers, two convolution layers, one pooling layer, one merging layer, and two fully connected layers. Firstly, the encoder divides a frame of the radar image into N CTUs having a size of 64×64 . The preprocessing layer is responsible for extracting the Y component of the CTU that contains the main visual information, which is used as the input information of the prediction model. Then, the Y component of the CTU is globally and locally averaged

and normalized on three branches in order to speed up the gradient descent to obtain the optimal solution.

Convolution layer: In order to obtain more features in the CTU with low complexity, we refer to the method in Inception Net [?]. By using different sizes of convolution kernels to obtain different sizes of receptive fields in CTU, the feature level is higher after operation. Therefore, we use convolution kernels with different sizes of 8×8 , 4×4 , and 2×2 in the first convolution layer to extract the low-level features of CTU division depth on three branches. In the second convolution layer, the convolution kernel with a 4×4 is used to simultaneously extract higher-level features on three branches, and finally 64 feature maps are obtained on three branches. In order to comply with the non-overlapping rule of CU quad-tree partition specified by HEVC, we set the stride of all convolution operations as the kernel edge length for non-overlapping convolution operation.

Pooling layer: After the convolution operation of the first layer, the maximum pooling process with stride of 4 and size of 4×4 is carried out on three branches to simplify the feature at the same time.

Merging layer: The features after merging are composed of all the feature maps after the operation of the second convolution layer. All features after convolution operation are merged and converted into vectors, which is more helpful to obtain global and local features.

Full connection layer: The merged feature vectors pass through three full connection layers successively, including two hidden layers and one output layer. The final output result is the probability of the corresponding label $D_{0,1,2,3,4}$ of the current CTU, and the probability maximum D_x is the search depth interval of CU in the current CTU. According to a large number of experiments, we found that QP is the main factor affecting the bit rate and CTU depth division in the process of compressing the shipborne radar digital video. Therefore, we add the selection of QP value as a main external feature to the feature vector of the first and second fully connection layer to improve the adaptability of the model under different QP values and the accuracy of predicting the depth interval of CTU.

In the training and test phases, in order to prevent the problem of gradient disappearance, all convolution layers and the first and second full connection layers are activated with the leaked linear unit (Leaky-ReLU), as shown in Equation (3). Leaky-ReLU, a variant of the modified linear unit (ReLU), solves the problem that neurons in the negative interval of the ReLU function do not learn by introducing a fixed slope α . The output layer is activated by the normalized exponential function (Softmax), as shown in Equation (4), where x_i represents the output value of the i th node, and n indicates the total number of output nodes. In the training stage, the cross-entropy function is used to calculate the predicted loss. The expression of the loss function of the i th training sample in batch processing is shown in Equation (5), where y'_i denotes the actual value of the first label and y_i represents the i th output value of Softmax.

$$\text{Leaky-ReLU}(x) = \begin{cases} x & x > 0 \\ \alpha x & x \leq 0 \end{cases} \quad (3)$$

$$\text{Softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}} \quad (4)$$

$$H_{y'}(y) = - \sum_i y'_i \log y_i \quad (5)$$

4. Analysis of Experimental Result

In order to verify the effectiveness of the proposed algorithm in reducing the time delay of compressing shipborne radar digital video, we embedded the trained model into the SCC dedicated test platform SCM8.7, and compiled and tested it on the 64-bit Ubuntu 20.04.2 LTS operating system. The hardware configuration of the test environment we built was an Intel Core i7-8700 CPU@3.20 GHz, the running memory was 16 G, and the neural

network was constructed using Tensorflow. For the encoder configuration, the All Intra (AI) default configuration was used for coding. The test set contained shipborne radar digital video of four different ranges in various scenarios. The number of samples is shown in Table ???. The coding savings ratio ΔT , BD-BR, and BD-PSNR in the VCED-M33 proposal are taken as the evaluation indexes of algorithm effectiveness. The smaller the value of BD-BR, the lower the bit rate after compression. The greater the value of BD-PSNR, the higher the quality after compression. The smaller the value of ΔT , the less time it takes to encode. Equations (6) and (7) list the calculation methods of PSNR and ΔT .

$$PSNR = 10 \cdot \lg \frac{M \cdot N \cdot A^2}{\sum_{n=0}^{N-1} \sum_{m=0}^{M-1} [f(m, n) - \hat{f}(m, n)]^2} \quad (6)$$

$$\Delta T = \frac{1}{n} \sum_{QP} \frac{T_{SCM8.7} - T_{prop}}{T_{SCM8.7}} \times 100\% \quad (7)$$

where $f(m, n)$ and $\hat{f}(m, n)$ are the images before and after encoding, M and N are the width and height of the image, and A is the maximum gray value of the image. $T_{SCM8.7}$ represents the coding time of the encoder SCM8.7, T_{prop} represents the coding time of the method in this paper, and n is the total coding times. The experimental results are shown in Table ??? below.

Table 4. Performance of the algorithm in different environments.

Radar Range	Background	BD-BR/%	BD-PSNR/dB	$\Delta T/\%$				
				QP = 22	QP = 27	QP = 32	QP = 37	Average
1.5 NM	Anchor	2.92	−0.21	−33.22	−36.43	−40.89	−42.69	−38.31
	Under way	1.22	−0.15	−42.56	−46.00	−49.35	−51.36	−47.32
	Port	2.95	−0.22	−36.14	−38.55	−41.36	−42.30	−39.59
3 NM	Anchor	3.00	−0.23	−25.37	−26.56	−29.98	−33.09	−28.75
	Under way	1.36	−0.16	−41.36	−42.33	−44.10	−46.85	−43.66
	Port	2.96	−0.22	−27.12	−29.88	−32.09	−33.84	−30.73
6 NM	Anchor	2.84	−0.21	−36.98	−39.76	−43.45	−44.52	−41.18
	Under way	1.57	−0.17	−33.65	−34.10	−35.87	−37.54	−35.29
	Port	2.51	−0.19	−38.01	−40.57	−43.81	−44.30	−41.67
12 NM	Anchor	2.30	−0.19	−40.29	−43.84	−46.98	−48.12	−44.81
	Under way	1.33	−0.16	−39.45	−41.28	−43.01	−45.47	−42.30
	Port	2.21	−0.18	−40.20	−42.31	−47.44	−48.13	−44.52
Average		2.26	−0.19	−36.20	−38.47	−41.53	−43.18	−39.84

The results in Table ?? show that a time delay optimization algorithm of compressing shipborne radar digital video based on deep learning generally reduces the encoding delay by about 39.84%, whereas the BD-BR only increases by 2.26%. Although the value of BD-PSNR decreased about 0.19 dB, it does not affect the subjective perception. In mooring and port waters where the echo area is more complex, there are more CTUs with a high depth. For example, in the range between 1.5 and 3 NM, the increase in target objects affects the encoding speed to a certain extent. In a navigation area with a simple echo area, if the range is 6 or 12 NM, because there are fewer targets, most of which are flat background areas, the encoding speed is higher.

In order to intuitively reflect the performance difference between the proposed algorithm and the traditional method in compressing the shipborne radar digital video, under the constraints of different QP values (22,27,32,37), we visualize the 3 NM range radar digital video results encoding port waters, as shown in Figure ???. The two approaches are represented with different colors. It can be seen that there is no significant difference between the R-D curve of the proposed method and that of SCM8.7, which indicates that the proposed algorithm performs well in terms of coding quality.

Figure 8. R-D curves contrast.

The final result of the experiment shows that our method changes with the alteration of the range and the complexity of the echo region. When the range is larger and the content is smoother, the algorithm consumes less time and has better performance. In addition, the performance of the algorithm is directly related to the QP value. The higher the QP value, the lower the number of high-depth CTUs, the fewer times of traversing the CU RD-cost, and the better the performance of the algorithm.

5. Discussion

The transmission and storage of shipborne radar digital video is of great significance to realize remote unmanned ships. In a marine environment with limited network bandwidth and storage resources, the interaction of ship–shore information and the use of efficient compression algorithms to process video data are the key methods to address information transmission and storage, in addition to the future development trend. Therefore, the compression delay has become one of the important factors affecting real-time communication between ships and the shore. We consider that intra-frame coding is the main reason for the compression delay. Considering the instability of maritime channel transmission, this study used a convolutional neural network to convert the CTU division depth problem of the coded radar image into a classification problem, which reduces the intra-frame computational complexity of coded radar images and optimizes the transmission delay. Next, on the basis of the existing research, we will examine the optimization of the computational complexity and control the time delay caused by inter-frame coding, and further reduce the transmission bit rate.

The ongoing research on unmanned ships is resulting in a gradual increase in various sensing devices on ships. The fusion of multi-sensor data promotes the development of ship automation, makes the display content of radar images richer, and also provides more reference information for remote driving. Therefore, the shipborne radar video dataset we collect will continue to be updated and improved in the future.

6. Conclusions

In this study, a time delay optimization algorithm for compressing the shipborne radar digital video using deep learning was proposed. First, the difference between shipborne radar digital video and traditional screen video was analyzed and a large number of shipborne radar digital video sequences was collected. We encoded the shipborne radar digital video using SCM8.7, the official test platform of SCC, and established a CTU depth interval data set. The dataset contains CTU division depth interval information for encoding shipborne radar digital video under different QP values. Second, an optimization algorithm flow for the reducing the shipborne radar digital video time delay based on deep learning was introduced. Finally, we built a neural network model and used a large

number of samples to train the model. In the process of compressing shipborne radar digital video, the encoder predicts the division depth interval of CTUs in advance by invoking the trained model, which reduces a lot of unnecessary RD-cost calculation, and accelerates the speed of encoding radar video. The actual radar transmission and display results show that, compared with the official SCC test platform SCM8.7, our optimization method results in a significant increase in encoding speed, saving about 39.84% of the encoding time overall, whereas BD-BR and BD-PSNR result in a reduction of only 2.26% and 0.19 dB.

Author Contributions: Methodology, H.L. and Y.Z.; investigation, H.L. and Z.W.; data curation, H.L. and Z.W.; writing—original draft preparation, H.L.; writing—review and editing, Y.Z. and Z.W.; supervision, Y.Z.; project administration, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the National Key R&D Program of China (No. 2018YFB1601502), the Liao Ning Revitalization Talents Program (No. XLYC1902071), the Fundamental Research Funds for the Central Universities (No. 3132019313).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AMVR	Adaptive Motion Vector Resolution
AVC	Advanced Video Coding
ACT	Adaptive Colour Transform
AIS	Automatic Identification System
BD-BR	Bjontegaard Delta Bit Rate
BD-PSNR	Bjontegaard Delta Peak Signal to Noise Rate
CTU	Coding Tree Unit
CU	Coding Unit
CNN	Convolutional Neural Network
EPFS	Electronic Position Fixing System
FLIS	Frame Level Interval Sampling
GT	Gross Tonnage
GOP	Group of Pictures
HEVC	High Efficiency Video Coding
ITU-T	International Telecommunication Union for Telecommunication Standardization Sector
ISO	International Organization for Standardization
IBC	Intra Block Copy
IEC	International Electro technical Commission
IMO	International Maritime Organization
JCT-VC	Joint Collaborative Team on Video Coding
MSC	Maritime Safety Committee
NM	Nautical Miles
OOW	Officer On Watch
PSNR	Peak Signal to Noise Ratio
PLT	Palette Mode
QP	Quantization Parameter
RGB	Red Green Blue
RD-cost	Rate Distortion Cost
SCC	Screen Content Coding
SOLAS	Safety of Life at Sea

References

1. IMO Solas. International Convention for the Safety of Life at Sea (SOLAS). In *International Maritime Organization*; IMO: London, UK, 2003.
2. Zhou, Z.; Zhang, Y.; Wang, S. A Coordination System between Decision Making and Controlling for Autonomous Collision Avoidance of Large Intelligent Ships. *J. Mar. Sci. Eng.* **2021**, *9*, 1202. [\[CrossRef\]](#)
3. IMO Resolution MSC.192(79). Adoption of the Revised Performance Standards for Radar Equipment. In *International Maritime Organization*; IMO: London, UK, 2004.
4. Topiwala, P.N.; Sullivan, G.; Joch, A.; Kossentini, F. Overview and Performance Evaluation of the ITU-T Draft H.26L Video Coding Standard. In *Applications of Digital Image Processing XXIV*; International Society for Optics and Photonics: San Diego, CA, USA, 2001; Volume 4472, pp. 290–306. [\[CrossRef\]](#)
5. Xu, J.; Joshi, R.; Cohen, R.A. Overview of the Emerging HEVC Screen Content Coding Extension. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *26*, 50–62. [\[CrossRef\]](#)
6. Budagavi, M.; Kwon, D.K. Intra Motion Compensation and Entropy Coding Improvements for HEVC Screen Content Coding. In *2013 Picture Coding Symposium, PCS 2013—Proceedings*; IEEE: Manhattan, NY, USA, 2013; pp. 365–368. [\[CrossRef\]](#)
7. Pu, W.; Karczewicz, M.; Joshi, R.; Seregin, V.; Zou, F.; Sole, J.; Sun, Y.C.; der Chuang, T.; Lai, P.; Liu, S.; et al. Palette Mode Coding in HEVC Screen Content Coding Extension. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2016**, *6*, 420–432. [\[CrossRef\]](#)
8. Badry, E.; Shalaby, A.; Sayed, M.S. Fast Algorithm with Palette Mode Skipping and Splitting Early Termination for HEVC Screen Content Coding. In *Proceedings of the 2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)*, Dallas, TX, USA, 4–7 August 2019; pp. 606–609. [\[CrossRef\]](#)
9. Zhu, W.; Zhang, K.; An, J.; Huang, H.; Sun, Y.C.; Huang, Y.W.; Lei, S. Inter-Palette Coding in Screen Content Coding. *IEEE Trans. Broadcasting* **2017**, *63*, 673–679. [\[CrossRef\]](#)
10. Zhang, L.; Xiu, X.; Chen, J.; Marta, K.; He, Y.; Ye, Y.; Xu, J.; Sole, J.; Kim, W.S. Adaptive Color-Space Transform in HEVC Screen Content Coding. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2016**, *6*, 446–459. [\[CrossRef\]](#)
11. Peng, W.H.; Walls, F.G.; Cohen, R.A.; Xu, J.; Ostermann, J.; MacInnis, A.; Lin, T. Overview of Screen Content Video Coding: Technologies, Standards, and Beyond. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **2016**, *6*, 393–408. [\[CrossRef\]](#)
12. Li, B.; Xu, J. A Fast Algorithm for Adaptive Motion Compensation Precision in Screen Content Coding. In *Data Compression Conference Proceedings*; IEEE: Manhattan, NY, USA, 2015; pp. 243–252. [\[CrossRef\]](#)
13. Correa, G.; Assuncao, P.; Agostini, L.; da Silva Cruz, L.A. Performance and Computational Complexity Assessment of High-Efficiency Video Encoders. *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 1899–1909. [\[CrossRef\]](#)
14. Xu, M.; Li, T.; Wang, Z.; Deng, X.; Yang, R.; Guan, Z. Reducing Complexity of HEVC: A Deep Learning Approach. *IEEE Trans. Image Process.* **2018**, *27*, 5044–5059. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Yi, Q.; Lin, C.; Shi, M. HEVC Fast Coding Unit Partition Algorithm Based on Deep Learning. *J. Chin. Comput. Syst.* **2021**, *42*, 368–373.
16. Wang, T.; Li, F.; Qiao, X.; Cosman, P.C. Low-Complexity Error Resilient HEVC Video Coding: A Deep Learning Approach. *IEEE Trans. Image Process.* **2021**, *30*, 1245–1260. [\[CrossRef\]](#) [\[PubMed\]](#)
17. Huang, H.; Schioppa, I.; Munteanu, A. Deep Learning Based Angular Intra-Prediction for Lossless HEVC Video Coding. In *Data Compression Conference Proceedings*; IEEE: Manhattan, NY, USA, 2019; p. 579. [\[CrossRef\]](#)
18. Schioppa, I.; Huang, H.; Munteanu, A. CNN-Based Intra-Prediction for Lossless HEVC. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 1816–1828. [\[CrossRef\]](#)
19. Huang, H.; Schioppa, I.; Munteanu, A. Low-Complexity Angular Intra-Prediction Convolutional Neural Network for Lossless HEVC. In *Proceedings of the IEEE 22nd International Workshop on Multimedia Signal Processing, MMSP 2020*, Tampere, Finland, 21–24 September 2020. [\[CrossRef\]](#)
20. Ting, H.C.; Fang, H.L.; Wang, J.S. Complexity Reduction on HEVC Intra Mode Decision with Modified LeNet-5. In *Proceedings of the 2019 IEEE International Conference on Artificial Intelligence Circuits and Systems, AICAS 2019*, Hsinchu, Taiwan, 18–20 March 2019; pp. 20–24. [\[CrossRef\]](#)
21. Kuang, W.; Chan, Y.L.; Tsang, S.H.; Siu, W.C. DeepSCC: Deep Learning-Based Fast Prediction Network for Screen Content Coding. *IEEE Trans. Circuits Syst. Video Technol.* **2020**, *30*, 1917–1932. [\[CrossRef\]](#)
22. Feng, Z.; Liu, P.; Jia, K.; Duan, K. HEVC Fast Intra Coding Based CTU Depth Range Prediction. In *Proceedings of the 2018 3rd IEEE International Conference on Image, Vision and Computing, ICIVC 2018*, Chongqing, China, 27–29 June 2018; pp. 551–555. [\[CrossRef\]](#)
23. Huang, S.; Zhang, Q.; Li, M.; Zheng, X. Fast HEVC SCC Intra-Frame Coding Algorithm Based on Deep Learning. *Comput. Eng.* **2020**, *46*, 42–47.
24. Kuanar, S.; Rao, K.R.; Bilas, M.; Bredow, J. Adaptive CU Mode Selection in HEVC Intra Prediction: A Deep Learning Approach. *Circuits Syst. Signal Process.* **2019**, *38*, 5081–5102. [\[CrossRef\]](#)
25. Bouaafia, S.; Khemiri, R.; Sayadi, F.E.; Atri, M. Fast CU Partition-Based Machine Learning Approach for Reducing HEVC Complexity. *J. Real-Time Image Process.* **2019**, *17*, 185–196. [\[CrossRef\]](#)

-
26. IMO SN.1/Circ.243/Rev.2. Guidelines for the Presentation of Navigation-Related Symbols, Terms and Abbreviations. In *NCSR 6/WP.4 Report of The Navigational Working Group*; IMO: London, UK, 2019.
 27. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going Deeper with Convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, USA, 7–12 June 2015; pp. 1–9.