

Article

HEVC's Intra Mode Selection Using Odds Algorithm

Ammar Armghan ^{1,*}, Junaid Tariq ², Fayadh Alenezi ¹, Norah Alnaim ³, Ayman Alfalou ⁴
and Saad Rehman ²

¹ Department of Electrical Engineering, Jouf University, Sakaka 72388, Saudi Arabia; fshenezi@ju.edu.sa

² Department of CSE, HITEC University, Taxila 47050, Pakistan; junaid.tariq@hitecuni.edu.pk (J.T.); saad.rehman@hitecuni.edu.pk (S.R.)

³ Department of Computer Science, College of Sciences and Humanities in Jubail, Imam Abdulrahman bin Faisal University, Dammam 32241, Saudi Arabia; nmalnaim@iau.edu.sa

⁴ L@bISEN, Equipe LSL, Yncrea Ouest, 20 Rue Cuirasse Bretagne, 29200 Brest, France; ayman.al-falou@isen-ouest.yncrea.fr

* Correspondence: aarmghan@ju.edu.sa

Abstract: The brute-force behaviour of High-Efficiency Video Coding (HEVC) is the biggest hurdle in the communication of the multimedia contents. Therefore, a novel method will be presented here to expedite the intra mode decision process of HEVC. In the first step, the feasibility of the odds-algorithm for the early intra mode decision is presented by using statistical evidence. Then, various elements of odds algorithm are analyzed and then mapped to the intra mode process (elements) of HEVC. Finally, the probability required by the odds algorithm is obtained by utilizing the correlation between the current and the neighboring blocks. The proposed algorithm accelerated the encoding process of the HEVC by 25% to 35%, while the Bjontegaard Delta Bit Rate (BD-BR) is 0.95% to 1.84%.

Keywords: odds algorithm; HEVC; intra mode; fast encoding



check for updates

Citation: Armghan, A.; Tariq, J.; Alenezi, F.; Alnaim, N.; Alfalou, A.; Rehman, S. HEVC's Intra Mode Selection Using Odds Algorithm. *Electronics* **2021**, *10*, 1839. <https://doi.org/10.3390/electronics10151839>

Academic Editor: Javid Taheri

Received: 22 June 2021

Accepted: 24 July 2021

Published: 31 July 2021

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

A question by a YouTuber: How good I am doing? This question can be answered by looking at the number of views and this will be most satisfying answer for any person. In U.S. alone, 75% of adults are YouTube users. According to Mark Zuckerberg, the impact of videos will increase in the future. One will be inclined to agree with Mark Zuckerberg after going through the following statistics [1]: TV content is 100% video-based, online marketing is 80% video-based, and so on. According to YouTube alone, 1 billion hours of videos are watched each day. Moreover, YouTube offers different quality levels for each video. Obviously, the user selects the HD (High Definition) level, even his/her bandwidth is less. To support all the users' different internet connections, there is a need of a good compression algorithm. A video compression algorithm reduces the size of the video by removing spatial and temporal redundancies. The main goal of such algorithms is that they try to preserve the quality of the video content while reducing the size of the video.

Another reason for reducing the size of the video is that now videos are captured at 30+ frames per second. This means that there are 30 pictures in each second. In simple words, there are around 1800 images that should be transferred from the source (YouTube) to destination (user) to watch only 1 min long video. Furthermore, as the resolution (height and width) of the image will increase, it will further increase the size of the data that will be required to be transmitted.

HEVC [2] is the latest compression algorithm; it is also known as H.265. This algorithm achieved many breakthroughs but in this article, we will focus on intra mode decision. It is worth mentioning that HEVC reduced the bits required by 50% [3] as compared to the previous compression algorithms. As a result, industry and researchers focused on HEVC and many advancements have been done to this marvelous algorithm. HEVC used big block size (64×64) to compress the smooth regions of the video and utilized small block

size (8×8) to compress the busy regions of the video. HEVC uses recursive partitioning of a 64×64 block into 4 blocks of equal sizes. This process continues until block size reaches 8×8 . For each block, regardless of its size, its contents are either intra or inter predicted. HEVC tries all the block sizes to achieve the best partition. The best partitioning in HEVC is related to the least Rate and Distortion (RD) cost. Here, “rate” means the bits required and “distortion” means the loss in the quality.

One can notice that if bits are reduced, then it further degrades the distortion, and if one keeps the quality, then bits are too high. Therefore, it's a chicken and hen dilemma. That is why HEVC tries different block sizes to find a best trade-off between bit rate (R) and quality (D). However, this procedure of HEVC increased the complexity of compression. As a result, HEVC became unfit for real-time applications. To overcome this issue, researchers produced fast algorithms that reduced the complexity of HEVC. This complexity reduction is measured in the form of time saving achieved with respect to HEVC normal time consumed to compress the same video.

To understand the working of HEVC, we have to study HEVC test Model (HM) software [4] that performs various tasks of HEVC. HM processes the video, frame by frame. Each frame is divided to square blocks of 64×64 pixels. This is the biggest block size in HEVC and its known as Coding Unit (CU). This CU will be either intra coded or inter coded, and its RD cost is saved. The CU is then further divided and 4 CUs of 32×32 pixels are achieved. The same prediction mechanism is applied on each of these 4 CUs. The accumulative RD cost of this level (RD cost of 4 CUs) will be compared with the RD of 64×64 , and the information about the best partition and RD cost is updated in HM. HEVC repeats the same CU division and prediction process on each of these 32×32 CU. Then, the accumulative RD cost of these 16 CUs will be compared with the previous best RD cost. In each step, the best RD cost is updated. This process continues until the CU size reaches 8×8 . At this level, one can see the best possible partition level and its related RD cost.

The division process explained above is not only complex, but also brute-force in nature. Now, we will discuss the prediction of CU. No matter what is the size of CU, it will be either intra or inter predicted [5]. Here, we will focus on intra prediction [6]. In order to predict the CU using intra prediction, the neighboring pixels of the CUs are projected on the region of the current CU in 33 different angles. These are known as angular modes. The direction of angular modes are shown in Figure 1. These angular modes handle the texture very well, but for smooth regions, HEVC has 2 more modes known as DC and Planar modes. Therefore, HEVC applies 35 intra modes on each CU to find an optimal mode for this CU. By optimal mode we mean that the intra mode that gives the least RD cost. HEVC performs this intra prediction mechanism on each CU to find the optimal RD cost. No doubt this process will find the best RD trade-off but the complexity and encoding time are increased.

The intra modes angles are very close to each other, and as a result some intra modes give the same RD cost. This anomaly was seen during the experimentation by the author. An experiment was set up to note the number of CUs that have same RD cost among its intra modes. The result of this experiment is given in Figure 2. Figure 2 shows that large number of CUs are facing this issue. Therefore, as some intra modes will give us the same RD cost, why should we evaluate them? Because evaluating them will just increase the computation time. The problem is that we do not know which two intra modes will result in the same RD cost until we evaluate them.

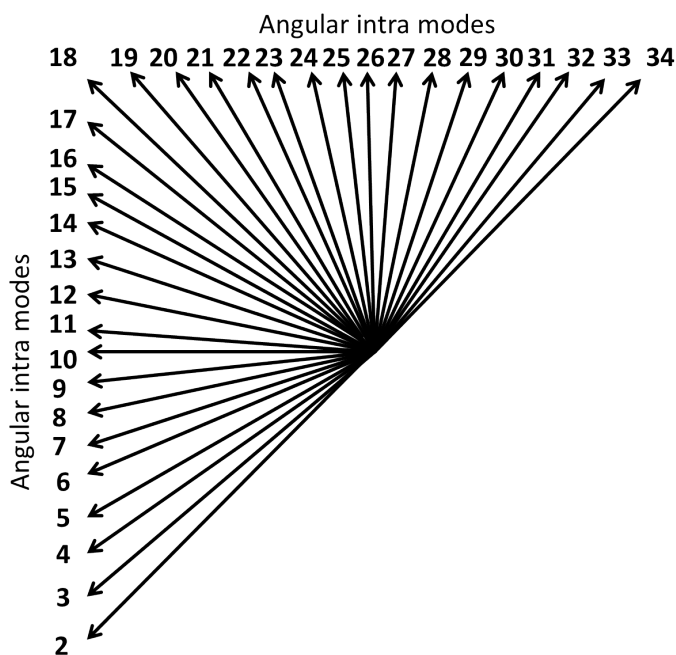


Figure 1. Modes in HEVC.

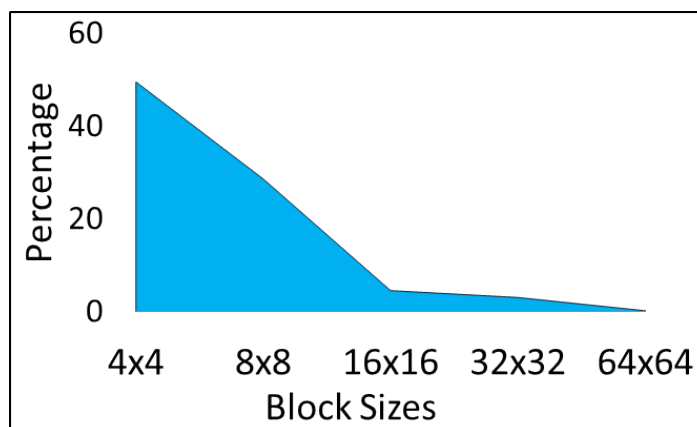


Figure 2. RD cost similarity among blocks.

The proposed work was motivated from the current trend and advancement in the field of statistic and computational efficient stopping strategy, known as the *odds algorithm* [7]. The odds algorithm successfully handles the odds situations of the real-world. The intra modes are passed to the odds algorithm that computes the stopping point for the intra mode decision process. Second, the odds algorithm utilizes the probability of modes being evaluated and the probability of the remaining modes and then gives the winning probability. Such winning probability is not only efficient, but also very useful because it is based on the complete population. Some of the contributions of this article include the following:

1. The elements of the odds algorithm are mapped to HEVC to find out intra mode for the current block.
2. The odds algorithm is utilized in a very unique way.
3. A novel stopping point computational model is proposed that is based on the odds algorithm.

This article is structured as follows. The related works, motivation, proposed model, results, and conclusion are presented in Sections 2–6, respectively.

2. Related Works

Researchers developed interest in the HEVC because it not only reduced the bit rate requirement by 50%, but also delivered the same subjective quality as its predecessors. Another area of HEVC is also explored by the researchers that deal with fast algorithms. By fast algorithm we mean that the encoding process of HEVC is reduced with negligible bit rate increase.

The authors of [8] proposed a model that is based on the classical secretary problem (CSP). In this algorithm, the stopping point is found using the CSP-based model, that helps in reducing the encoding time of HEVC. This article saved 30% of total encoding time. Kuang, in [9], makes an early decision about the intra mode using the decision tree approach of machine learning. This article saved 47% of the total encoding time. The author of [10] studied the factors of intra coding and then proposed a learned framework that achieves a satisfactory trade-off between complexity reduction and coding performance.

Wang, in [11], studied the correlation between parent CU's intra mode and the child CU's intra mode. The author then reduces the intra modes for the current block by looking at the intra mode for the parent CU. This article reduced the total encoding time by 47%. The author of [12] saved 60% of the total encoding time. In this algorithm, the author tries a subset of intra modes. In [13], Schafer deals with the Matrix-Based intra prediction method. The author describes the complete process of designing Matrix-Based intra prediction. The work presented by Yang in [14] makes different groups of the prediction units by analyzing the texture of the block. Moreover, 35 intra modes are also grouped to reduce the number of modes for RMD (rough mode decision) module of HEVC. This article reduced the encoding time by 22% on average. Wang, in [15], formed three groups by looking at the complexity of the CU. The author of this article proposed his own method to compute the complexity. The author of [16] uses the RD cost of the previous information to predict the RD cost for the current block. This predicted RD cost is then used as a stopping RD cost.

The author of [17] proposed a hardware design to achieve high throughput for the intra mode decision. This is the first work to propose a design for AV1 intra prediction. In [18], Zhang utilized the gradient of the current CU to predict the intra mode. This work mapped the gradient of the CU with the intra mode. Hu, in [19], utilized a regression algorithm to predict the intra mode. This regression algorithm saved 55% of the total encoding time. Tariq in [20] proposed a model that is based on the optimal stopping theory. This algorithm saved 35% of the total encoding time of HEVC and the BDBR is 1.24%.

Kuanar, in [21], used a CNN to make early intra mode decision. This article saved 45% of the total encoding time of HEVC. In [22], Huang proposed a generic early decision model. According to the author of [22], this model can be used to perform early CU decision, early PU decision, early intra mode decision, etc. This article saved 66% of the total encoding time of HEVC. Yan, in [23], utilized random forest to make early intra mode decision. This article saved 58% of the total encoding time of HEVC. Over half (52%) of time is saved by Hossein in [24]. This article makes an early decision about the angular intra mode. This decision is based on the planar mode's output.

The author of [25] utilized the deviation that is present between the pixels of the CU. This deviation is then used to make early intra mode decision. This article saved 20% of the total encoding time of HEVC and BDBR is only 0.2%. Gwon, in [26], utilized a classifier to make an early intra mode decision. This article saved 31% of total encoding time of HEVC. Tariq, in [27], proposed a model that is used to make early intra mode decision in HEVC. In [28], the author utilized guided-filter to make an early decision about the intra mode for the current block. The probability of modes is used as an input for the guided filter. This article saved 30% of the total encoding time.

In [29], the author applied Benford's law which is a statistical model. The author claimed to save 32% of the encoding time and BDBR is 1.54% on average. Tariq, in [30], achieved fast intra mode decision using firefly (FFA) algorithm. This FFA-based algorithm saved 27% of the encoding time with BDBR of 1.09% on average. In [31], fast intra mode decision model is presented, and early termination is achieved using a threshold that is

based on Presman threshold. This work further simplified the Presman threshold, and its details can be found in [31]. This work saved 34.10% of the encoding time while BDBR is 1.54% on average. In [32], the author applied world-war-2 (WW2) technique to make an early decision about the intra mode for the current block. This work saved 26.88% of the encoding time while the BDBR is 1.97% on average.

The proposed work, in comparison to the state-of-the-art works presented above, tries to extract and incorporate all the available information regarding all the modes to make a stopping decision. In statistics, this is known as total population (probabilities of all the eight intra modes). The proposed work separates the probabilities of modes evaluated so far and the probabilities of remaining modes. These two separations are handled with the help of odds algorithm, because its formulation and working is already proven. Moreover, an odds algorithm is capable of handling odds situations of the real world as it incorporates the total population (probabilities of evaluated and remaining modes). In the coming sections, various elements of the odds algorithm are analyzed and graphically presented for easy understanding. Later, these elements are mapped to HEVC for early intra mode decision.

3. Motivation

The logic behind using an odds algorithm will be presented here and then it will be mapped to the intra mode decision problem of HEVC. The main contribution of this section will be the synchronization and mapping of various elements of the odds algorithm with the elements of intra procedure. In this way, we can apply the odds algorithm to its greatest potential.

3.1. Odds Algorithm

The odds algorithm [7] was proposed by Bruss to find out the optimal strategy. This strategy may be applied to stopping problems that include classical secretary problem, optimal stopping theory, etc. Real-world problems sometimes result in odd situations and behaviour, and therefore odds algorithms are required. The importance and contribution of the odds algorithm will be presented here along with its explanation.

The main contribution of the odds algorithm is that it maximizes the probability of finding the optimal element from a sequential list that satisfies a strategy. Most importantly, the odds algorithm performs this element finding at the time of observation of that specific element. In other words, no revisiting of missed elements is permitted. This feature might look strict in nature, but in real-time the best bidder (first highest) works on this algorithm. That is why the author of this article argues that this algorithm is also useful when selecting an element (intra mode) from the list because each new element (intra mode) adds complexity to the intra mode selection process.

For example, a car will be sold to the highest bidder. The bidders see the car one by one and make an offer to the seller, who will immediately either accept or reject the offer. Moreover, each bid will be random and the seller has a fear that this bid might be the last good offer. Thus, selecting the last good offer means maximizing the probability of selling to the highest bidder.

Suppose we have n bidders denoted as B_1, B_2, \dots, B_n , where $B_k = 1$ means that specific bid (k) is interesting. On the other hand, $B_k = 0$ means that offer is not interesting. Formally, we can say that

$$\begin{aligned} p_k &= P(B_k = 1) \\ q_k &= P(1 - p_k) \end{aligned} \quad (1)$$

The odds (r_k) of the odds-algorithm about the k^{th} option turning to be interesting is represented using

$$r_k = \frac{p_k}{q_k} \quad (2)$$

The odds algorithm adds up the odds to obtain R_k as

$$R_k = r_1 + r_2 + \dots + r_k \quad (3)$$

At the same time, odds algorithm computes Q_k as

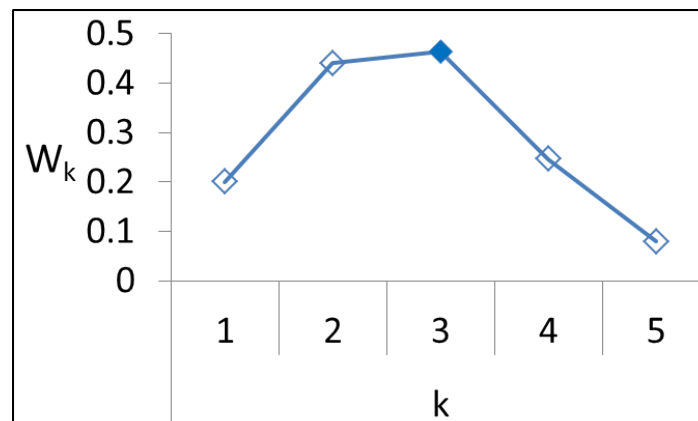
$$Q_k = q_1 \times q_2 \times \dots \times q_k \quad (4)$$

The stopping point k and the winning probability W_k is found using odds algorithm as

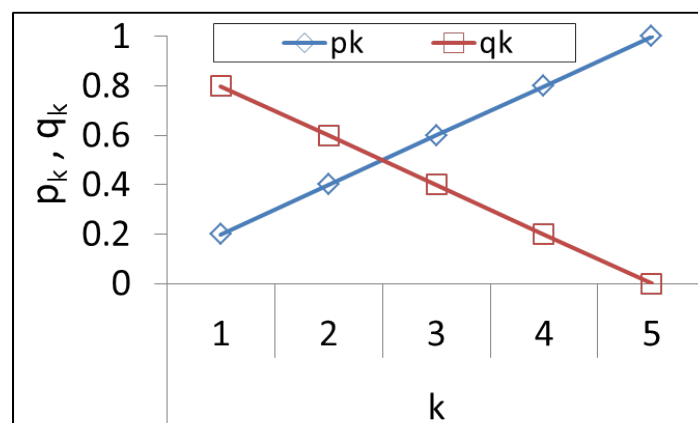
$$k = \min\{k : R_k > 1.0\} \quad (5)$$

$$W_k = Q_k R_k$$

This was the odds algorithm. Its visualization will be presented here to highlight one interesting factor. Consider 5 values each having probability of 0.2 such that if you add these 5 probabilities, you will get 1. The visualization of W_k element of odds algorithm is shown in Figure 3a, where the x-axis represents 5 values. Figure 3a shows that there is a point (at 3rd element) that gives the maximum winning probability (W_k). Interestingly, there is only 1 maximum winning probability in the odds algorithm. This will be proved in the next subsection in which odds algorithm will be applied on two examples. This is the unique working procedure of the odds algorithm. On the other hand, if one evaluates these values (0.2) one by one, then p_k will increase because it shows probability of *success* and q_k will decrease as it shows probability of *failure*. This process is also shown in Figure 3b. In the coming subsection, detail analysis of odds algorithm will be performed and then it will be mapped to the intra process of HEVC.



(a)



(b)

Figure 3. Success and failure concept. (a): Plot of W_k . (b): Plot of p_k and q_k .

3.2. Probability for Odds Algorithm

This sub-section will further perform the analysis of odds algorithm by applying it on two examples. As we know that odds algorithm works on probability. Therefore, the probability of intra modes is obtained for applying the odds algorithm. As current block in HEVC has neighboring blocks and due to their correlation with each other, the probability of intra mode J can be obtained by consulting the modes of the neighboring blocks. To obtain this probability matrix, first initialize a 35×35 matrix with zeros. Then, execute HEVC encoding process and whenever HEVC selects the intra mode for the current block, extract the intra mode of the above neighboring block (M). Then, go to J row and M column of this matrix and increment its value. After this, divide the J by the sum of all the J s present in this row and it will give the probability. In this work, we computed *above* matrix only, using *BasketballDrill* test sequence to keep the computation minimum as this is fast intra mode decision algorithm. This matrix is shown in Figure 4. In Figure 4, x -axis represents M , y -axis represents J , and bars represent the count. Figure 4 shows that some modes have huge probability and some has less probability.

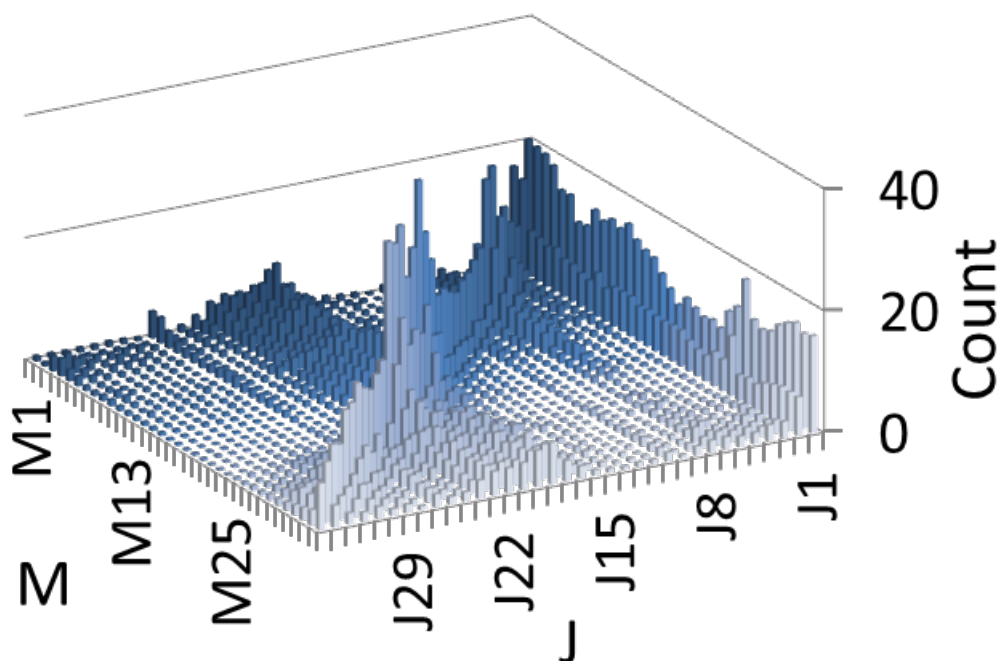
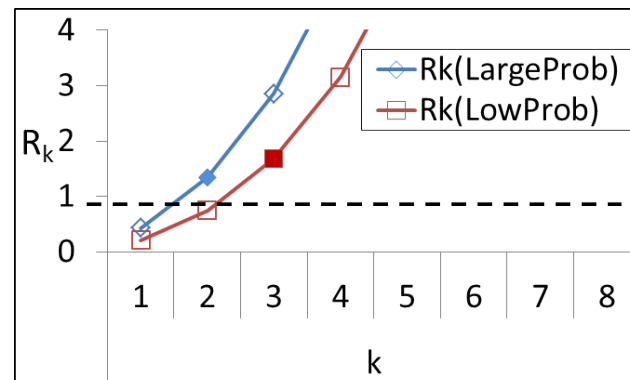


Figure 4. Probabilities of intra mode based on its neighboring blocks.

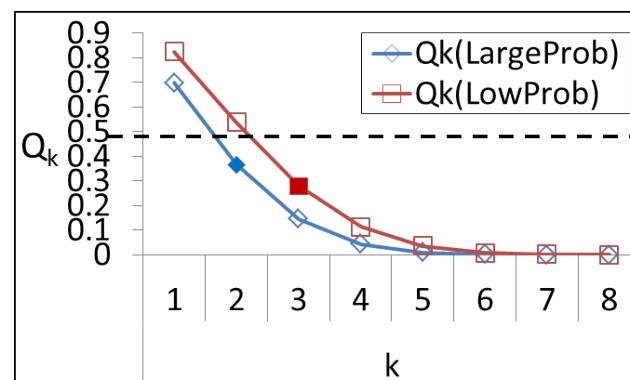
Two sets of values are selected from this probability matrix for analysis purpose. Set of values are selected because RMD module of HEVC shortlist up to 8 intra modes for the next module, i.e., RDO module. We have termed these sets as $LargeProb = \{ 0.30, 0.17, 0.12, 0.09, 0.07, 0.06, 0.05, 0.05 \}$ and $LowProb = \{ 0.17, 0.17, 0.13, 0.11, 0.09, 0.08, 0.07, 0.07 \}$. In $LargeProb$, the initial values are large which represent initial values have high chance to be selected and $LowProb$ represent the case in which initial values have low chance to be selected.

The odds algorithm is applied on the two sets and their results are graphically presented in Figure 5. Figure 5a shows the result of R_k . The odds algorithm finds the optimal stopping element when $R_k \geq 1$. Figure 5a shows that when probability is large ($LargeProb$ case), this limit, i.e., 1, is achieved at the 2nd element. On the other hand, when probability is low ($LowProb$ case), the limit is delayed to 3rd element. Figure 5b shows the Q_k element of odds algorithm. The Q_k element shows the probability of loss. Figure 5b shows that when probability is low ($LowProb$ case), the chance of failure is large. The optimal condition is the place where the loss is less than or equal to the probability of winning. The neutral point in probability is given by 0.5 value and this can be considered as the

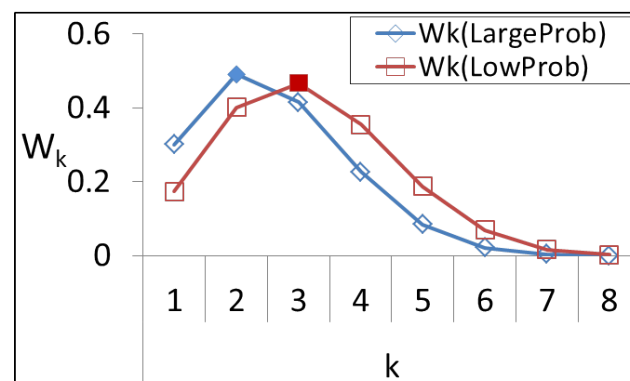
minimum winning probability. Last, Figure 5c shows the win probability of odds algorithm and this is the most important and interesting element of the odds algorithm. So far, we have figured out that 2nd element is the best in *LargeProb* case and 3rd element is the best in *LowProb* case. The win probability of odds algorithm represents the same information but in a different way. In Figure 5c, the highest win probability of odds algorithm is at the same element, i.e., 2nd and 3rd for *LargeProb* and *LowProb* case, respectively. After this point (highest win probability), the win probability decreases. The good thing about the odds algorithm is that it stops at the highest win probability (highest bidder) and it does not waste time that is spent during recall by other algorithms.



(a)



(b)



(c)

Figure 5. Output of odds algorithm on two examples. (a): Plot of R_k . (b): Plot of Q_k . (c): Plot of W_k .

Due to odds's algorithm simplicity and recall avoidance behaviour, it will be applied on the probability of intra modes to reduce the complexity of HEVC.

4. Proposed Odds-Based Algorithm

The proposed algorithm presented here achieves fast intra mode selection using odds algorithm. The motivation section of this article has already presented odds algorithm and two intra mode decisions are performed (two sets) using it. In this section, all the elements of the odds algorithm will be mapped to the intra mode process (elements) of HEVC.

As natural images have correlation in them. Therefore, precomputed probability matrix will be first achieved, because it will be unnatural that a totally different mode is selected for the block. In other words, the mode of the current block will be related to the modes of its neighboring blocks. Therefore, the probability of the modes shortlisted by the RMD module for the current block will be passed to the odds algorithm. The odds algorithm will return the k and this will be the termination point for the intra mode decision process for the current block.

The odds algorithm will adjust dynamically depending upon the probabilities of the modes. If probability is high, then k will be small (early terminate the intra process). On the other hand, if probability is small, then k will be large (delay in termination). This happens because the odds algorithm is based on the probability of success and failure. To illustrate this success and failure concept, we make use of the basic probability equation:

$$\sum_{s=1}^{Total} P(s) = \sum_{s=1}^k P(s) + \sum_{j=k+1}^{Total} P(j) \quad (6)$$

Dividing by $\sum_{j=k+1}^{Total} P(j)$, we get

$$\frac{\sum_{s=1}^{Total} P(s)}{\sum_{j=k+1}^{Total} P(j)} - 1 = \frac{\sum_{s=1}^k P(s)}{\sum_{j=k+1}^{Total} P(j)} \quad (7)$$

The RHS is the R_k element of the odds algorithm.

$$R_k = \frac{\sum_{s=1}^k P(s)}{\sum_{j=k+1}^{Total} P(j)} \quad (8)$$

This shows the strength of the odds algorithm as it is not only simple, but also has a strong foundation. One can also use the LHS of (7) for early termination and then rule in (5) also needs to be updated accordingly. However, this will result in a new algorithm, and it is out of scope of this article because this article is about application of odds algorithm. Therefore, the true form of odds algorithm is applied to achieve the fast intra mode decision. Finally, the proposed algorithm based on odds algorithm becomes

$$K^* = \min \left\{ 1 \leq k \leq n : \sum_{i=1}^k R_k \geq 1 \right\} \quad (9)$$

where

$$R_k = r_1 + r_2 + \dots + r_k$$

$$r_k = \frac{p_k}{q_k}$$

The proposed algorithm stops at the first K^* that satisfies the rule $\sum_{i=1}^k R_k \geq 1$. As explained earlier, this is the win probability of odds law. Therefore, the intra mode decision is stopped after evaluating mode K^* . The block diagram of the proposed algorithm is shown in Figure 6 in which N elements are evaluated one by one until either stopping criteria is met or the system run out of elements.

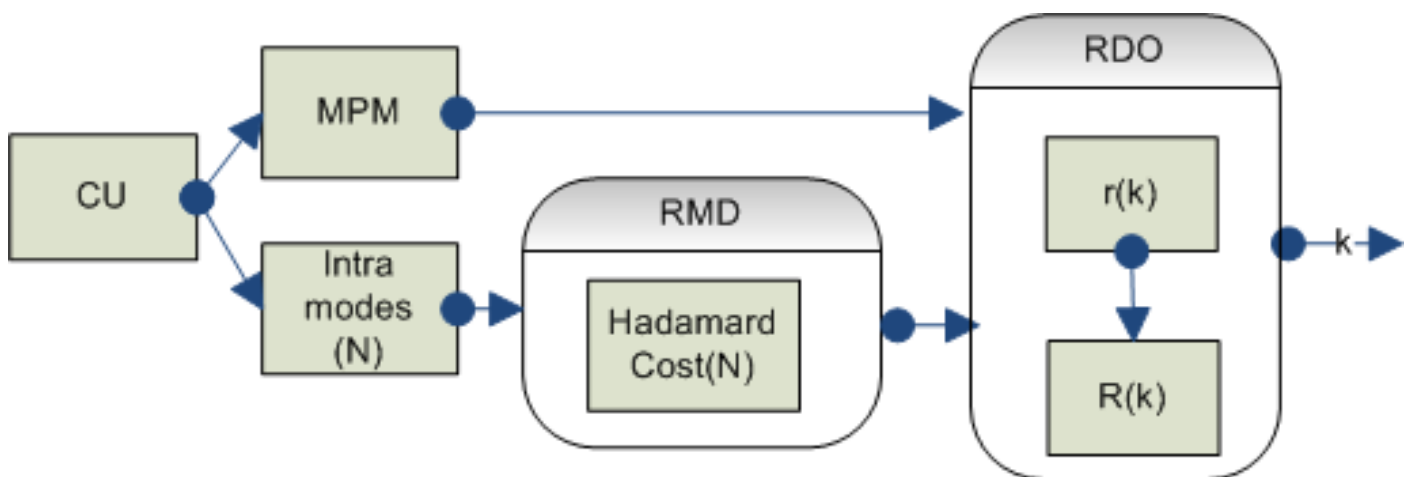


Figure 6. Block diagram of the proposed algorithm.

Minimum Duration to Find the Optimal Element

In this subsection, we will discuss about the minimum number of elements that will be evaluated by the proposed algorithm before taking any decision. The stopping criteria of proposed algorithm resides in the equation $R_k > 1$. This limit will be met when p_k becomes greater than q_k . Moreover, this will happen if the probability of success is greater than the probability of failure.

$$p_k > q_k \tag{10}$$

In order words, it means that one should be sure of his success. Lets see this from another angle: Suppose we have evaluated k elements out of total N elements, then $N - k$ represents the remaining elements and according to odds algorithm it can be written as

$$\frac{k}{N - k} > 1 \tag{11}$$

where 1 represents the stopping criteria of the odds algorithm. Simplifying, we get

$$\begin{aligned} \frac{k}{N - k} &> 1 \\ k &> N - k \\ k + k &> N \\ 2k &> N \\ k &> \frac{N}{2} \end{aligned} \tag{12}$$

This also shows that one should evaluate 50% of the elements before stopping. In this way, a good number of elements are evaluated, and making a decision at this stage will enhance the chance of winning.

5. Experimental Results

This section will present and discuss the encoding results of the proposed algorithm for HEVC. A limitation of this algorithm will also be presented with facts and figures. In the end, the comparison will be conducted with the existing algorithms.

HM software (16.9 version) was downloaded from the address in [4], and the proposed algorithm is implemented in it to obtain the encoding results. The following settings are used during the encoding process to make a fair comparison with existing algorithms: 4 QPs i.e., {22, 27, 32, and 37}; videos from A-F classes; and *All-Intra-Main* configuration are selected as directed in [33]. Three performance metrics, i.e., BD-PSNR, Time-saving (ΔT),

and BD-BR are used as directed in [34]. Detail and mathematical definitions of BD-BR and BD-PSNR can be found in [35]. The ΔT is measured using

$$\Delta T = \frac{(T_{HEVC} - T_{PROPOSED})}{T_{HEVC}} \times 100\% \quad (13)$$

5.1. Encoding Results of Odds Algorithm Based Early Termination in HEVC

The results of the odds algorithm based intra mode decision algorithm for the HEVC data set are presented in Table 1. In Table 1, ΔP , ΔR , and ΔT represent BD-PSNR, BD-BR, and time-saving, respectively. The encoding results are obtained for different values of R_k . The results for $R_k = 1.0$ represent the exact working of the odds algorithm, because odds algorithm suggested this value (1.0). In order to improve the time saving, the results are also obtained for $R_k = 0.5$ and $R_k = 0.3$. The results for $R_k = 0.3$ gives a reasonable balance between ΔR and ΔT . Table 1 shows that the proposed algorithm reduces 35% of the encoding complexity of HEVC.

Table 1. Results of odds algorithm.

Classes	Video	$R_k = 1.0$			$R_k = 0.5$			$R_k = 0.3$		
		ΔP	ΔR	ΔT	ΔP	ΔR	ΔT	ΔP	ΔR	ΔT
A 2560 × 1600	Traffic	−0.05	0.93	24.88	−0.09	1.67	32.22	−0.10	1.94	35.17
	PeopleOnStreet	−0.06	1.12	26.55	−0.10	1.77	34.1	−0.12	2.1	36.72
B 1920 × 1080	Cactus	−0.04	1.01	25.06	−0.06	1.48	32.66	−0.07	1.78	35.44
	Kimono	−0.05	1.28	24.53	−0.07	1.99	32.15	−0.08	2.33	34.93
C 832 × 480	RaceHorses	−0.07	1.07	25.02	−0.10	1.4	32.77	−0.10	1.43	34.73
	PartyScene	−0.08	1.01	25.41	−0.14	1.72	33.32	−0.15	1.85	35.09
D 416 × 720	BQSquare	−0.08	1.01	25.71	−0.14	1.57	33.95	−0.17	1.96	35.74
	BasketballPass	0.00	−0.03	25.67	−0.05	0.74	33.28	−0.08	1.21	35.67
E 1280 × 720	FourPeople	−0.08	1.32	26.40	−0.11	1.87	33.54	−0.12	2.12	36.22
	Johnny	−0.06	1.47	25.97	−0.08	1.86	32.89	−0.09	2.22	35.69
F 1280 × 720	ChinaSpeed	−0.08	1.01	25.69	−0.14	1.57	33.00	−0.17	1.96	35.46
	BasketballDrillText	−0.01	0.22	25.56	−0.04	0.8	32.98	−0.06	1.13	35.65
Avg.		−0.06	0.95	25.54	−0.09	1.54	33.07	−0.11	1.84	35.54

These results show that if the value of R_k is reduced, then time saving will be increased. On the other hand, if the value of R_k is increased, then time saving will be decreased. Demanding upon the situation, one can alter the working of the algorithm by simply changing one variable.

By changing the value of R_k , one can see the difference in time saving, but what about the visual difference? The answer to this question is given in Figure 7. In Figure 7a, the value of R_k is 0.5 and a PSNR comparison with the original picture is given in Figure 7c. Dark blue color in Figure 7c indicates exact match or difference is zero. Similarly, Figure 7b is encoded using $R_k = 0.3$ and its PSNR difference with the original picture is shown in Figure 7d. Figure 7a,b clearly shows that there is no visual difference but time saving of $R_k = 0.3$ is better. The application of the proposed model on another video sequence (PartyScene) is shown in Figure 8. For easy comparison, encoding result of HEVC is also shown in Figure 8. Visual results of Figure 8 show that there is not much difference and both the figures look almost the same.

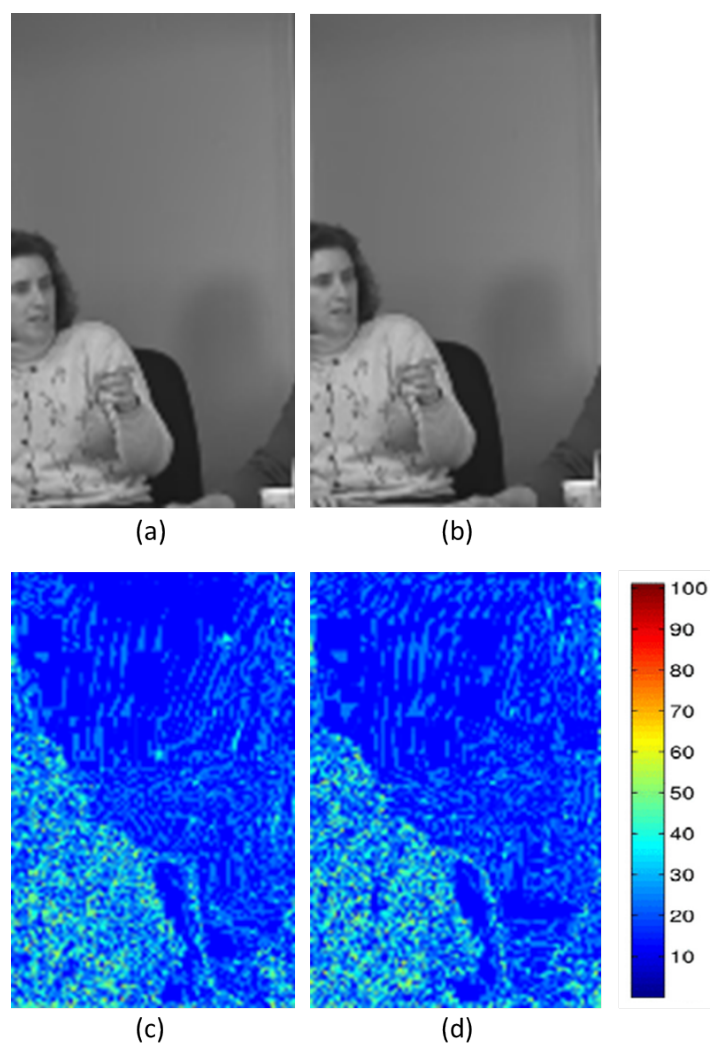


Figure 7. PSNR comparison. (a): Picture encoded using $R_k = 0.5$. (b): Picture encoded using $R_k = 0.3$. (c): PSNR comparison of $R_k = 0.5$ picture with the original image. (d): PSNR comparison of $R_k = 0.3$ picture with the original image.



Figure 8. Application of proposed algorithm on PartyScene test video sequence of HEVC.

5.2. Comparison with Literature

The literature is full of fast intra algorithms. Out of which, some of the novel algorithms are selected from the literature and presented in Table 2. The term *Odds Algorithm* in Table 2 represents the proposed algorithm.

The two most important and recent intra mode decision algorithms [20,36] are implemented and tested on the same platform for comparison purpose. The results of these algorithms are also presented in Table 2. Table 2 shows that the algorithm from in [20] saves 29% of the encoding time, while the algorithm from in [36] saves 20% of the total encoding time. Table 2 also shows that the proposed algorithm saves more time than the two aforementioned algorithms.

Table 2. Odds algorithm vs. existing algorithms.

Classes	Video	[20]			[36]			Odds Algorithm		
		ΔP	ΔR	ΔT	ΔP	ΔR	ΔT	ΔP	ΔR	ΔT
A 2560 × 1600	Traffic	-0.07	1.21	30.61	-0.036	0.65	17.63	-0.10	1.94	35.17
	PeopleOnStreet	-0.07	1.22	31.2	-0.025	0.45	17.97	-0.12	2.1	36.72
B 1920 × 1080	Cactus	-0.04	1.01	30.52	-0.023	0.59	17.46	-0.07	1.78	35.44
	Kimono	-0.03	0.74	29.34	-0.02	0.61	16.53	-0.08	2.33	34.93
C 832 × 480	RaceHorses	-0.07	1.04	30.37	-0.026	0.42	17.77	-0.10	1.43	34.73
	PartyScene	-0.12	1.55	31.59	-0.044	0.56	18.82	-0.15	1.85	35.09
D 416 × 720	BQSquare	-0.12	1.29	28.57	-0.037	0.39	21.16	-0.17	1.96	35.74
	BasketballPass	-0.07	1.14	26.89	-0.021	0.32	20.41	-0.08	1.21	35.67
E 1280 × 720	FourPeople	-0.08	1.36	30.37	-0.039	0.68	18.73	-0.12	2.12	36.22
	Johnny	-0.06	1.5	28.8	-0.027	0.65	17.85	-0.09	2.22	35.69
F 1280 × 720	ChinaSpeed	-0.12	1.29	29.7	-0.037	0.39	43.08	-0.17	1.96	35.46
	BasketballDrillText	-0.08	1.49	29.82	-0.034	0.63	18.43	-0.06	1.13	35.65
Avg.		-0.08	1.24	29.82	-0.03	0.53	20.49	-0.11	1.84	35.54

Figure 9 analyzes the proposed algorithm from another perspective. Figure 9 presents the time saving achieved by various algorithms of the literature. These algorithms include *Benford* [29] published in 2020, *Rdcost* [37] published in 2019, *Texture* [20] published in 2018, *OST* [36], and *Tian* [25] published in 2019. Figure 9 shows that the proposed algorithm saves a good amount of time.

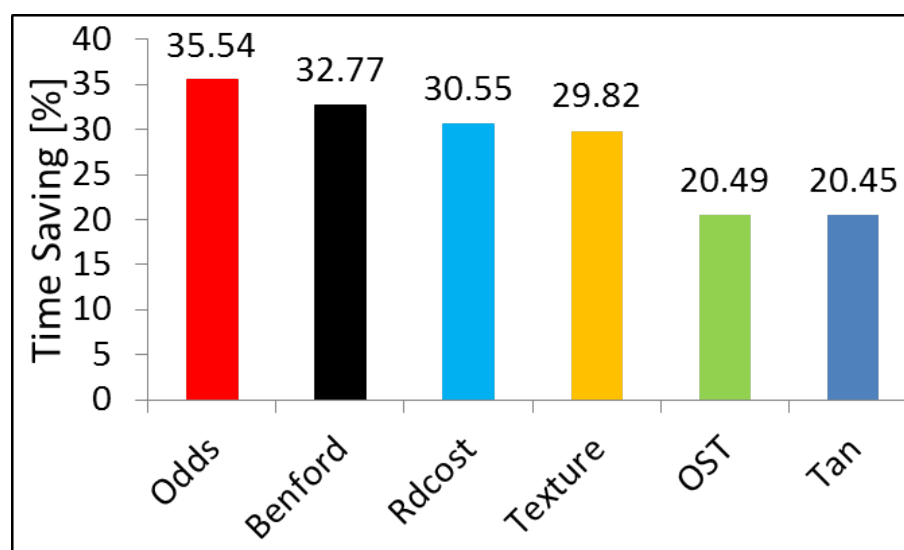


Figure 9. Time saving achieved by various algorithms.

In future, we are planning to apply the proposed algorithm on Versatile Video Coding (VVC) to see how it performs and handles large amount of intra modes present in VVC.

6. Conclusions

The intra mode computation process of HEVC is expedited up to 35% by utilizing the odds algorithm. The proposed algorithm successfully handles various odds situations of the video coding by efficiently incorporating the odds strategy of the odds algorithm. Moreover, the proposed algorithm is dynamic and adapts to the situation by simply changing the R_k element of the odds algorithm. The proposed algorithm solves the same problem as done by the existing works of the literature, but the methodology of the proposed algorithm is very unique as this is the first article that used odds algorithm for intra mode decision. Moreover, the proposed algorithm outperforms the latest works like in [29–31] in terms of time saving.

Author Contributions: Conceptualization and methodology, J.T., A.A. (Ammar Armghan) and F.A.; validation, N.A., S.R. and A.Alfalou; writing—original draft preparation, J.T. and A.A. (Ammar Armghan); writing—review and editing, J.T., A.A. (Ammar Armghan), and A.A. (Ayman Alfalou); All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

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

References

1. Biteable. Available online: <https://biteable.com/blog/video-marketing-statistics/> (accessed on 1 May 2020).
2. Kemal, U.; Andersson, K.; Fuldseth, A.; Bjontegaard, G.; Endresen, L.; Lainema, J.; Hallapuro, A.; Ridge, J.; Rusanovskyy, D.; Zhang, C.; et al. High performance, low complexity video coding and the emerging HEVC standard. *IEEE Trans. Circuits Syst. Video Technol.* **2010**, *20*, 1688–1697.
3. Ohm, J.R.; Sullivan, G.J.; Schwarz, H.; Tan, T.K.; Wiegand, T. Comparison of the coding efficiency of video coding standards including high efficiency video coding (HEVC). *IEEE Trans. Circuits Syst. Video Technol.* **2012**, *22*, 1669–1684. [CrossRef]
4. HEVC Test Model. Available online: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/ (accessed on 5 April 2020).
5. Junaid, T.; Kwong, S.; Yuan, H. Spatial/temporal motion consistency based MERGE mode early decision for HEVC. *J. Vis. Commun. Image Represent.* **2017**, *44*, 198–213.
6. Junaid, T.; Kwong, S.; Yuan, H. HEVC intra mode selection based on rate distortion (RD) cost and sum of absolute difference (SAD). *J. Visual Commun. Image Represent.* **2016**, *35*, 112–119.
7. Thomas, B.F. Sum the odds to one and stop. *Ann. Probab.* **2000**, *28*, 1384–1391.
8. Junaid, T. Intra mode selection using classical secretary problem (CSP) in high efficiency video coding (HEVC). *Multimed. Tools Appl.* **2019**, *78*, 31533–31555.
9. Wei, K.; Chan, Y.-L.; Tsang, S.-H.; Siu, W.-C. Machine learning-based fast intra mode decision for HEVC screen content coding via decision trees. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 1481–1496.
10. Zhibo, C.; Shi, J.; Li, W. Learned fast HEVC intra coding. *IEEE Trans. Image Process.* **2020**, *29*, 5431–5446.
11. Yu, W.; Su, Z. An efficient intra prediction algorithm for HEVC intra-coding. In Proceedings of the 2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Chengdu, China, 17–19 October 2020; pp. 407–412.
12. Hao, Z.; Ma, Z. Fast intra mode decision for high efficiency video coding (HEVC). *IEEE Trans. Circuits Syst. Video Technol.* **2014**, *24*, 660–668.
13. Michael, S.; Stallenberger, B.; Pfaff, J.; Helle, P.; Schwarz, H.; Marpe, D.; Wiegand, T. Efficient fixed-point implementation of matrix-based intra prediction. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 25–28 October 2020; pp. 3364–3368.
14. Jiale, Y.; Wei, A. Fast mode decision algorithm for intra prediction in HEVC. In Proceedings of the 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12–14 June 2020; Volume 1, pp. 1018–1022.
15. Peng, W.; Ni, C.; Zhang, G.; Li, K. R-Lambda model based CTU-level rate control for intra frames in HEVC. *Multimed. Tools Appl.* **2017**, *78*, 1–15.
16. Yeh, C.-H.; Li, M.F.; Chen, M.J.; Chi, M.-C.; Huang, X.-X.; Chi, H.-W. Fast mode decision algorithm through inter-view rate-distortion prediction for multiview video coding system. *IEEE Trans. Ind. Inform.* **2014**, *10*, 594–603. [CrossRef]
17. Moscarelli, C.M.; Waskow, B.H.; Goebel, J.W.; Palomino, D.M.; Correa, G.R.; Agostini, L.V. A High-Throughput Hardware Architecture for AV1 Non-Directional Intra Modes. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2020**, *67*, 1481–1494.

18. Zhang, T.; Sun, M.; Zhao, D.; Gao, W. Fast Intra-Mode and CU Size Decision for HEVC. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *27*, 1714–1726. [[CrossRef](#)]
19. Hu, Q.; Shi, Z.; Zhang, X.; Gao, Z. Fast HEVC intra mode decision based on logistic regression classification. In Proceedings of the 2016 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB), Nara, Japan, 1–3 June 2016; pp. 1–4.
20. Tariq, J.; Kwong, S. Adaptive stopping strategies for fast intra mode decision in HEVC. *J. Vis. Commun. Image Represent.* **2018**, *51*, 1–13. [[CrossRef](#)]
21. 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](#)]
22. Huang, B.; Chen, Z.; Cai, Q.; Zheng, M.; Wu, D. Rate-Distortion-Complexity Optimized Coding Mode Decision for HEVC. *IEEE Trans. Circuits Syst. Video Technol.* **2019**, *30*, 795–809 [[CrossRef](#)]
23. Yan, Z.; Cho, S.-Y.; Welsen, S. Fast Intra Prediction Mode Decision for HEVC Using Random Forest. In Proceedings of the 2019 International Conference on Image, Video and Signal Processing, Shanghai China, 25–28 February 2019; pp. 45–49.
24. Hosseini, E.; Pakdaman, F.; Hashemi, M.R.; Ghanbari, M. A computationally scalable fast intra coding scheme for HEVC video encoder. *Multimed. Tools Appl.* **2019**, *78*, 11607–11630. [[CrossRef](#)]
25. Tian, R.; Zhang, Y.; Duan, M.; Li, X. Adaptive intra mode decision for HEVC based on texture characteristics and multiple reference lines. *Multimed. Tools Appl.* **2019**, *78*, 289–310. [[CrossRef](#)]
26. Gwon, D.; Choi, H. Relative SATD based Minimum Risk Bayesian Framework for Fast Intra Decision of HEVC. *KSII Trans. Internet Inf. Syst.* **2019**, *13*, 385–405.
27. Tariq, J. High-performance intra-mode accelerator for HEVC. *Vis. Comput.* **2019**, *35*, 1–15. [[CrossRef](#)]
28. Tariq, J. Guided filter based intra mode accelerator for HEVC. *Multimed. Tools Appl.* **2020**, *79*, 20299–20318. [[CrossRef](#)]
29. Tariq, J.; Ijaz, A. HEVC Intra Mode Selection Using Benford’s Law. *Circuits Syst. Signal Process.* **2020**, *40*, 418–437. [[CrossRef](#)]
30. Tariq, J.; Armghan, A.; Ijaz, A.; Ashraf, I. Pure intra mode decision in HEVC using optimized firefly algorithm. *J. Vis. Commun. Image Represent.* **2020**, *68*, 102766. [[CrossRef](#)]
31. Tariq, J.; Armghan, A.; Ijaz, A.; Ashraf, I. Light weight model for intra mode selection in HEVC. *Multimed. Tools Appl.* **2021**, *80*, 21449–21464. [[CrossRef](#)]
32. Tariq, J.; Armghan, A.; Alenezi, F.; Ijaz, A.; Rehman, S.; Alfalou, A.; Khan, J.A. HEVC Fast Intra-Mode Selection Using World War II Technique. *Electronics* **2021**, *10*, 985. [[CrossRef](#)]
33. Bossen, F. Common test Conditions and Software Reference Configurations. In Joint Collaborative Team on Video Coding (JCT-VC); JCTVC-F900. 2011. Available online: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKewjNpeCZmo7wAhXlb94KHficAIMQFjAAegQIBBAD&url=https%3A%2F%2Fwww.itu.int%2Fwftp3%2Fav-arch%2Fjctvc-site%2F2011_07_F_Torino%2FJCTVC-F_Notes_dH.doc&usg=AOvVaw17hNODzjtQ1Totp2B3BjVP (accessed on 18 February 2021).
34. Bjontegaard, G. Calculation of Average PSNR Differences between RD-Curves. VCEG-M33. 2001. Available online: https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUKewiXsurwm47wAhVK7GEKHRvNDLQQFjABegQIBBAD&url=https%3A%2F%2Fwww.itu.int%2Fwftp3%2Fav-arch%2Fvideo-site%2F0104_Aus%2FVCEG-M33.doc&usg=AOvVaw28nfdGxLOuM9xOYxBLc5wA (accessed on 2 March 2021).
35. Tan, T.K.; Weerakkody, R.; Mrak, M.; Ramzan, N.; Baroncini, V.; Ohm, J.; Sullivan, G.J. Video quality evaluation methodology and verification testing of HEVC compression performance. *IEEE Trans. Circuits Syst. Video Technol.* **2015**, *26*, 76–90. [[CrossRef](#)]
36. Zhao, T.; Kwong, S.; Wang, H.; Kuo, C.-C.J. H. 264/SVC mode decision based on optimal stopping theory. *IEEE Trans. Image Process.* **2012**, *21*, 2607–2618. [[CrossRef](#)]
37. Tariq, J. RD-cost as statistical inference for early intra mode decision in HEVC. *Multimed. Tools Appl.* **2019**, *78*, 16783–16801. [[CrossRef](#)]