DEBLOCKING FILTER EFFECTS IN HIGH EFFICIENCY VIDEO CODING (HEVC)

Suhas S K
Asst.Prof: Dept of Electronics and Communication Engg
South East Asian College of Engineering & Technology
Bangalore, India.
email:suhashoodi@gmail.com

Dr. Gurulingappa M Patil
Principal: JSS Academy of Technical Education
Noida U.P, India.
email:gmpatil999@gmail.com

Abstract—This paper describes the in-loop deblocking filter used in the upcoming High Efficiency Video Coding (HEVC) standard to reduce visible artifacts at block boundaries. The deblocking filter performs detection of the artifacts at the coded block boundaries and attenuates them by applying a selected filter. Compared to the H.264/AVC deblocking filter, the HEVC deblocking filter has lower computational complexity and better parallel processing capabilities while still achieving significant reduction of the visual artifacts.

Index Terms—Block-based coding, deblocking, video coding, video filtering, video processing.

I. Introduction

High efficiency video coding (HEVC) is a new video coding standard currently being developed jointly by ITU-T SG 16 Q.6, also known as the Video Coding Experts Group (VCEG), and by ISO/IEC JTC 1/SC 29/WG 11, also known as the Moving Picture Experts Group (MPEG) in the joint collaborative team on video coding (JCT-VC). The first version of the HEVC standard is planned to be finalized in January 2013, while the development of the scalable and 3-D extensions of HEVC is expected in the following years. Similar to the previous video coding standards, such as H.264/AVC, the upcoming HEVC standard is based on a hybrid coding scheme using block-based prediction and transform coding. First, the input signal is split into rectangular blocks that can be predicted from previously decoded data either by motion-compensated prediction [3] or intra prediction. The resulting prediction error is coded by applying block transforms based on an integer approximation of the discrete cosine transform, which is followed by the quantization and coding of the transform coefficients.

While H.264/AVC [2] divides a picture into fixed size macroblocks of $16 \times 16$ samples, HEVC divides a picture into coding tree units (CTU) of $16 \times 16$, $32 \times 32$ or $64 \times 64$ samples. The coding tree units can be further defined as a transform unit (TU). The size of the transforms used in the prediction error coding can vary from $4 \times 4$ to $32 \times 32$ samples, thus allowing transforms larger than in H.264/AVC, which uses $4 \times 4$ and $8 \times 8$ transforms. As the optimal size of the above-mentioned blocks depends typically on the picture content, the reconstructed picture is composed of blocks of various sizes, each block being coded using an individual prediction mode and the prediction error transform divided into smaller blocks using a quadtree structure; such a block, called a coding unit (CU), can further be split into prediction units (PUs) and is also a root for the transform quadtree. Each of the child nodes of the transform quad tree.

In a coding scheme that uses block-based prediction and transform coding, discontinuities can occur in the reconstructed signal at the block boundaries. Visible discontinuities at the block boundaries are known as blocking artifacts. A major source of blocking artifacts is the block-transform coding of the prediction error followed by coarse quantization. Moreover, in a motion-compensated prediction process, predictions for adjacent blocks in the current picture might not come from adjacent blocks in the previously coded pictures, which creates discontinuities at the block boundaries of the prediction signal. Similarly, when applying intra prediction, the prediction process of adjacent blocks might be different causing discontinuities at the block boundaries of the prediction signal.

Two approaches to reduce blocking artifacts are post-filtering and in-loop filtering. Post-filtering is not specified by the video coding standard and can be performed, e.g., in the display buffer. The implementer has a freedom to design an algorithm driven by application-specific requirements. In-loop filters operate within the encoding and decoding loops. Therefore, they need to be normative to avoid drift between the encoder and decoder.

The HEVC draft standard defines two in-loop filters that can be applied sequentially to the reconstructed picture. The first one is the deblending filter and the second one is the sample adaptive offset (SAO) that are currently included into the main profile. This paper describes the first of these two in-loop filters, the deblending filter. Depending on the configuration, SAO can be applied to the output of the deblending filtering process.

Fig. 1. 1-D example of block boundary with blocking artifact and Illustration of picture samples and horizontal and vertical block boundaries.
The deblocking filter in HEVC has been designed to improve the subjective quality while reducing the complexity. The latter consideration is important since the deblocking filter of the H.264/AVC standard, constitutes a significant part of the decoder complexity. As a result, the HEVC de-blocking filter is less complex as compared to the H.264/AVC deblocking filter, while still having the capability to improve the subjective and objective quality.

Another aspect that received significant attention in the HEVC deblocking filter design is its suitability for parallel processing. Deblocking in HEVC has been designed in a way to prevent spatial dependences across the picture, which, together with other design features, enables easy parallelization on multiple cores.

In the following sections, an overview of the HEVC deblocking filter design is provided. For more details, the reader is referred to [1], and to the corresponding input contributions to the JCT-VC. The initial deblocking filter design was adopted from [5]. The filtering decisions and operations, as described in Sections II and III, mainly result from the adoption of the contributions in [6] and [7]. For sequence and picture-level adaptivity (see Section IV) the main adopted contribution is [12]. The parallel processing capabilities, as described in Section V, mainly result from adoption of [8]–[10].

II. Filtering Decisions

A. Block Boundaries for Deblocking

As mentioned above, independent coding of blocks creates discontinuities at block boundaries. An example of a block boundary with a blocking artifact is shown in Fig. 1. Blocking artifacts can easily be noticed by the human visual system when the signal on both sides of the block boundary is relatively smooth, but are more difficult to notice when the signal shows high variation. Furthermore, if the original signal across the block boundary is subjected to higher variations, artifacts can easily be noticed by the human visual system.

The boundary strength can take one of the three possible values: 1) the block boundary is a prediction unit or transform unit boundary; 2) the boundary strength is greater than zero; and 3) variation of signal on both sides of a block boundary is below a specified threshold (see Fig. 4). When certain additional conditions (Section II-D) hold, a strong filter is applied on the block edge instead of the normal deblocking filter.

For block boundaries with an associate Bs greater than zero, and for which (1) holds, deblocking filtering is performed. There are two deblocking filtering modes in HEVC, namely, a normal filtering mode and strong filtering mode. For each block boundary of four samples in length, the deblocking filter switches between the normal and the strong filtering mode based on the local signal characteristics.

B. Boundary Strength (Bs) and Edge-Level Adaptivity

Boundary strength (Bs) is calculated for boundaries that are either prediction unit boundaries or transform unit boundaries. The boundary strength can take one of the three possible values: 0, 1, and 2. The definition of the Bs is shown in Table I.

For the luma component, only block boundaries with Bs values equal to one or two are filtered. This implies that there is typically no filtering within the static areas. This helps avoid multiple subsequent filtering of the same areas where pixels are copied from one picture to another with a residual equal to zero, which can cause oversmoothing. The difference in filtering operations between Bs equal to one and Bs equal to two is described in Section III-D.

In the case of the chroma components, only boundaries with Bs equal to two are filtered. This implies that only those block

D. Decisions Between Normal and Strong Deblocking

Whether to apply strong or normal deblocking is also determined based on the first and the fourth lines across the block boundary of four samples. Boundaries are filtered where at least one of the two adjacent blocks is intra predicted.

\[\text{TABLE I}\]
\text{Definition of Bs Values for the Boundary Between Two Neighboring Luma Blocks}

<table>
<thead>
<tr>
<th>Conditions</th>
<th>Bs</th>
</tr>
</thead>
<tbody>
<tr>
<td>At least one of the blocks is Intra</td>
<td>2</td>
</tr>
<tr>
<td>At least one of the blocks has non-zero coded residual coefficient and boundary is a transform</td>
<td>1</td>
</tr>
<tr>
<td>Absolute differences between corresponding spatial motion vector components of the two blocks are (\geq 1) in units of integer pixels</td>
<td>1</td>
</tr>
<tr>
<td>Motion-compensated prediction for the two blocks refers to different reference pictures or the number of motion vectors is different for the two blocks</td>
<td>0</td>
</tr>
<tr>
<td>Otherwise</td>
<td>0</td>
</tr>
</tbody>
</table>
C. Local Adaptivity and Filtering Decisions

If Bs is greater than zero, additional conditions are checked for luma block edges to determine whether the deblacking filtering should be applied to the block boundary or not.

As we can see from Fig. 1, a blocking artifact is characterized by low spatial activity on both sides of the block boundary, whereas there is discontinuity at the block boundary. Therefore, for each block boundary of four-sample length on the 8 x 8 sample grid that satisfies the conditions described above, the following condition is checked to decide whether the deblacking filtering is applied (see Fig. 3):

\[ \left| p_{2,0} - 2p_{1,0} + p_{0,0} \right| + \left| p_{2,3} - 2p_{1,3} + p_{0,3} \right| < 3/16 \delta \]  

where threshold \( \delta \) is defined to adjust the quantization parameter on a per-block boundary basis. Similarly, if the threshold \( \delta \) is not exceeded, the modified value \( p_{0,0} \) is obtained by clipping \( \delta \) to zero:

\[ \delta_0 = (9(q_0 - p_0) - 3(q_1 - p_1) + 8) \text{ max } 0 \]

In these cases, the normal deblocking filter operations should not modify the signal.

In the normal filtering mode for a segment of four lines (see Fig. 3), filtering operations are applied for each line. In the following, the second indices of pixels, indicating the line number, are omitted for brevity.

The filtered pixel values \( p_0 \) and \( p_0 \) are calculated as

\[ q_0 = \frac{9(q_0 - p_0) - 3(q_1 - p_1) + 8}{10} \]  

where the value of \( q_0 \) is obtained by clipping \( \delta_0 \) to zero:

\[ q_0 = (9(q_0 - p_0) - 3(q_1 - p_1) + 8) \text{ max } 0 \]

Similarly, if (6) is true, then \( q_1 \) is calculated as

\[ q_1 = q_1 + q_1 \]

where the offset values \( p_1 \) and \( q_1 \) are obtained by clipping the corresponding \( \delta_1 \) and \( q_1 \) values, which are calculated as

\[ \delta_1 = (((p_2 + p_0 + 1) >> 1) - p_1 + 0) \text{ max } 0 \]

Neglecting the clipping operation, the impulse response of the filter that corresponds to the quantization parameter of the block at position \( p_0 \) is (8 19 - 1 9 - 3)/32.

The sequence of filtering decisions for each line of pixels in the normal filtering mode is summarized in Fig. 5.

III. Filtering Operations

A. Normal Filtering Operations

When a picture contains an inclined surface (or linear ramp signal) that crosses a block boundary, the filter will be active. In these cases, the normal deblacking filter operations should not modify the signal.

In the normal filtering mode for a segment of four lines (see Fig. 3), filtering operations are applied for each line. In the following, the second indices of pixels, indicating the line number, are omitted for brevity.

If (5) is true, then each modified value \( \rho \) in each line across the block boundary is obtained by

\[ \rho_0 = \rho_0 + 0 \]

\[ \rho_0 = \rho_0 - 0 \]

where the value of \( \rho_0 \) is obtained by clipping \( \delta_0 \) to zero:

\[ \rho_0 = (9(q_0 - p_0) - 3(q_1 - p_1) + 8) \text{ max } 0 \]

Similarly, if (6) is true, then \( q_1 \) is calculated as

\[ q_1 = q_1 + q_1 \]

where the offset values \( p_1 \) and \( q_1 \) are obtained by clipping the corresponding \( \delta_1 \) and \( q_1 \) values, which are calculated as

\[ \delta_1 = (((p_2 + p_0 + 1) >> 1) - p_1 + 0) \text{ max } 0 \]

Neglecting the clipping operation, the impulse response of the filter that corresponds to the quantization parameter of the block at position \( p_0 \) is (8 19 - 1 9 - 3)/32.

The sequence of filtering decisions for each line of pixels in the normal filtering mode is summarized in Fig. 5.
The offset values for modification of pixels $q_0$, $q_1$, and $q_2$ are calculated by exchanging $q$ and $p$ in (15), (16), and (17). Impulse responses of the filters that correspond to modification of pixels $\rho_0$, $\rho_1$, and $\rho_2$ are $(1 2 2 1)/8$, $(1 1 1)/4$, and $(2 3 1 1 1)/8$, respectively, if the clipping operation is neglected.

C. Chroma Deblocking

As mentioned previously, chroma deblocking is only performed when $B_s$ is equal to two. In this case, no further deblocking decisions are done. Only pixels $\rho_0$ and $\rho_1$ are modified as in (7) and (8). The deblocking is performed with the $C$ value, which is obtained by clipping the following $\delta_C$ offset value:

$$\delta_C = (((\rho_0 - q_0) \ll 2) + p_1 - q_1 + 4) \gg 3$$

which corresponds to filtering by the filter with the impulse response of $(1 4 4 -1)/8$.

D. Clipping

To prevent excessive blurriness, deblocking filtering is done on a signal after QP-dependent clipping. Clipping is applied to the $\delta$ values after their calculation and before modification of the pixel values. The values used in filtering are obtained by clipping the $\delta$ values to the range $-c$ to $c$ as in (19). Clipping provides more adaptivity to deblocking filtering. The clipping is applied by performing the following operations:

$$\delta = \text{Min}(\text{Max}(-c, \delta), c)$$

where the value of $c$ is equal to $IC\cdot (n)$ for $\rho_0$ and $q_0$, and $IC\cdot (n)/2$ for $p_1$ and $q_1$ in the case of normal filtering. In the case of strong filtering, $c$ is set equal to $2IC\cdot (n)$. Variable $n$ is equal to QP when both blocks adjacent to the boundary are intra predicted and QP+2, if one of the blocks is intra predicted ($Bs = 2$).

The dependence of the parameter $\beta$ on QP is illustrated in Fig. 6. The blocking artifacts strength is generally greater for intra predicted blocks. Therefore, larger modifications of pixel values are allowed for intra-blocks than those for inter-blocks by using the clipping value $IC\cdot (QP + 2)$ for block boundaries with $Bs$ equal to 2.

The filtered pixel values $\rho_0$, $q_0$, $p_1$ and $q_1$ for normal filtering and $\rho_0$ and $q_0$ for chroma deblocking are also clipped to stay in the range defined by the bit depth $N$

$$p = \text{Min}(\text{Max}(0, p), 2^N - 1).$$

The clipping operation is described in Section III-D. Neglecting the clipping operation, the impulse response of this filter is $(3 7 9 -3)/16$. The offset value $\delta_0$ corresponds to the deviation of the signal at the sides of the block boundary from a perfect ramp. The offset is zero if the signal across the block boundary forms a ramp.

Furthermore, the deblocking filtering is applied to the row or column of samples across the block boundary, if and only if the following expression holds:

IV. Sequence and Picture Level Adaptivity

Since different video sequences have different characteristics, deblocking strength can be adjusted on a sequence and even on a picture basis.

As mentioned earlier, the main sources of blocking artifacts are block transforms and quantization. Therefore, blocking artifact severity depends, to a large extent, on the quantization parameter QP. Therefore, in the deblocking filtering decisions, the QP value is taken into account. Thresholds $\beta$ and $IC$ depend on the average QP value of two neighboring blocks with common block edge [13] and are typically stored in corresponding tables. The dependence of these parameters on QP is shown in Figs. 6 and 7.

The parameter $\beta$ controls what edges are filtered, controls the selection between the normal and strong filter, and controls how many pixels from the block boundary are modified in the normal filtering operation. One can observe that the value of $\beta$ increases with QP. Therefore, deblocking is enabled more frequently at high QP values compared to low QP values, high QP values correspond to coarse, and low QP values correspond to fine quantization. One can also see that the deblocking operation is effectively disabled for low QP values by setting one or both of $\beta$ and $IC$ to zero.

The parameter $IC$ controls the selection between the normal and strong filter and determines the maximum absolute value of modifications that are allowed for the pixel values for a certain QP for both normal and strong filtering operations. This helps adaptively limit the amount of blurriness introduced by
the deblocking filtering.

The deblocking parameters $\alpha$ and $\beta$ provide adaptivity according to the QP and prediction type. However, different sequences or parts of the same sequence may have different characteristics. It may be important for content providers to change the amount of deblocking filtering on the sequence or even on a slice or picture basis.

Compared to H.264/AVC, the complexity of the deblocking filter has been significantly reduced in HEVC due to several factors that are described in this section. Performing deblock-ing on a grid of $8 \times 8$ samples as opposed to a grid of $4 \times 4$ samples in H.264/AVC reduces the number of deblocking operations by a factor of two. Deblocking of the chroma component in the 4:2:0 format is also performed on the grid of $8 \times 8$ samples. Furthermore, the chroma blocks are filtered only in cases when one of the adjacent blocks is intra predicted. This decreases the amount of chroma filtering further for inter-coded slices. Filtering on an $8 \times 8$ sample grid may potentially lead to reduction in subjective quality. However, since the number of $4 \times 4$ blocks in the picture for HEVC is generally lower than that for H.264/AVC and $4 \times 4$ blocks in HEVC are usually used in the areas with higher temporal or spatial activity, applying filtering on an $8 \times 8$ sample grid is a tradeoff between computational complexity and subjective quality.

Another source of complexity reduction in HEVC deblocking is related to the transform and prediction unit size. In H.264/AVC, the largest transform size is $8 \times 8$, whereas the largest prediction unit size is $16 \times 16$ samples, i.e., a macroblock. However, in HEVC the largest transform size is $32 \times 32$ and the largest prediction unit size is $64 \times 64$ samples. This additionally reduces the average amount of operations (although not necessarily for the worst case) since deblocking is never performed inside these large blocks.

Deblocking in HEVC has been designed to prevent spatial dependences of the deblocking process across the picture. There is no overlap between the filtering operations for one block edge, which can modify at most three pixels from the block edge, and the filtering decisions for the neighboring parallel block edge, which involves at most four pixels from the block edge. Therefore, any vertical block edge in the picture can be deblocked in parallel to any other vertical edge. The same holds for horizontal edges. Note, however, that sample values modified by deblocking of vertical block boundaries are used as the input for deblocking of horizontal block boundaries.

Another advantage of the highly parallelizable HEVC de-blocking filter is that it provides enough cycle margins to enable a combination of the deblocking filter and SAO in the same building block in hardware implementations. In a typical architecture, the HEVC deblocking filter only consumes from 84 to 88 cycles per $16 \times 16$ block, which is less than half of the typical 200 cycles per $16 \times 16$ block cycle budget (for a 1080p@120 fps video running at 250 MHz clock rate) [14]. Combining the deblocking filter and SAO in the same building block is beneficial in terms of hardware area cost, since SAO and deblocking can share the same memory interface, in contrast to having separate building blocks and memory interfaces for SAO and deblocking.

Since deblocking in HEVC is computationally less intensive and more parallelizable than in H.264/AVC, it can be said that the HEVC deblocking is much less of a bottleneck when implementing a video decoder. The deblocking in HEVC is a better tradeoff among coding efficiency (i.e., subjective and objective quality), throughput, and implementation complexity when compared to the H.264/AVC design.

VI. Results

This section demonstrates the objective and subjective impact of deblocking filtering. Tables II–V show the BD-rate resulting from disabling the deblocking filtering for various configurations used in the HEVC standardization [16]. These configurations are all-intra where only intra prediction is used.

**TABLE II**

Average Bit Rate Increase at the Same Quality by Disabling the Deblocking Filter for the All-Intra Configuration

<table>
<thead>
<tr>
<th></th>
<th>Y</th>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class A</td>
<td>1.9%</td>
<td>4.2%</td>
<td>3.7%</td>
</tr>
<tr>
<td>Class B</td>
<td>1.7%</td>
<td>4.5%</td>
<td>5.1%</td>
</tr>
<tr>
<td>Class C</td>
<td>0.9%</td>
<td>3.7%</td>
<td>4.3%</td>
</tr>
<tr>
<td>Class D</td>
<td>0.7%</td>
<td>3.0%</td>
<td>3.4%</td>
</tr>
<tr>
<td>Class E</td>
<td>2.1%</td>
<td>7.4%</td>
<td>8.8%</td>
</tr>
<tr>
<td>Class F</td>
<td>0.6%</td>
<td>1.9%</td>
<td>1.8%</td>
</tr>
<tr>
<td>Overall</td>
<td>1.3%</td>
<td>4.0%</td>
<td>4.4%</td>
</tr>
</tbody>
</table>

**TABLE III**

Average Bit Rate Increase at the Same Quality by Disabling the Deblocking Filter for the Random-Access Configuration

<table>
<thead>
<tr>
<th></th>
<th>Y</th>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class A</td>
<td>3.6%</td>
<td>2.1%</td>
<td>1.9%</td>
</tr>
<tr>
<td>Class B</td>
<td>3.2%</td>
<td>1.9%</td>
<td>1.9%</td>
</tr>
<tr>
<td>Class C</td>
<td>2.1%</td>
<td>1.5%</td>
<td>1.9%</td>
</tr>
<tr>
<td>Class D</td>
<td>1.5%</td>
<td>1.1%</td>
<td>1.2%</td>
</tr>
<tr>
<td>Class F</td>
<td>1.2%</td>
<td>0.9%</td>
<td>0.9%</td>
</tr>
<tr>
<td>Overall</td>
<td>2.6%</td>
<td>1.6%</td>
<td>1.7%</td>
</tr>
</tbody>
</table>

**TABLE IV**

Average Bit Rate Increase at the Same Quality by Disabling the Deblocking Filter for the Low-Delay Configuration

**TABLE V**

Average Bit Rate Increase at the Same Quality by Disabling the Deblocking Filter for the Low-Delay P-Frame Configuration

<table>
<thead>
<tr>
<th></th>
<th>Y</th>
<th>U</th>
<th>V</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class B</td>
<td>4.9%</td>
<td>2.5%</td>
<td>2.7%</td>
</tr>
<tr>
<td>Class C</td>
<td>2.6%</td>
<td>1.5%</td>
<td>2.1%</td>
</tr>
<tr>
<td>Class D</td>
<td>1.6%</td>
<td>1.4%</td>
<td>0.8%</td>
</tr>
<tr>
<td>Class E</td>
<td>6.2%</td>
<td>7.8%</td>
<td>9.0%</td>
</tr>
<tr>
<td>Class F</td>
<td>1.7%</td>
<td>1.0%</td>
<td>0.4%</td>
</tr>
<tr>
<td>Overall</td>
<td>3.3%</td>
<td>2.5%</td>
<td>2.7%</td>
</tr>
</tbody>
</table>

As a positive number in the tables indicates an increased bit rate at the same quality, the HEVC deblocking filter leads to an average bit rate reduction of 1.3%–3.3% at the same quality, dependent on the configuration. For certain sequences, more than 6% bit rate reduction is achieved.
The deblocking filter in the upcoming HEVC standard improves both the subjective and objective quality of the coded video sequences, while being less computationally expensive than the deblocking filter in H.264/AVC. The decrease in computational complexity is achieved by reconsidering a number of tools. The HEVC deblocking filtering operations can also be easily performed in parallel on multiple processors, which is important for coding and decoding higher resolution video sequences.

VII. Conclusion

BIBLIOGRAPHY

Suhas S K. is a research scholar in VTU RRC Belgaum. received his BE (Electronics and Communication Engineering) from South East Asian College of Engineering and Technology, Visvesvaraya Technological University, Belgaum, Karnataka, India, in the year 2011 and completed his M.Tech (Digital Electronics) from Srinivas Institute of Technology, Mangalore, affiliated to Visvesvaraya Technological University, Belgaum, Karnataka, India, in the year 2013 and since then he is actively involved in teaching and research and has Five years of experience in teaching. He is pursuing PhD (ECE) from VTU. At present he is working as Assistant Professor in South East Asian college of Engineering and Technology, Bangalore Affiliated to Visvesvaraya Technological University. His area of interest is in the field of Signal Processing, Bio Medical Signal Processing, Communication System

Dr. Gurulingappa M Patil received his BE (E&C) from Sri. Jayachamarajendra College of Engineering from Mysore University, Karnataka, India, in the year 1984 and completed his Master of Engineering from University of Roorkee, presently Indian Intitute of Technology, Roorkee, in 1990. and since then he is actively involved in teaching and research and has Thirty three years of experience in teaching. He obtained his Ph. D. from Department of Biomedical Engineering, University College of Engineering, Osmania University, Hyderabad, in February, 2011. He worked at various capacities in affiliated University Engineering Colleges. For Visvesvaraya Technological University he worked as a Chairman for Board of Examiners, and member of Board of Studies etc. At present he is working as Principal in JSS Academy of Technical Education, Noida U. P. He has good exposure in the field of signal Processing, Bio Medical Signal Processing. He has published more 38 research papers in the refereed international journals and presented contributed research papers in refereed international and national conferences. He is a member of IEEE, Bio Medical Society of India, Instrument Society of India, Indian Society for Technical Education, International Association of Engineers, MIAENG, Hongkong