IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 30, 2020, accepted December 15, 2020, date of publication December 24, 2020,
date of current version January 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2020.3047073

Epipolar Plane Image-Based Lossless and Near-Lossless Light Field Compression

M. UMAIR MUKATI, (Graduate Student Member, IEEE),
AND SØREN FORCHHAMMER, (Member, IEEE)
DTU Fotonik, Department of Photonics Engineering, Technical University of Denmark, 2800 Kongens Lyngby, Denmark

Corresponding author: M. Umair Mukati (mummu@fotonik.dtu.dk)

This work was supported by the EU's H2020 ITN Programme through the MSCA Grant Agreement (RealVision) under Grant 765911.

ABSTRACT Recent advances in the computer technology has enabled many theoretical ideas in computer vision to become practical also within camera technology. Light field technology aims at replacing the traditional camera design by acquiring the intensity of light rays to realize advanced applications in media and medical industries. The abundant amount of information poses high processing, transmission and storage requirements. Therefore, a need for efficient compression has gained researchers attention in this area. We present EPIC, an efficient lossless compression scheme for light field images which extends CALIC by exploiting light field structure. We have shown that the proposed scheme achieves around 6% improvement in compression performance relative to HEVC-lossless anchor with a significant reduction in the encoding time. We have also extended the proposed method for the near-lossless case which achieves similar rate-distortion performance as HEVC-lossy anchor in addition to a high encoding efficiency and a guaranteed-error bound.

INDEX TERMS CALIC, efficient compression, light field, lossless, near-lossless.

I. INTRODUCTION

A light field (LF) is a set of intensities of the light rays emanating from a scene and captured by a LF camera. This data can be used to render images with different optical settings such as focal lengths, aperture sizes and view-points. Apart from these capabilities, recently LF has been used in various practical applications such as novel view rendering [1], video stabilization [2], seeing through foreground occlusion [3], 3D in-vivo imaging of middle ear [4], etc. To support image rendering with different optical settings, the LF should be available either at a server or the decoder. A challenge with LFs is the enormous data size, which is a burden on the storage device and the transmission channel. For example, a LF image captured by LYTRO Illum (a commercial graded LF camera) after decoding acquires 174.6MB of storage space, for a spatial resolution of 434 × 625 pixels with 225 angular dimensions. Therefore, LF compression is a well-known problem in the research community.

Due to the property of natural scenes and the structure of the LF camera, each feature in the scene is captured by several views of the LF, which results in highly redundant data. Thus supporting the possibility to achieve high compression ratios in contrast to that of traditional images. This redundancy can be easily observed as linear patterns or edges in Epipolar Plane Images (EPI), an LF representation. In the proposed scheme, we utilize the EPI representation to efficiently exploit correlation in the LF structure with much simpler processing.

LFs have found most of their applications in the media industry which can tolerate distortion. Therefore most of the compressors in the literature are based on lossy compression schemes. Some applications can not tolerate distortion artifacts introduced by compression algorithms, such as medical diagnosis, where the distortion may remove or obscure very significant and life-saving information, while possibly introducing misleading artifacts into the image [5]. The lossless LF compression schemes in the literature offer decent compression ratios at the cost of hours of encoding time [6] and/or complex computation [7]. For this reason, we have recently proposed an efficient lossless LF compression scheme, EPIC [8], which extends CALIC to exploit the LF structure for enhanced compression performance. In this article, we extend this approach to gain further efficiency in compression ratio (5.3% reduction), processing time (71.3% reduction[1]) and computational requirements. To achieve further compression

The associate editor coordinating the review of this manuscript and approving it for publication was Shiqi Wang.

[1] Optimized EPIC-I requires 1.73 min. to encode a LF.

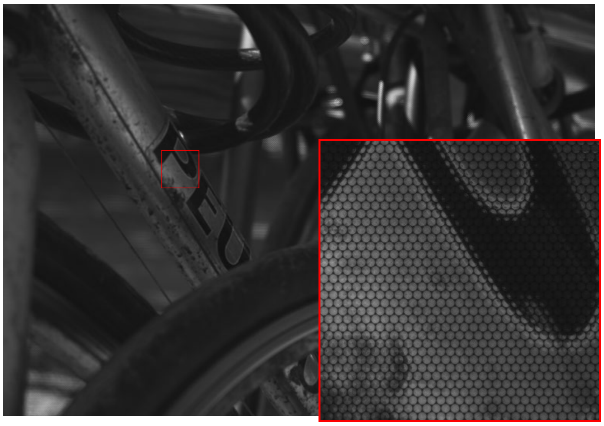


FIGURE 1. Raw lenslet image captured by the Lytro Illum camera. In the magnified view, the hexagonal grid can be observed, where each circle is the projection of the rays approaching from the main-lens on the photo-sensor.

gains, the proposed scheme has also been modified for the near-lossless case. We have demonstrated that by using the proposed approach we not only achieve better compression performance than relevant schemes but also a reconstructed LF that has better quality and max. absolute error.

The rest of this article is organized in the following manner. In Section II, we present a brief overview of relevant lossless image and LF compression schemes. In Section III, we describe the proposed approach for lossless LF compression. In Section IV, we evaluate the performance of the proposed approach by comparing with the relevant lossless compression schemes. Finally, we present the conclusion in Section V.

II. LOSSLESS COMPRESSION OF LF IMAGES

One possible approach to losslessly compress LF images is to utilize state-of-the-art lossless image compression schemes. Generally, these schemes exploit the correlation through prediction based on the local neighborhood and adaptively encodes the prediction error through context modeling. JPEG-LS [9] is popular due to its efficiency as it utilizes only three-pixel causal neighborhood for prediction and error modeling. On the other hand, CALIC [10] offers higher compression than JPEG-LS utilizing a six-pixel causal neighborhood for prediction and a complex context modeling of error. To encode the LF with video encoders such as AVC and HEVC, it has been rearranged as a Pseudo Video Sequence through zig-zag or spiral scanning in [11] and [12], respectively. These encoders can be configured for lossless compression of the LF images.

The lossless compression schemes designed to exploit the LF structure can be divided into two main categories to compress either raw lenslet images (Fig. 1) or the rectified LF utilizing decoders such as in [13]. Among the initial schemes for lossless LF image compression, a method was proposed by Perra [14] to compress lenslet images. This scheme exploits the inherent redundancy in the lenslet structure by estimating the displacement between pixels that maximizes correlation between adjacent blocks of pixels to optimize the prediction process. Another lenslet compression scheme was proposed in [15], where the pixels inside each microlens are vectorized through spiral scanning, subsequently the elements of the vector are predicted using a sparse model based on the neighboring microlens pixels and the previously decoded pixels of the current microlens. In [16], Voronoi cells are aligned to the micro-lens array and a predictor is designed to exploit inter-view (within micro-lens) and spatial correlation (among neighboring micro-lenses). These compression schemes generally requires an additional LF rectification step at the decoder. In addition, the compression gains are also limited due to the unrectified structure of the lenslet image. Therefore, the compression of the rectified LFs has gained popularity in the research community.

Helin *et al.* [17], [18] sequentially encodes rectified LF views following a spiral scan order as in [12]. Before encoding, each view is segmented utilizing a corresponding quantized disparity map and an image segmentation technique. Then each segment is predicted using a sparse model based on the dictionary formed by the corresponding segments in the reference views. On the other hand, [19] utilized a simpler adaptive prediction scheme. The small prediction residuals are encoded using a context-based entropy encoder, where the context is formed by applying image segmentation to the reference image. Whereas, the large residuals are encoded using a Golomb Rice algorithm. A threshold to distinguish between small and large residuals is estimated by minimizing the estimated codelength. Another set of schemes utilizes minimum rate predictors (MRP) to find an optimal predictor that can minimize the total code length [6], [20], [21]. In [21], several LF representations are utilized to analyze the coding performance of MRP, and it was concluded that pseudo video and EPI representations gave a better performance. This work is extended in [6] to exploit the LF structure by utilizing a 4D neighbourhood for prediction. The MRP-based lossless compression scheme proved to give an optimal compression performance, however the computation involved in finding an optimal predictor makes the encoding process extremely slow in both cases.

In [22], Schiopu *et al.* proposed to use a Convolutional Neural Network-based architecture utilizing six causal neighboring micro-lens blocks to predict the pixels in the current micro-lens, whereas the prediction residuals are encoded following CALIC's encoding scheme. This method achieves better performance than CALIC, however, the network has around a million parameters which makes the prediction process computational intensive for a regular PC. In [7], the authors achieve lossless LF compression in two stages. In the first stage, a network MPS-CNN synthesize the full LF image from a sub-sampled LF encoded using REP-CNN [23]. Based on the six previously decoded and the two synthesized micro-lens blocks, the second stage predicts the pixels in the current micro-lens using another network PMSP-CNN. Finally, the prediction residuals are encoded using the method described in [22]. There are more than 3 million parameters used in these two stages, in addition to it, the sub-sampled LF is initially encoded using REP-CNN which further adds to the computational complexity.

In the literature, it can be noticed that there is a strong trade-off between coding efficiency and computational complexity when it comes to lossless LF compression. In our previous work for lossless LF compression [8], we have addressed this problem by modifying CALIC to efficiently exploit correlation present in the LF structure by introducing effective prediction and context formation techniques. We have introduced an EPI Slope-based predictor, which estimates the slope of the EPI line and subsequently predicts the intensity using 1-D quadratic interpolation. Additionally, the difference in the two EPI-based predictions was incorporated in the context formation process. This approach outperformed schemes utilizing general lossless image compression methods as described earlier and gave a coding performance comparable to [17] with a reasonable amount of encoding time. In this article, we further improve the coding and time efficiency by introducing changes in the predictor design and context formation process.

III. PROPOSED METHOD - EPIC

CALIC is an efficient lossless coding scheme [10], which encodes and decodes the pixels of the image in raster scan order. Based on the number of distinct intensities of the neighboring pixels, one out of the two modes, i.e. Binary and Continuous modes, is selected as the mode of operation to encode a given pixel. Continuous mode is the most important part of CALIC, which has three main processing blocks, i.e., intensity prediction using a non-linear predictor, error energy estimation for adaptive entropy coding and context-based bias cancellation from the prediction errors. CALIC was designed to achieve optimal performance with traditional images, however, it can not achieve optimal performance for LF images due to its incapability to fully exploit the LF structure. In our approach, we present a list of modifications in the continuous mode of CALIC (Fig. 2) to achieve enhanced compression performance with the LF images while keeping the computational complexity at a minimum.

With an assumption that the light rays captured by the LF camera travels through a lossless medium at some time instant, the LF can be parametrized using a 4D coordinate system [24], [25]. Two parallel planes separated by some distance f is commonly used to represent the intersecting light ray. Hence, a light ray can be parameterized by $L(s, t, u, v)$; (s, t) denotes the point of intersection on the first plane (i.e. camera plane), and (u, v) denotes the point of intersection on the second plane (i.e. image plane). In this representation, the point (u, v) (spatial coordinates) marks the spatial location for the intersecting light ray, while the point (s, t) (angular coordinates) determines its angular orientation with respect to the point (u, v). A fifth dimension is commonly used to represent the intensities at different color channels. A rectified LF has some well-known representations such as multi-perspective images, macro-pixel array and EPI. These representations can be obtained by sub-sampling and/or combining the 4D coordinates in different forms.

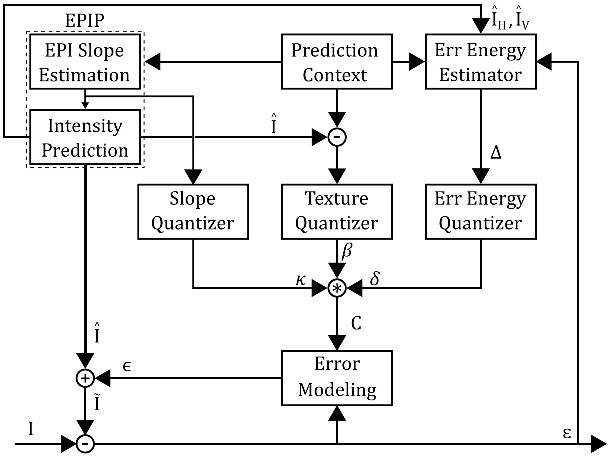


FIGURE 2. Block diagram of the continuous mode of CALIC after the proposed modifications.

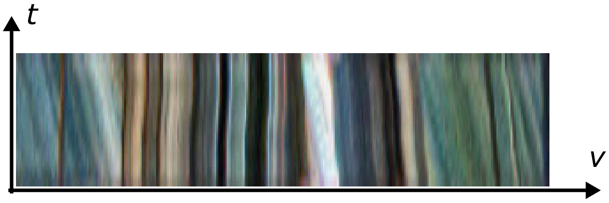


FIGURE 3. Sample of a horizontal EPI extracted from the Bikes LF in [26].

As shown in Fig. 3, EPI representation consists of linear patterns or edges, whose slope encodes the depth of the corresponding feature. This characteristic also reflects the redundancy in the LF structure which can be efficiently exploited using only a small neighborhood of pixels for prediction and context formation. The EPI representation of a LF $L(s, t, u, v)$ is obtained by fixating an angular coordinate and a spatial coordinate of the same orientation. Although, several EPI representations can be obtained as described in [27], for simplicity, we consider only two EPI representations; the first is formed by fixing vertical components of angular and spatial coordinates of the LF, i.e., $I_{s^*,u^*}(t, v) = L(s^*, t, u^*, v)$ and the second is formed by fixing horizontal coordinates, i.e., $I_{t^*,v^*}(s, u) = L(s, t^*, u, v^*)$.

A. EPI-BASED PREDICTION

CALIC utilizes a non-linear predictor to estimate the intensity at the current pixel (X) using a causal neighborhood. This predictor is designed to work well with regular images with a high efficiency. However, it can not exploit the nature of EPI. In this article, we propose an EPI-based predictor (EPIP) to replace CALIC's predictor for improved intensity prediction by slightly increasing the complexity, which takes into consideration the EPI structure by estimating the slope of the EPI line[2] passing through X followed by intensity prediction using a quadratic interpolator based on the estimated slope.

[2] Here, EPI line is referred to the line passing through X in the direction of pattern or the edge along which the intensity remains same.

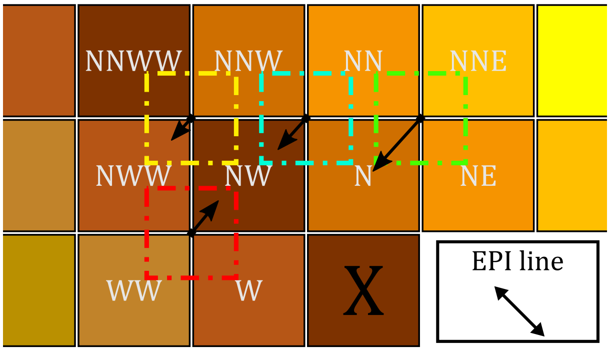


FIGURE 4. Neighboring pixels of *X* are divided using four overlapping blocks of 2 × 2 pixels, represented by dot-dashed squares of different colors.

+1	-1
+1	-1

Horizontal

+1	+1
-1	-1

Vertical

FIGURE 5. Filters used to estimate X and Y-components of the gradient vector. Horizontal and vertical filters are named as F_x and F_y, respectively.

1) SLOPE ESTIMATION

The slope of the EPI line can be estimated by finding horizontal and vertical gradient vectors using filters such as Roberts, Sobel operator etc. As shown in Fig. 4, due to the raster scan order of CALIC, only the pixels in the causal neighborhood are available, therefore, the gradients can not be estimated directly. Instead, by assuming that disparity maps are locally smooth, we estimate the gradient vector at X by blending the gradient vectors of the four neighboring regions (contrary to the set of 3 neighboring regions in [8]), as shown in Fig. 4. Each region $r \in R$ is composed of a 2×2 block of pixels B_r, which is annotated by a dot-dashed square of unique color in the figure. The gradient vector for each region r is estimated by applying 2×2 filters F_x and F_y, as shown in Fig. 5, to B_r as follows:

$$\nabla_{r_x} = \sum_n F_x(n) B_r(n),$$
$$\nabla_{r_y} = \sum_n F_y(n) B_r(n),$$

where n denotes the index of all the elements in the block and the filter. ∇_{r_x} and ∇_{r_y} represents X and Y-components of the gradient vector ∇_r, respectively. Note that the estimated gradient vectors should be perpendicular to the EPI line.

The angle of the estimated gradient vectors may differ from one another due to the presence of noise, occlusion or a feature at another depth in some regions. As demonstrated in Fig. 4, some vector directions can be completely opposite to other vectors due to the non-linear pattern variation. Therefore, for robust slope estimation the gradient vectors of the neighboring regions must be carefully incorporated.

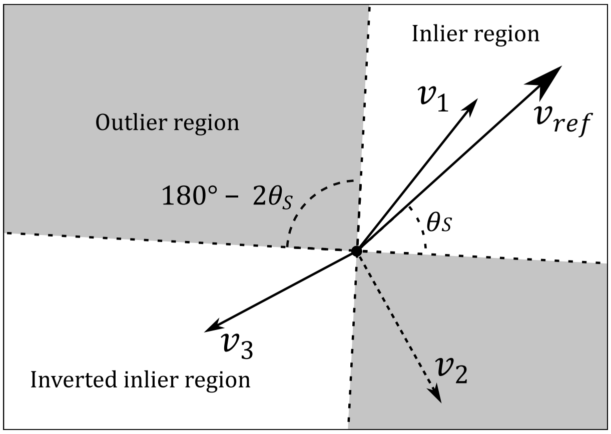


FIGURE 6. Estimated gradient vectors are visualized in the Cartesian coordinates. In this example, the vector ∇_2 lying in the outlier region will be filtered out for robust EPI slope estimation.

In contrast to [8], we remove any outlying vector, out of the four estimates, by applying a constraint on the angle θ_r of the gradient vector ∇_r. The gradient vector with the largest magnitude is selected as the reference vector ∇_{ref}. Any gradient vector which does not meet the following criteria is removed, as illustrated in Fig. 6:

$$\left|\tan\left(\theta_r - \theta_{ref}\right)\right| < |\tan\theta_S|, \tag{1}$$

where θ_{ref} is the angle of the reference vector ∇_{ref} and θ_S is used to set the angular range to define the inlier region keeping θ_{ref} as its center. The optimal estimates for the slope is achieved by setting $\theta_S = 45°$. The criteria in (1) can be simplified as follows:

$$\begin{aligned} \left|\tan\left(\theta_r - \theta_{ref}\right)\right| &< 1, \\ ||\nabla_r \times \nabla_{ref}|| &< \left|\nabla_r \cdot \nabla_{ref}\right|. \end{aligned} \tag{2}$$

Previously in [8], we averaged the estimated angles for the final estimate. Here, a robust estimate for the gradient vector ∇, perpendicular to the EPI line, is achieved through the weighted sum of the inlying gradient vectors,

$$\begin{aligned} \nabla &= \sum_{r_+} \omega_{r_+} \nabla_{r_+}, \\ \omega_{r_+} &= \left|\nabla_{r_+}\right| \cdot \operatorname{sgn}\left(\nabla_{r_+} \cdot \nabla_{ref}\right) \end{aligned}$$

where, $r_+ \in R_+$, a subset of R which satisfy the condition in (2) and ω_{r_+} is the weight assigned to the gradient vector of region r_+. Here $\left|\nabla_{r_+}\right|$ is the magnitude of the gradient vector ∇_{r_+}. Assigning more weight to the vector with larger magnitude resulted in better slope estimation. A region of this form $\left(\begin{smallmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{smallmatrix}\right) = \left(\begin{smallmatrix} a & b \\ b & a \end{smallmatrix}\right)$, cannot be used to distinguish if the angle of the gradient vector is 45° or $-45°$. Instead of estimating the gradient magnitude using the gradient components, defining the gradient magnitude as $|\nabla_r| = |r_{11} - r_{22}| + |r_{12} - r_{21}|$ can avoid the reliance of slope estimation on the gradient vector using such regions. To detect whether a vector lies in the inlier region or the inverted inlier region, the sign of the dot product can be used, which is negative when the difference in angle is more than 90°. It is utilized in the weight equation to invert a gradient vector if it lies in the opposite inlier region. Finally, the slope of the EPI line can be determined by dividing the X-component from the Y-component of the gradient vector ∇.

2) INTENSITY PREDICTION

Ideally in a non-occluded region, the intensity remains the same along the EPI line. We utilize this property to predict the intensity using the slope estimated in the previous section. As illustrated in Fig. 7, consider a distance axis whose origin is at X and it is perpendicular to the EPI slope. Then the intensity of a neighboring pixel should be the same as that of its projection on the distance axis, which is denoted as I_j. The distance from the origin to the projection of the neighboring pixel positions d_j can be calculated as follows:

$$d_j = \frac{\nabla \cdot V_j}{\|\nabla\|_2^2} \tag{3}$$

where $j \in \{N, W, NW, NE\}$ only, to avoid large errors in prediction in the presence of distortion in the texture variation and V_j is the xy-displacement of the neighboring pixel j with respect to X (e.g. $V_{NE} = [+1, -1]$). The basis vector of the distance axis is defined as the estimated gradient vector ∇ in (3), since both the vectors are perpendicular to the EPI line.

Given the intensity of the pixels I_j and their projected distances d_j, the intensity at X can be interpolated with the polynomial curve fitting. We choose to fit the quadratic curve to the set of points to capture any non-linearity in the texture variation. We avoided using higher order polynomials to prevent overfitting. Therefore, the relationship between the projected distance and the intensity of a neighboring pixel can be written as:

$$I_j = \alpha d_j^2 + \beta d_j + \gamma \tag{4}$$

In matrix notation, (4) can be written as follows:

$$\begin{pmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{pmatrix} = \begin{pmatrix} d_1^2 & d_1 & 1 \\ d_2^2 & d_2 & 1 \\ d_3^2 & d_3 & 1 \\ d_4^2 & d_4 & 1 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix}$$

$$I = DB$$

$$B = \left(D^T D\right)^{-1} D^T I \tag{5}$$

where, the coefficients in **B** is the least square solution for (4).

The origin of the distance axis lies at X, therefore the y-intercept (γ) in (4) provides the intensity at X. Equation (5) can be simplified to find γ by:

$$\gamma = \frac{(V_1 \times V_2) \cdot V_3}{(V_1 \times V_2) \cdot V_4}, \tag{6}$$

where,

$$V_1 = (p_4, p_3, p_2), \qquad V_2 = (p_3, p_2, p_1),$$
$$V_3 = (q_2, q_1, q_0), \qquad V_4 = (p_2, p_1, p_0).$$
$$p_n = \sum_j d_j^n, \qquad q_n = \sum_j d_j^n I_j.$$

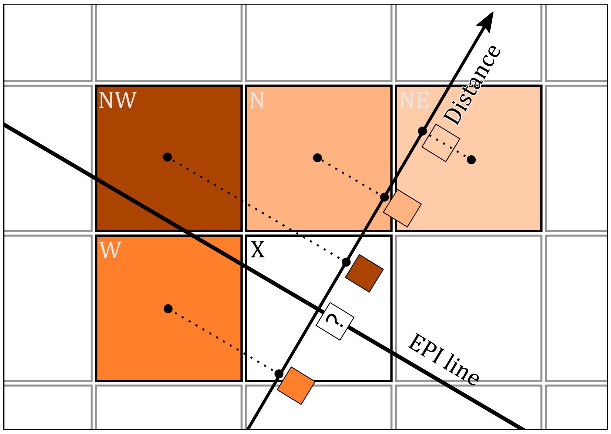


FIGURE 7. **The displacement of the neighboring pixels with respect to *X* are projected on the distance axis. Along with the projected distances and the intensities of the neighboring pixels, the intensity at *X* is predicted using a quadratic interpolator.**

It can be proven that γ in (6) is independent of scaling the distance axis, therefore, the division with the squared-norm of the gradient vector $\|\nabla\|_2^2$ in (3) can be omitted. In the case, when distances are either only positive or only negative, the intensity of the nearest neighbour in terms of the projected distance is taken as the prediction. For robust prediction, to account for the smoothness in the region, a weighted average is taken between the prediction γ and an average of intensities I_j, to assign more weight to the prediction when the local activity is high.

$$\hat{I} = \omega_p \gamma + \left(1 - \omega_p\right) \frac{1}{n} \sum_j I_j, \tag{7}$$

where,

$$\omega_p = \frac{\nu}{\nu + c},$$

here, $\nu = d_h + d_v$, the sum of horizontal variation (d_h) and vertical variation (d_v) in the neighborhood. d_h and d_v are estimated the same way as in CALIC using the neighborhood of X in EPI representation. c is a constant for smoothness in prediction. We use $c = 3$, which gives the optimal performance.

3) ENHANCED PREDICTION WITH MULTI-ORIENTATION EPIs

The prediction process explained above is sensitive to the presence of noise and occlusions, which can affect the accuracy of the slope estimation, thus resulting in incorrect prediction. However, this process only utilizes an EPI representation for prediction. The predictions from two EPI representations can be incorporated to have a robust prediction to prevent the performance degradation due to such uncertainties.

Under ideal conditions, occlusion is the main source of error in slope estimation. A vertical occluder cannot occlude scene points captured by $I_{t,v}$, similarly, a horizontal occluder cannot occlude points captured by $I_{s,u}$ [27]. Thus, the prediction from an EPI representation having its orientation closer to the direction of occlusion should be assigned more weight in the averaged prediction. We have observed that a region in one EPI representation which is more effected by occlusion compared to the region in another EPI has higher local variation. Hence, we assign more weight to the prediction of an EPI representation having lower local activity in the following manner.

$$\hat{I} = \frac{\nu_V \hat{I}_H + \nu_H \hat{I}_V}{\nu_H + \nu_V}$$
$$\nu_H = d_h^H + d_v^H \quad \nu_V = d_h^V + d_v^V \tag{8}$$

where, ν_H and ν_V represents the local variations in the neighborhood of X in $I_{s,u}$ and $I_{t,v}$, respectively.

The authors of CALIC observed that the variance of prediction errors strongly correlates to the smoothness of the neighboring region. Equation (8) also considers this observation as it assigns more weight to the prediction having higher smoothness in the neighboring region.

B. CODING CONTEXT FORMATION

The relationship of the prediction error with the statistics of its surroundings can be exploited to further reduce redundancy in the LF by conditioning the prediction error on some correlated source. This correlated source is formally denoted as the ‘‘Error Energy Estimator’’ in CALIC. CALIC relies on two elements to model this function, i.e., the smoothness of the region and the previous prediction errors corresponding to the adjacent neighboring pixels. It is observed that the variance of prediction error is negatively correlated to the smoothness of the neighboring region, therefore, the local variation can serve as the magnitude of error energy. The magnitude of previous prediction errors is also a vital estimator of the error energy, since the errors in the prediction tends to occur consecutively in the nearby region.

We propose to improve this model by utilizing the difference in the predictions as an estimator of error energy. From the two EPI representations, two predictions are obtained using EPIP for X. A small difference between these predictions reflects the confidence in accuracy of the final prediction, while the large difference points to the fact that at least one of the predictions is incorrect, thereby increasing the probability of making an error in the prediction. Combining all these elements, we write our error energy model Δ as:

$$\Delta = a\,(\nu_H + \nu_V) + b\,\epsilon + c\left|\hat{I}_H - \hat{I}_V\right| \tag{9}$$

where, $\epsilon = |\epsilon_N| + |\epsilon_W| + \frac{1}{2}\,|\epsilon_{NW}| + \frac{1}{2}\,|\epsilon_{NE}|$, the sum of the previous prediction errors belonging to the spatial neighborhood of X. The coefficients a, b and c balances the contribution of each estimator in Δ. In an offline design process, these coefficients are optimized to minimize the squared difference between Δ and the magnitude of the prediction error $|\epsilon|$. For (9), $a = \frac{1}{6}$ and $b = c = 1$ returns an optimal error energy estimator.

Therefore, by conditioning the predictor error on Δ, its entropy can be reduced. For time and space efficiency, CALIC instead conditions the prediction errors on quantized error energy levels. As in CALIC, we have quantized Δ using an 8-level quantizer. The bin edges of the quantizer are optimized such that the conditional entropy

$$-\sum_{d=[1..7]} \sum_{e} p(\epsilon) \log p\left(\epsilon | q_d \le \Delta < q_{d+1}\right) \tag{10}$$

is minimized. For this purpose, a large dataset is prepared consisting of (ϵ, Δ) pairs by utilizing LFs from the EPFL dataset. Dynamic programming is used to find the optimal bin edges, as described in [28]. The 8-level quantizer having the following bin edges

$$q_1 = 1, q_2 = 4, q_3 = 7, q_4 = 11,$$
$$q_5 = 18, q_6 = 29, q_7 = 58,$$

minimizes the expression in (10). The prediction error is encoded through adaptive entropy coding by conditioning the quantized error energy defined above.

C. CONTEXT-BASED ERROR BIAS CANCELLATION

Conditioning the prediction error on the error energy model effectively reduces its entropy. However, it can not exploit the dependence of the prediction error on the high order structure of the neighboring region, such as, the texture pattern. Typically, the possible combinations of such contexts are very large which leads to the problem of context dilution [29], i.e., low number of samples for prediction error resulting in unreliable estimates of the error probability for each context $p(\epsilon|C)$. Instead of estimating the probabilities of the prediction error, CALIC estimates the expectation of the prediction error under each context $E\{\epsilon|C\}$ which can be reliably generated in this situation. The conditional expectation is then used to remove any bias in the prediction $\hat{I}$.

CALIC utilizes a compound context to model the prediction error, which is composed of 144 binary texture contexts β and 4 error energy contexts δ. As can be seen in Fig. 7, EPIP utilizes the intensities from the four neighboring pixels to interpolate the intensity at X. As shown in Fig. 8, the EPI line can pass through any one of the four octants. Then, in each of the four cases, there will be a fixed number of intensities at the left and the right sides of the distance axis, e.g., if the EPI line is passing through octant Q_1, then there will be one sample on the left side of the distance axis, while the remaining three samples will be on the right side of the distance axis. The prediction is based on the least squares solution, therefore, the quality of the prediction may vary from one octant to another depending on the number of supports (intensities) across the distance axis. A special case arises when the EPI line is passing through Q_4, then all the supports will be at one side of the distance axis depending on the gradient direction. Earlier in [8], for this situation, we have avoided the extrapolation in EPIP by assigning the intensity of the nearest pixel in terms of its distance from the origin of the distance axis. Therefore, the quality is expected to be worse compared to the cases when the line is passing through other octants. This proves that the prediction quality or in another sense the

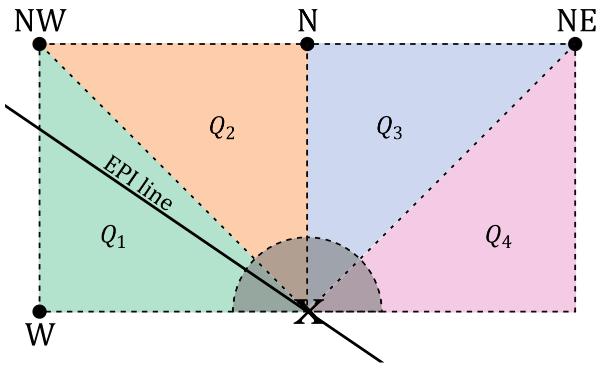


FIGURE 8. The angle of EPI line can range from 0° – 180°. This range is subdivided into four regions (octant). Each octant signifies the quality of the prediction using EPIP due to different number of projected neighboring pixels of the two sides of the distance axis.

error energy is also a function of octant from which the line is passing. Hence, we improve the compound context of CALIC by incorporating a 2-bit octant index as context κ, forming a compound context with 2304 possible combinations, which is reasonable increase from 576 contexts, given the enormous size of LFs.

Two independent compound contexts can be used to robustly estimate expectation of the prediction error under each compound context using $I_{s,u}$ and $I_{t,v}$, respectively.

$$E\{\epsilon|C(\delta,\beta,\kappa)\} = \frac{S_H(\delta,\beta,\kappa)+S_V(\delta,\beta,\kappa)}{N_H(\delta,\beta,\kappa)+N_V(\delta,\beta,\kappa)} \qquad (11)$$

where, $E\{\epsilon|C(\delta,\beta,\kappa)\}$ is the expectation of the prediction error under the compound context $C(\delta,\beta,\kappa)$. S_H and S_V accumulates the prediction error under $C(\delta,\beta,\kappa)$, and N_H and N_V contains the number of occurrence of $C(\delta,\beta,\kappa)$ for $I_{s,u}$ and $I_{t,v}$, respectively. These variables are rescaled when N_H or N_V reaches a maximum value as described in [10].

D. CODING SEQUENCE

EPIC will encode/decode the LF views sequentially. Based on the available views at the decoder, the views are processed with different modes of EPIC, as shown in Fig. 9. To decode a view, EPIC requires its four neighboring views (two from an adjacent horizontal and two from an adjacent vertical direction). If only two horizontal views are available at the decoder then the view is decoded using EPIC-H, similarly the view is decoded using EPIC-V if only two vertical views exists. EPIC-H predicts the intensity and models the error based on $I_{s,u}$ only, while EPIC-V relies on $I_{t,v}$. In the case when two views are not available in either of the two directions, then the views are independently decoded using standard CALIC.

To encode a LF, our method first encodes the four bottom-left views independently using CALIC, followed by encoding of views from left-to-right in the bottom two rows using EPIC-H and views from bottom-to-top in the left two columns using EPIC-V. The rest of the views are encoded following a raster scan order starting from the bottom-left position of the remaining views.

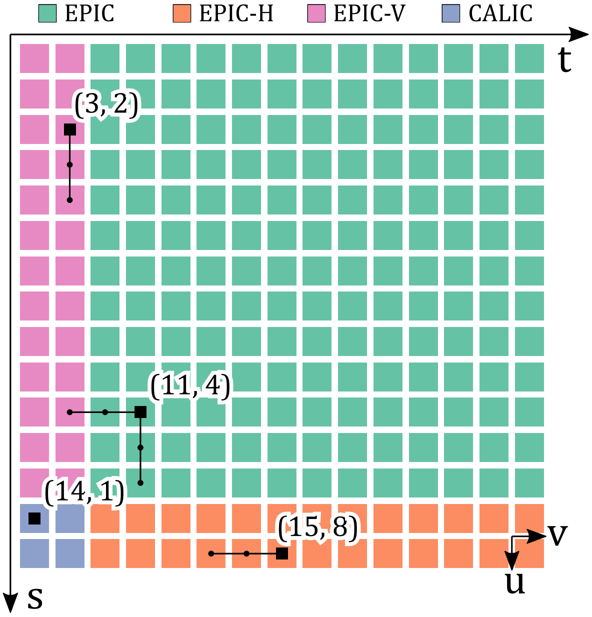


FIGURE 9. The LF views are indexed by coordinates (s, t), while the pixels in the corresponding view are indexed by (u, v). The views are represented as square blocks, while their color highlights which mode of EPIC was used to encode it. Examples for encoding four views are shown in the figure. The small black square represents the current view being encoded and the connecting lines points to the views required to encode them.

For better illustration, the examples are provided in Fig. 9, reflecting modes of operation for coding different views. For example, due to the absence of required neighboring views in horizontal and/or vertical direction, the view (14, 1) is encoded independently using CALIC, whereas, for view (3, 2) the mode EPIC-V is triggered which utilizes the decoded views at (4, 2) and (5, 2) in its vertical neighbors, while the view (11, 4) is encoded with the main approach of this article EPIC utilizing four neighboring views, i.e., (11, 3), (11, 2), (12, 4) and (13, 4).

$I_{t,v}$ is a vertical EPI representation, which is typically formed by stacking a vertical slice at index v from all the views with index t, such that the slice from the top-most view is at the top-level of the stack. As the slope of the line passing through X in each EPI representation encodes the depth of the scene, the two slopes should always be the same. It means that these lines will ideally pass through the same octant each time. Since, the quality of the prediction is influenced by the octant from which the EPI line passes through, the quality of the averaged prediction will be a function of octants. To eliminate the dependence of quality on the octant, a quick solution is to flip the EPI line vertically, then the EPI line which was passing through the octants Q_1, Q_2, Q_3 or Q_4 will pass through the octants Q_4, Q_3, Q_2 or Q_1, respectively. To realize this solution, it can be noticed from the examples in Fig. 9 that the bottom two views are utilized instead of the upper two views to form the causal neighborhood in $I_{t,v}$ representation.

(a) Bikes (b) Danger de Mort (c) Flowers (d) Stone Pillars (e) Vespa (f) Ankylosaurus
(g) Desktop (h) Magnets (i) Fountain & Vincent (j) Friends (k) Color Chart (l) ISO Chart

FIGURE 10. **Thumbnails of LFs utilized for analysis from the EPFL dataset [26].**

IV. PERFORMANCE EVALUATION

The proposed method is designed for coding dense LFs (having thin baseline between consecutive views). Due to the continuous nature of EPI lines in such LFs, it becomes possible to estimate the slope of the EPI line using a small neighborhood of pixels. To analyze the performance of the proposed scheme, we have utilized dense LF datasets from [26], [30]. The LFs provided in the EPFL dataset [26] are captured using the Lytro Illum B01 camera. The raw data captured by the camera is rectified using the LF toolbox provided by [13]. The rectified LF has an angular resolution of 15 × 15 and a spatial resolution of 434 × 625. For compatibility with the baseline methods used in this article, we have utilized 8-bit LF representation. The camera captures color in RGB color space. The proposed method does not exploit inter-channel redundancy, therefore we thought, it is more appropriate to analyse the performance on some decorrelated color space, such as YCbCr. Therefore, the decoded 16-bit RGB LF is converted to an 8-bit YCbCr LF using the function provided in the JPEG Pleno LF common testing condition document [31]. The resulting uncompressed LF image requires 174.6MB of disk space.

The HCI LF dataset [30] consists of synthetic LFs and their corresponding disparity maps. The provided disparity maps can be utilized as a ground truth to evaluate the performance of the slope estimation in EPIP. From the dataset, the **sideboard** LF, consists of significant disparity and texture variations, hence it is used to analyze the performance of EPIP.

For the experiments, we have utilized an HP ZBook 15 G4 with Intel Core i7-7700HQ processor clocked at 2.80GHz and 16 GB RAM. The proposed approach is implemented with C++, roughly requiring a minute to encode a LF. As CALIC, the proposed scheme is symmetric, therefore a similar amount of memory and time is required to decode a LF. The memory requirements of the proposed scheme is extremely low, as the prediction is based on the few neighboring pixels requiring only to buffer few lines from the adjacent four views at a time. In contrast to CALIC, EPIC utilizes the sophisticated predictor EPIP (Sec. III-A) and richer context, which increases encoding time by 20% as can be observed from Table 1.

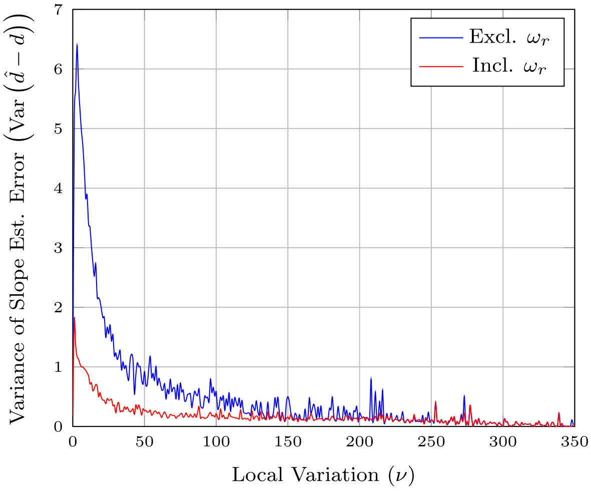


FIGURE 11. **Evaluation of the effect of using ω_r in terms of slope estimation accuracy for reference vector selection.**

A. EPIP PERFORMANCE ANALYSIS

The performance of EPIP is dependent on two important parts; the slope estimation of the EPI line and the intensity prediction based on quadratic interpolation. In an ideal case, in order for the quadratic interpolator to predict accurately, the slope must be precisely estimated. Thus, slope estimation is a crucial step in EPIP.

The slope of the vector perpendicular to the EPI line represents the disparity d in that region. A local neighborhood with an apparent feature is a major factor in deriving the accuracy of the slope estimation. In other words Fig. 11 reveals that, as the neighboring region gets smoother, the estimated slope $\hat{d} = \nabla_y / \nabla_x$ becomes unreliable. To prevent this from causing a poor intensity prediction, this problem is treated by assigning more weight to the average of the neighboring intensities in (7). However, there is still some contribution of the poor prediction in the final estimate. In dense LFs, the disparity between the consecutive views ranges from -4 to 4 for LFs captured by the Lytro Illum camera and ranges from

TABLE 1. Comparison of compression performance of different lossless image compression schemes. Average compression rate measured in bits-per-pixel calculated for EPFL LF dataset (in 24-bit YCbCr format).

	General lossless schemes					LF lossless schemes			
Sequence	X264	X265	HEVC	JPEG-LS	CALIC	CMS	MMRP	EPIC-I	Proposed
Bikes	7.70	7.72	7.33	7.80	7.50	7.41	5.50	7.17	6.84
Danger de Mort	7.14	7.13	6.83	7.57	7.21	7.04	5.04	6.77	6.36
Flowers	7.34	7.33	6.96	7.30	7.13	7.13	5.19	6.87	6.52
Stone Pillars	7.38	7.52	7.23	7.66	7.37	7.06	5.42	7.11	6.75
Vespa	6.98	7.10	6.70	7.02	6.84	6.74	5.07	6.57	6.25
Ankylosaurus	7.78	7.82	7.46	7.74	7.50	7.16	5.74	7.38	7.01
Desktop	5.98	6.02	5.65	5.94	5.69	5.78	4.42	5.71	5.42
Magnets	7.73	7.78	7.46	7.70	7.47	7.13	5.79	7.31	6.95
Fountain & Vincent	8.42	8.47	7.95	8.31	7.99	8.03	6.02	7.82	7.45
Friends	6.61	6.77	6.35	6.82	6.61	6.20	4.76	6.40	6.04
Color Chart	8.00	8.09	7.49	7.88	7.70	7.39	5.90	7.74	7.29
ISO Chart	8.20	8.18	7.77	8.11	7.83	7.72	6.00	7.64	7.26
Average bitrate (bpp)	7.44	7.49	7.10	7.49	7.24	7.07	5.41	7.04	6.68
Encoding time (min.)	0.15	12.03	53.36	0.59	0.81	6.53	15.6hr	1.73	1.01

*Average time in minutes to encode a LF.

-1.5 to 1.5 for synthetic LFs in the HCI dataset. In principle, the slope should lie within these bounds, but as we approach toward the outer views of the LF the effect of lens distortion becomes significant causing a violation to this bound. Hence, as a remedy for noise and distortion, we favor the gradient vector, out of the four gradient vectors, obtained in the slope estimation process, closer to the zero disparity by using the following weighting function:

$$\omega_r = \frac{\nabla^2_{r_x}}{\nabla^2_{r_x} + \nabla^2_{r_y}} \tag{12}$$

where, ω_r becomes one, when the disparity or slope of the normal vector to the EPI line reaches zero $\left(|\nabla^2_{r_y}|/|\nabla^2_{r_x}| \approx 0\right)$. Therefore, instead of assigning the largest gradient vector as the reference vector, we use the following equation to choose the reference vector:

$$\text{ref} = \underset{r \in R}{\arg\max}\, (|\nabla_r|\, \omega_r)$$

The above equation utilizes the magnitude of the gradient vector $|\nabla_r|$ and a weight function ω_r as a bias to favor the selection of the gradient vector closer to the zero disparity, unless the magnitude of the gradient vector having higher absolute disparity is relatively large. It can be observed from Fig. 11 that reduction in error variance is achieved in smoother regions using ω_r (12), demonstrating the effectiveness of this solution.

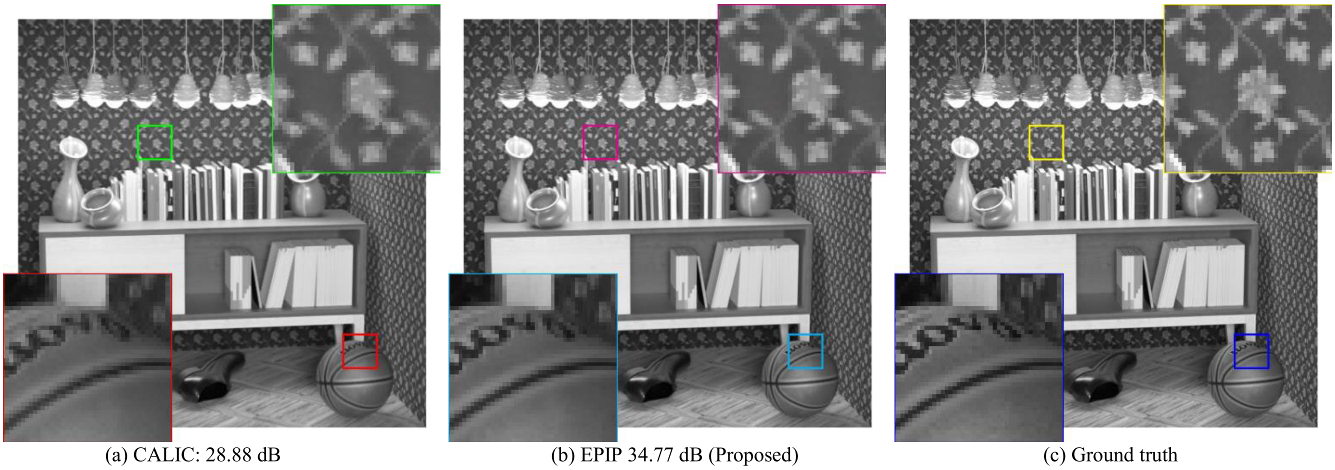
To demonstrate the superiority of prediction using EPIP over the non-linear predictor of CALIC, we have designed an experiment in which CALIC's predictor utilizes the same neighboring pixels from both the EPI representations, to generate two predictions, which are then fused together utilizing (8). It can be visualized from Fig. 12, that EPIP is able to reproduce sharp edges and texture in the scene with higher accuracy compared to the CALIC's predictor, thus, EPIP improves the prediction quality by around 5.89 dBs.

B. LOSSLESS COMPRESSION

To evaluate the performance of the proposed scheme, it is compared with lossless image and video compression schemes, such as HEVC [32], AVC [33], JPEG-LS [9] and CALIC [10], categorized as "General lossless schemes" in this article. The proposed scheme is also compared with the category of lossless compression schemes designed for LF images, i.e. Context Modeling of Subaperture images (CMS) [19], Multi-reference Minimum Rate Predictors (MMRP) [6] and the previous version of EPIC (EPIC-I) [8].

For HEVC, two kinds of reference implementations are available. Based on JPEG Pleno's recommendation [31], the X.265 encoder [34], used to generate the anchors for HEVC, is designated as *X265*. The latest implementation of HEVC (HM) is also used to report the performance of HEVC and is referred to as *HEVC* in this article. The libx264 implementation provided with open-source ffmpeg library [35] is used to report the performance for AVC and is denoted by *X264*. The LFs can be passed to these encoders under different representations, i.e. lenslet, pseudo video sequence or as stack of EPIs [21]. The authors in [21] reported that HEVC achieves the best performance with the pseudo video sequence representation of LF. Therefore, the views of the 8-bit YCbCr LF obtained earlier, are rearranged into a sequence of frames through spiral scanning, then stored in the yuv file to be processed using one of the above encoders.

To obtain optimal LF compression performance of *CALIC* and *JPEG-LS*, an EPI representation is used [21]. In this case, the LF is stored by vertically stacking all the horizontal EPIs


(a) CALIC: 28.88 dB

(b) EPIP 34.77 dB (Proposed)

(c) Ground truth

FIGURE 12. Performance comparison of pixel-wise prediction using EPI representation between (a) CALIC's predictor and (b) EPIP, the proposed predictor. The ground truth is provided in (c).

in the LF, such that the resulting image has 97650 × 625 pixels. CALIC is limited to encode a single channel image, therefore each channel in YCbCr is encoded separately.

To encode the LF, these encoders are configured as follows:

- X264: ffmpeg v2.8.15, constant rate factor: 0 and preset: very-slow.
- X265: HEVC v3.4, constant rate factor: 0, preset: very-slow, lossless flag: enabled and profile: MAIN444-8.
- HEVC: HM reference software, v16.22, configuration file: *encoder_randomaccess_main_rext.cfg*. Special configurations: CostMode: lossless, QP: 0, TransquantBypassEnable: 1, CUTransquantBypassFlagForce: 1, InputChromaFormat: 444 and FastSearch: 1.
- JPEG-LS: JPEG v3, lossless with scan interleaved mode (-ls 0) and bypass colour transformation (-c) enabled.
- CALIC: executable available online [36], max. absolute error: 0.

In the YCbCr color space, the nature of the color components is different from the luma component, i.e. the color channels are relatively smooth compared to the luma channel. Therefore, to yield further compression using the proposed scheme, we redefine the quantization levels in Sec. III-B for the color components as:

$$q_1 = 1, q_2 = 4, q_3 = 6, q_4 = 8,$$
$$q_5 = 11, q_6 = 16, q_7 = 27.$$

In Table 1, the compression performance of the general lossless schemes are presented. Utilizing the same LF representation (EPI), the proposed scheme achieves 8.3% and 12.1% improvement in compression, but with a slightly higher encoding time, in comparison with CALIC and JPEG-LS, respectively. Among the frame-based lossless compression schemes, X264 proved to be more computationally efficient than X265 and HEVC. The compression performance of HEVC is around 5% better than X264 and X265, but it is ∼300× and ∼4× slower in encoding a LF, respectively. On the other hand, the proposed scheme achieves better compression performance than all the three schemes (X264, X265 and HEVC). It achieves 6.3% improvement in compression compared to HEVC, while being ∼53× faster. The proposed scheme reduces the bit-rate by 11.3% in comparison to the fastest X264 encoder at the cost of ∼4× higher encoding time.

We compare the performance of the proposed and the other lossless LF compression schemes, to assess the performance of the proposed scheme among the approaches exploiting the LF structure. For this purpose, the schemes CMS [19], MMRP [6] and EPIC-I [8] are utilized. Since EPIC is based on a highly efficient lossless compression scheme, CALIC, it is much simpler in computation than CMS and MMRP. Therefore, it is ∼6× and ∼936× faster than CMS and MMRP, respectively. Furthermore, it achieves 5.8% higher compression than CMS. MMRP achieves the highest compression among all the approaches, which exploits redundancy within LF structure using Minimum Rate Predictors, at the cost of hours (> 15hrs) of encoding time for a single LF. However, our scheme achieves outstanding compression in reasonable amount of encoding time. In contrast to EPIC-I [8], the new method improves the compression by 5.4% through the improvements in prediction and context modeling. The new method is also more efficient in space and time compared to the previous one, due to the improvement in the efficiency of the predictor and buffering few lines of views at a time.

HEVC, MMRP and CMS are asymmetric coding schemes, since the decoder do not have complex prediction optimization blocks which highly reduces the decoding time. Therefore, these coding schemes are feasible for broadcasting applications, where the data is encoded once and decoded several times. Whereas, to decode the bitstreams generated by JPEG-LS, CALIC and the proposed scheme,

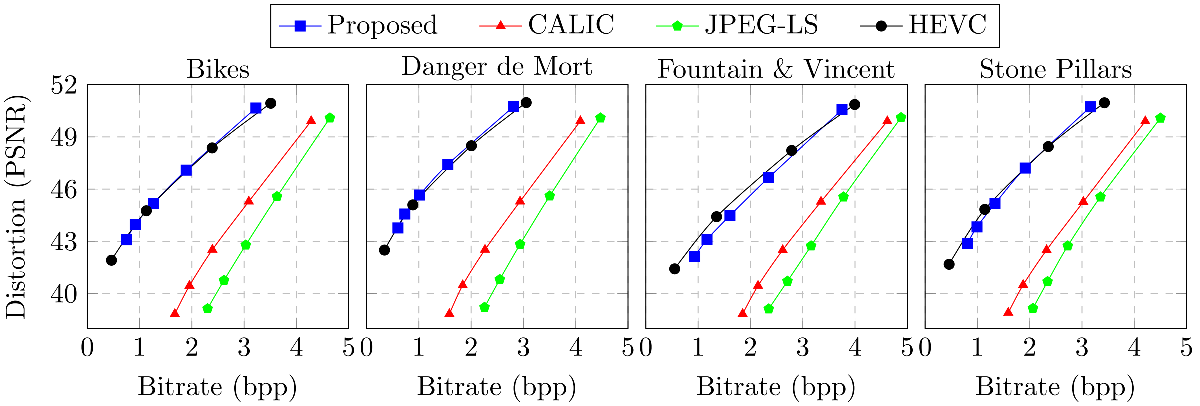


FIGURE 13. Rate-distortion performance comparison of near-lossless compression schemes using $\sigma \in \{1, 2, 3, 4, 5\}$ and lossy mode of HEVC with $QP \in \{7, 10, 15, 20\}$.

the decoder needs to replicate the prediction and context formation process at the encoder, which depends on the previously decoded pixels, therefore, the decoding time is fairly much the same as the encoding time.

C. NEAR-LOSSLESS COMPRESSION

Due to the uncertainties in the LF structure, it can be ideally compressed losslessly bounded by its entropy. In other words, the uncertainties become a bottleneck in achieving higher compression. Compromising on the constraint of lossless compression, uncertainties can be avoided up to some extent, thus helping to achieve higher compression.

The proposed lossless method for LF compression can be extended to near-lossless coding by applying uniform quantization to the prediction error to reduce the error energy, hence reducing the entropy. As in JPEG-LS, the near-lossless case is linked to a parameter σ which specifies the max. absolute error in the final reconstruction (This is identical to the inf-norm of the reconstruction error). This parameter σ can be used to set the tradeoff between the compression ratio and the reconstructed quality. The prediction error ϵ can be quantized as follows:

$$Q[\epsilon] = \left\lfloor \frac{\epsilon + \sigma}{2\sigma + 1} \right\rfloor$$

To analyze the performance of the proposed approach in near-lossless mode, it is compared with the near-lossless mode of CALIC and JPEG-LS with the same configuration as described for lossless modes but now applied for the near-lossless configuration with σ. In CALIC, σ can be defined by passing Max. Absolute Error as the input argument, while $-m$ flag is used to set σ for JPEG-LS. Besides the anchors utilizing near-lossless compression schemes, a comparison with HEVC is also provided to assess the rate-distortion performance of the proposed scheme in the near-lossless mode. In this scenario, HEVC is configured to encode with very low QPs to achieve high quality reconstruction that is comparable to the quality obtained using near-lossless modes. HEVC does not set bounds on the absolute error as near-lossless schemes. Although, it is possible to gain a higher rate-distortion performance using the proposed scheme by violating the condition of max. absolute error, this comparison will lead to discern the effectiveness of the proposed scheme even with this bound. Here, the HEVC encoder utilizes HM reference software, v16.22 and *randomaccess_main_rext* configuration and lossy mode.

TABLE 2. The max. absolute error $\hat{\sigma}$ of HEVC for quantization parameters used in Fig. 13.

QP / Seq.	7	10	15	20
Bikes	7	12	22	31
Danger de Mort	6	11	20	38
Fountain & Vincent	8	11	28	39
Stone Pillars	8	10	18	37

Four LF images, i.e., I01 Bikes, I02 Danger de Mort, I09 Fountain & Vincent and I04 Stone Pillars from the EPFL dataset are utilized to evaluate the rate-distortion performance of the described schemes in the near-lossless case. As expected due to improved prediction and enhanced context formation, it can be observed from Fig. 13 that, EPIC outperforms CALIC and JPEG-LS for all LF sequences.

From Fig. 13, it can be observed that the proposed EPIC has a similar performance as HEVC. With an increase in σ, the prediction process in EPIC gets affected by the higher error energy in the neighboring pixels, thus the reduction of EPIC's performance in comparison to HEVC with an increase of σ is evident. HEVC gains higher quality at the cost of significant increase in encoding time, e.g. to increase the quality by decreasing QP from 20 to 7, the encoding time roughly increases from 30 minutes to 90 minutes. The max. absolute error in case of HEVC is reported in Table 2. It can be observed that the measured $\hat{\sigma}$ for HEVC varies across different sequences which suggests that the value is highly dependent on the scene complexity. Additionally, the proposed method does not only offer a guarantee of an error-bound but also a similar quality reconstruction of LF in a reasonable amount of time. Furthermore, it may be observed that even the $\hat{\sigma}$ of HEVC at the lowest QP exceeds all error maxima for the near-lossless modes.

V. CONCLUSION

We have proposed an efficient lossless compression scheme for dense LF images which exploits the LF structure by modifying CALIC. The proposed scheme takes advantage from the simple nature of EPIs for efficient prediction and context formation. We have demonstrated that the proposed method provides good lossless compression performance, better than most of the relevant schemes (roughly 6% improvement compared to HEVC-lossless), while remaining highly efficient in terms of computational time (53 times faster than HEVC-lossless) and memory requirements. The effectiveness is also demonstrated for the near-lossless case, as the proposed method is superior both in required rate and quality for the reconstructed LF image in comparison to relevant schemes. It turns out to achieve comparable compression performance as HEVC in the high quality region, while being highly efficient in computational time and maintaining a low max. absolute error.

ACKNOWLEDGMENT

The authors would like to express their thanks to Xiaolin Wu for providing the CALIC source code and some useful conversation that helped in improving the previously proposed design of EPIC in terms of efficiency and compression performance.

REFERENCES

[1] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, N. K. Kalantari, R. Ramamoorthi, R. Ng, and A. Kar, "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–14, Jul. 2019.

[2] B. M. Smith, L. Zhang, H. Jin, and A. Agarwala, "Light field video stabilization," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 341–348.

[3] Y. Wang, T. Wu, J. Yang, L. Wang, W. An, and Y. Guo, "DeOccNet: Learning to see through foreground occlusions in light fields," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 118–127.

[4] N. Bedard, T. Shope, A. Hoberman, M. A. Haralam, N. Shaikh, J. Kovačević, N. Balram, and I. Tošić, "Light field otoscope design for 3D *in vivo* imaging of the middle ear," *Biomed. Opt. Exp.*, vol. 8, no. 1, pp. 260–272, 2017.

[5] R. Ansari, E. Ceran, and N. D. Memon, "Near-lossless image compression techniques," *SPIE Visual Commun. Image Process.*, vol. 3309, pp. 731–742, Jul. 1998.

[6] J. M. Santos, P. A. A. Assuncao, L. A. D. S. Cruz, L. M. N. Tavora, R. Fonseca-Pinto, and S. M. M. Faria, "Lossless compression of light fields using multi-reference minimum rate predictors," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2019, pp. 408–417.

[7] I. Schiopu and A. Munteanu, "Deep-learning-based macro-pixel synthesis and lossless coding of light field images," *APSIPA Trans. Signal Inf. Process.*, vol. 8, no. 17, p. e20, 2019.

[8] M. U. Mukati and S. Forchhammer, "EPIC: Context adaptive lossless light field compression using epipolar plane images," in *Proc. Data Compress. Conf. (DCC)*, Mar. 2020, pp. 43–52.

[9] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Process.*, vol. 9, no. 8, pp. 1309–1324, Aug. 2000.

[10] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Commun.*, vol. 45, no. 4, pp. 437–444, Apr. 1997.

[11] F. Dai, J. Zhang, Y. Ma, and Y. Zhang, "Lenselet image compression scheme based on subaperture images streaming," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 4733–4737.

[12] A. Vieira, H. Duarte, C. Perra, L. Tavora, and P. Assuncao, "Data formats for high efficiency coding of Lytro-Illum light fields," in *Proc. Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2015, pp. 494–497.

[13] D. G. Dansereau, O. Pizarro, and S. B. Williams, "Decoding, calibration and rectification for lenselet-based plenoptic cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1027–1034.

[14] C. Perra, "Lossless plenoptic image compression using adaptive block differential prediction," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 1231–1234.

[15] I. Tabus and P. Helin, "Microlens image sparse modelling for lossless compression of plenoptic camera sensor images," in *Proc. 25th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2017, pp. 1907–1911.

[16] A. Miyazawa, Y. Kameda, T. Ishikawa, I. Matsuda, and S. Itoh, "Lossless coding of light field camera data captured with a micro-lens array and a color filter," in *Proc. Int. Workshop Adv. Image Technol. (IWAIT)*, Jan. 2018, pp. 1–4.

[17] P. Helin, P. Astola, B. Rao, and I. Tabus, "Sparse modelling and predictive coding of subaperture images for lossless plenoptic image compression," in *Proc. 3DTV-Conf., True Vis. Capture, Transmiss. Display 3D Video (DTV-CON)*, Jul. 2016, pp. 1–4.

[18] P. Helin, P. Astola, B. Rao, and I. Tabus, "Minimum description length sparse modeling and region merging for lossless plenoptic image compression," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 1146–1161, Oct. 2017.

[19] I. Schiopu, M. Gabbouj, A. Gotchev, and M. M. Hannuksela, "Lossless compression of subaperture images using context modeling," in *Proc. 3DTV Conf., True Vis. Capture, Transmiss. Display 3D Video (3DTV-CON)*, Jun. 2017, pp. 1–4.

[20] J. M. Santos, P. A. A. Assuncao, L. A. da Silva Cruz, L. Tavora, R. Fonseca-Pinto, and S. M. M. Faria, "Lossless light-field compression using reversible colour transformations," in *Proc. 7th Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2017, pp. 1–6.

[21] J. M. Santos, P. A. A. Assuncao, L. A. da Silva Cruz, L. M. N. Tavora, R. Fonseca-Pinto, and S. M. M. Faria, "Lossless coding of light field images based on minimum-rate predictors," *J. Vis. Commun. Image Represent.*, vol. 54, pp. 21–30, Jul. 2018.

[22] I. Schiopu and A. Munteanu, "Macro-pixel prediction based on convolutional neural networks for lossless compression of light field images," in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 445–449.

[23] I. Schiopu and A. Munteanu, "Residual-error prediction based on deep learning for lossless image compression," *Electron. Lett.*, vol. 54, no. 17, pp. 1032–1034, Aug. 2018.

[24] M. Levoy and P. Hanrahan, "Light Field Rendering," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn.*, 1996, p. 31–42.

[25] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*, 1996, pp. 43–54.

[26] M. Rerabek and T. Ebrahimi, "New light field image dataset," in *Proc. 8th Int. Conf. Qual. Multimedia Exper.*, 2016.

[27] H. Sheng, P. Zhao, S. Zhang, J. Zhang, and D. Yang, "Occlusion-aware depth estimation for light field using multi-orientation EPIs," *Pattern Recognit.*, vol. 74, pp. 587–599, Feb. 2018.

[28] X. Wu, "Optimal quantization by matrix searching," *J. Algorithms*, vol. 12, no. 4, pp. 663–673, Dec. 1991.

[29] M. J. Weinberger, J. J. Rissanen, and R. B. Arps, "Applications of universal context modeling to lossless compression of gray-scale images," *IEEE Trans. Image Process.*, vol. 5, no. 4, pp. 575–586, Apr. 1996.

[30] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4D light fields," in *Computer Vision—ACCV*. Cham, Switzerland: Springer, 2017, pp. 19–34.

[31] *JPEG PLENO—Light Field Coding Common Test Conditions*, Standard ISO/IEC JTC 1/SC29/WG1, Vancouver, BC, Canada, 2018.

[32] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[33] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[34] MulticoreWare. *X265*. Accessed: Sep. 7, 2020. [Online]. Available: https://www.videolan.org/developers/x265.html

[35] F. Developers. *FFMPEG Tool (Version 2.8.15) [Software] 2000–2018*. Accessed: Sep. 7, 2020. [Online]. Available: http://ffmpeg.org/
[36] *CALIC Executable Link*. Accessed: Sep. 7, 2020. [Online]. Available: https://www.ece.mcmaster.ca/~xwu/calicexe

M. UMAIR MUKATI (Graduate Student Member, IEEE) received the B.E. degree in industrial electronics from the NED University of Engineering and Technology, Pakistan, in 2014, and the M.S. degree in electronics with specialization in computational imaging from Istanbul Medipol University, Turkey, in 2017. He is currently pursuing the Ph.D. degree with the Coding and Visual Communication Group, Department of Photonics Engineering, Technical University of Denmark. During his master's study, he worked on improving the spatial resolution and virtually extending the aperture size of light field image. He has been a part of Coding and Visual Communication Group since 2018. As an Early Stage Researcher with the EU MSCA ITN RealVision, his goal is to encode light field image data for flexible access and processing. His research interests include exploring unconventional methods for light field coding and compression, novel view synthesis, and enhancing light field capabilities.

SØREN FORCHHAMMER (Member, IEEE) received the M.S. degree in engineering and the Ph.D. degree from the Technical University of Denmark, Kongens Lyngby, Denmark, in 1984 and 1988, respectively. Since 1988, he has been with the DTU Fotonik, Department of Photonics Engineering, Technical University of Denmark, where he is currently a Professor. He is also the Head of the Coding and Visual Communication Group, DTU Fotonik. He is currently a Coordinator of the EU MSCA ITN RealVision. His interests include source coding, image, and video coding, distributed source coding, processing for image displays, 2-D information theory, coding for optical communication, and visual communications.