An Efficient Compression Scheme for the Multi-Camera Light Field Image

Eric Cornwell, Li Li, Zhu Li, and Yangfan Sun
School of Computing and Engineering, University of Missouri-Kansas City
Kansas City, MO 64110, USA
Email: eectz7@mail.umkc.edu, lil1@umkc.edu, lizhu@umkc.edu, ysb5b@mail.umkc.edu

***Abstract*—Current light field compression techniques lack robustness to handle both rate-distortion optimized motion compensation as well as latency during the encoding and decoding process. This paper focuses on a contribution approach that uses advanced frame prediction with affine and translational motion models and optimized view prediction structures. This method allows a significant compression performance gain over the current state of art of hierarchical temporal coding by 13.9%. The proposed method introduces an optimized encoding order that takes advantage of each group of pictures structure in order to leverage the dense perspective model of light field imagery. Both a global perspective model and a local affine model can be combined to show substantial distortion reduction at low processor costs. This contribution approach leads to an efficient and robust compression scheme for light field datasets.**

I. Introduction

Light field photography [1] is an emerging computational imaging technology that can be used to rectify lost data not captured by traditional 2-D photography [2]. This data can be characterized as depth information with refocusing attributes. These parameters are exploited by the nature of commercial light field cameras such as the Lytro Illum (Mountain View, California), as well as dense multi-camera arrays. A single image capture with these imaging devices produces multiple sub-image angular samples called sub-aperture images [2]. The abundance of sub-aperture images provide a source for more light-ray absorption on a light field camera photosensor and can be theorized by the schematic shown in Fig. 1. In comparison to a light field camera that uses a microlens array, a multi-camera array can be used to resolve greater angular ambiguities with higher resolution. The studies outlined in this paper will be done using the dense camera array devices. These arrays provide a much better spatial resolution and larger volumetric scene reconstruction as opposed to microlens light field cameras due to the combined photosensor area.

A consequence of the multi-focus and multi-perspective ability is the large data footprint that is inherent to light field images taken by dense camera array captures. For this reason, light field specific compression methods need to be implemented to handle the big data contained in the captures. Because of the multi-perspective views at the various sample locations, an abundance of pixels is adherent to this technology. This can be thought of as a two-dimensional matrix of 2D cameras with each sample location representing a separate camera. Essentially, this brings about four dimensions that summarize light field photography. Due to the abundant data present with each light field capture, an efficient compression scheme is needed for data transmission and viewing.

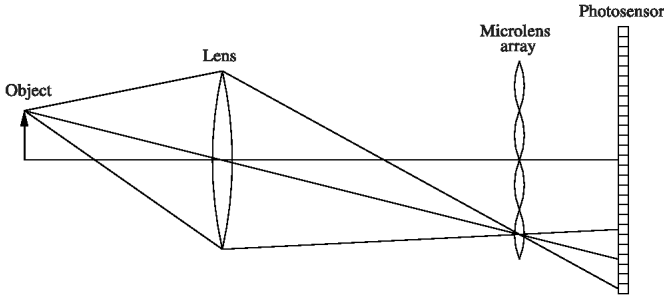

Fig. 1. Conceptual schematic of a light field camera

Careful attention must be paid to image quality when compressing because a lossy image could hinder the ultimate intent of the media. This could result in consequences that inadvertently defeat the purpose of light field photography by image degradation. A valuable tool to aid in the compression scheme is inter-prediction based on subsequent frames in the sequence. This compression can be achieved through various contributions such as motion estimation, affine transformation, temporal frame prediction, and a combination of picture grouping techniques. This paper offers multiple contributions for light field synthesis and compression for dense camera array datasets that have shown to output a bit-rate reduction over hierarchical coding performance while improving peak signal-to-noise ratio (PSNR).

The purpose of this paper is to investigate a practical compression method for light field datasets and to provide a demonstration and analysis of a robust compression scheme. A brief summary of light field photography and the technology behind it will be described. Once a solid foundation of the technology is discussed, a deeper dive into compression methodologies will be presented. The demonstration technique will output a compressed light field sequence using various video coding techniques such as temporal frame prediction and camera perspective shifts to predict subsequent frames. The analysis includes an evaluation of signal-to-noise ratios between methods by measuring Luma deviations of light field media from a multi-camera array provided by Technicolor Research Lab.

978-1-5090-3649-3/17/$31.00

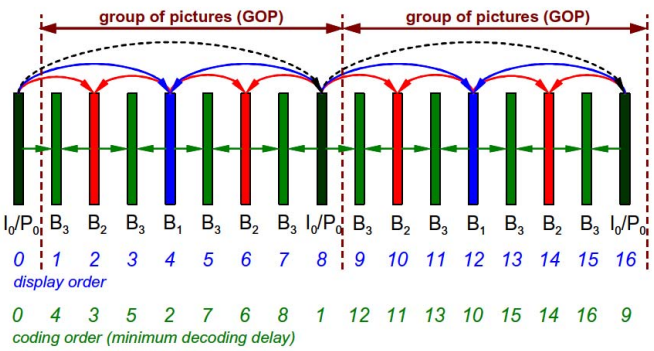

Fig. 2. 1-D hierarchical coding structure

The layout of this paper is as follows. Section II describes related work that was found to be a building block of the suggested compression method proposed in this paper. Section III dives into the light field dataset, encoding order, and proposed compression methods. It includes further studies into a contribution compression scheme where a variety of affine and motion prediction attributes will be used to take full advantage of the dense camera array architecture. The global motion model will be explained in detail, as well as the local affine model. The experimental results will highlight the gains achieved by individual and combined methods to accentuate the contributed scheme as shown in Section IV. Section V will summarize the results of this paper and offer an emphasis on future work in light field compression methodology.

II. Related works

Various imaging compression techniques have been studied in order to gain an insight into a practical light field compression scheme. A hierarchical coding scheme [3] has been presented by Schwarz et al. that uses a structured temporal approach for frame coding. This method selects sets of arbitrary coding structures in order to robustly define the group of pictures (GOP). The process then marks reference frames that are independent of the slice type. The frames are partitioned into I (intra-coded) or B (bidirectional predicted) frames, in which the B frames are marked with a hierarchical precedence on key frames. This structure directly impacts the coding order in order to minimize the decoding delay and optimize the rate-distortion (R-D) performance which is shown in Fig. 2.

Previous work has also been done to investigate practical compression methods for efficient block motion estimation algorithms, as well as global perspective transformation to encompass frame warping and translation. The global perspective motion model has been studied by Yu et al. [4] in which a set of motion vectors (MVs) characterize the motion between the original and reference frames. The output of this algorithm generates a warped reference frame for the original frame. It should be noted that this method lacks accuracy due to the naive nature of the algorithm to only focus on the global space and not capture local motion regions. There are also some methods [5] focusing on generating multiple warped reference frames according to multiple groups of perspective parameters. However, such kind of methods may be very complex mostly due to the addition of multi-parameter affine motion estimation (ME).

There were also some works trying to develop some local affine motion model algorithms to capture the local complex motions. For example, Huang et al. [6] developed a complete affine motion framework including lots of affine modes such as affine skip, affine inter, and affine merge for the High Efficiency Video Coding (HEVC) [7] to better characterize the local complex motions. In order to solve the local affine model complexity issue, Li et al. [8] proposed a reduction of the affine motion compensation (MC) parameter network by eliminating two of the six parameters to account for the various motion models contained in a sequence such as translation, rotation, and zooming. They have also provided some fast affine ME and MC algorithms to make the affine motion model more feasible for the modern video coding framework. These techniques can be used to fine-tune the coding pipeline for specific light field compression.

III. Proposed algorithms

The proposed algorithms can be mainly divided into three parts: the proposed 2-D hierarchical coding structure, the global perspective model, and the local 4-parameter affine motion model. They will be introduced in detail in the following subsections.

A. Hierarchical Temporal Model and Frame Coding Order

The first aspect of the proposed compression scheme is the hierarchical temporal model. We try to extend the 1-D hierarchical coding structure to a 2-D hierarchical coding structure to adapt to the light field image captured by the 4×4 camera array. The frame coding coder and frame types of the 16 views are shown in Fig. 3. The frame encoding order are designed following the hierarchical approach suggested by Schwarz et al. in order to code key reference frames. The method of precedence chosen for this paper was weighted on frame location relative to the light field capture. The top-left frame was the initial frame for the encoder and was the only frame that used intra-prediction. The heaviest weight was given to the other corners and outside frames, whereas the center frames were given lowest precedence. The reasoning for this is due to the fact that most of the information stored in the center frames is redundant with the exception of obscured objects in the scene and/or occlusions. The parallax phenomenon was shown to have a direct impact on these occlusions and had a greater effect on objects closer to the camera source. This can be seen more apparent later in the paper when the global perspective motion model results are displayed. It should also be noted that all the previous coded frames can be used as the reference frames of the current frame to exploit the correlations among various frames as much as possible.

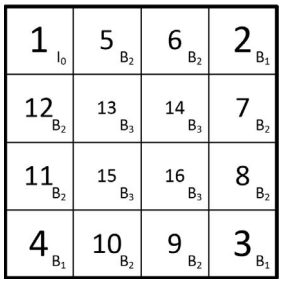

Fig. 3. 2-D hierarchical coding structure

TABLE I
THE QP SETTING OF DIFFERENT TEMPORAL LAYERS

Hierarchical layer	QP offset
1	2
2	6
3	8

Besides the 2-D hierarchical coding order, we also try to code each frame with a more reasonable quantization parameter (QP) to further optimize the R-D performance. From Fig. 3, we can see that four hierarchical layers are used in the 2-D hierarchical coding structure. For different hierarchical layers, since the influence of the frame distortions on the following frames will be quite different from each other, the QPs of various frames should also be set according to the influence of the frame distortions. According to our empirical studies, we find that the following settings as shown in table 1 will be beneficial to the overall performance. In table I, the QP offset of the other hierarchical layers compared with the intra frame is shown.

B. Global perspective model

To calculate the global perspective model between two neighboring frames, this paper provides two methods with different trade-offs between the prediction accuracy and the computational complexity. The first one is the direct calculation method using the intrinsic and extrinsic matrices. The second one is the key-points matching based methods.

1) The direct calculation method: This method allowed the reference frame prediction to be expanded to other frames by using a global space frame transform. An understanding of relative camera positions in the sequence is required in order to match subsequent frames with the correct camera. This method leverages the encoding order previously stated to predict the next frame in the sequence using a reference frame.

This approach to light field video compression involves leveraging the intrinsic and extrinsic parameters of the camera to reconstruct camera views for frame prediction. The intrinsic matrix provides a transformation from camera coordinates to image coordinates using translation, scaling, and shearing as follows.

$$K = \begin{bmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{1}$$

The intrinsic matrix K includes x-axis focal length f_x, y-axis focal length f_y, axis skew s, x-axis offset x_0, and y-axis offset y_0. The second matrix used to calibrate the camera is the extrinsic matrix. This matrix describes the transformation from camera coordinates to world coordinates.

$$Q = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \end{bmatrix} \tag{2}$$

The extrinsic matrix Q includes the rotation and the translational matrices about the x, y, and z axis. The intrinsic matrix K and extrinsic matrix Q can then form a transformation from world coordinate to one perspective.

$$\begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = s_i K_i Q_i \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{3}$$

In Eq. 3, x_i and y_i are the perspective shifted image coordinates, x_w, y_w, and z_w are the world coordinates, s_i is a scaling factor, K_i is the intrinsic matrix, Q_i is the extrinsic matrix.

Then the perspective transformation, which is a computer vision method that uses geometric properties to relate planar surfaces to one another, is derived to transform the coordinates from one perspective to the next. This does not use any motion estimation algorithms and can be solved using matrix operations as long as the translation and rotation vectors are known for each camera pose. The perspective transform can be easily derived as follows.

$$\begin{bmatrix} x_i \\ y_i \\ w_i \end{bmatrix} = \begin{bmatrix} c_1 & c_2 & c_3 \\ c_4 & c_5 & c_6 \\ c_7 & c_8 & 1 \end{bmatrix} \begin{bmatrix} x_j \\ y_j \\ 1 \end{bmatrix} \tag{4}$$

This is a projection using translation, rotation, and scaling parameters that are contained in a previously calculated camera intrinsic and extrinsic matrix from calibration. As can be seen from Eq. 4, there are 8 unknown coefficients, they will be coded using 32 bits per coefficient in the slice header.

A comparison of the predicted view versus the actual frame was achieved by visualizing the residual as shown in Fig. 4 and calculating the PSNR of the two images. The residual produced a PSNR value of 21dB which can be seen by the abundance of white pixels in the image. As stated earlier in this paper, the parallax effect is evident in this approach and the objects closer to the camera exhibited more adherent noise. For this reason, the global homography transformation is optimal for planar surfaces that are in the far-field range. Although this approach only exhibited marginal PSNR gains, this was a relatively fast operation due to only the matrix operation performed on the frame.

2) The feature matching method: In the feature matching based method, the homography matrix estimation was calculated using random sample consensus (RANSAC) and speeded up robust features (SURF) [9] of the two known frames. This is an iterative feature matching process that provides an estimated homography result based on back projection from

Fig. 4. Residue of the stitched image using the direct homography projection

Fig. 6. Residual of the predicted frame using SURF feature matching

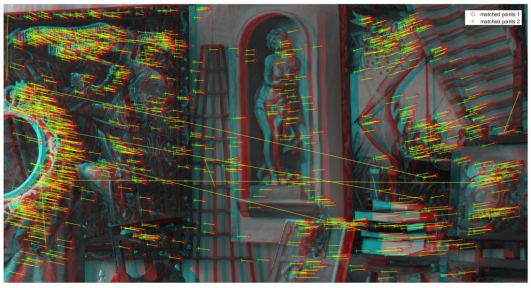

Fig. 5. SURF extraction and matching

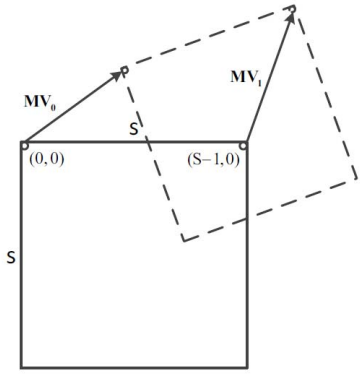

Fig. 7. Representation of the local 4-parameter affine motion model

one view to another. The SURF method allowed features to be extracted from the two reference frames and then matched using key-points which can be observed in Fig. 5. Because of the abundance of key-points yielding erroneous matches, RANSAC was used to filter out any outliers. Once the appropriate key-points are mapped, the resulting projection can be achieved by averaging the key-point vectors to approximate the translation, rotation, and scaling.

The residual of the RANSAC/SURF method yielded a PSNR of 23dB is shown in Fig. 6, which is 2dB higher compared with the direct calculation method. This gain hints at the possibility that the provided camera parameters were not calculated as precisely needed for the direct calculation method. Although the RANSAC/SURF estimation yielded slightly better results, it should be noted that the algorithm estimation time increased by a factor of 150 times. This is a consequence of the iterative method inherent of the RANSAC algorithm. In the experiments part shown in the next section, we give the results using the RANSAC/SURF method for a better R-D performance.

C. Local 4-parameter affine motion model

The third and final contribution to the proposed combined light field compression involved the 4-parameter local affine motion model described. Originally, the local affine model included the six degrees of freedom (6-DOF) to solve camera motions such as camera track, boom, pan, tilt, zoom, and roll. In [7], the six parameters of the affine motion model are reduced to four parameters to get a better balance between the model accuracy and the number of header bits. In this paper, we also apply the four-parameter affine motion model to coding the light field image to try to achieve better coding efficiency. Also, since the camera lens is calibrated and stationary relative to the camera sensor, the 4-parameter affine motion model can be better approximated to the camera and object motions for the light field image.

The use of the 4-parameter affine motion model allows the complexity of the local affine model to decrease substantially by reducing the calculations by 1/3 compared with the six-parameter affine motion model. As shown in Fig. 7, since four parameters are needed for the local transformation, two motion vectors in the top left corner and top right corner were used to represent the four parameters within a given block of pixels. These motion vectors were used to interpolate the reference block to the encoded block and are needed to transmit to the decoder to reconstruct the block. Similar to the newest video coding standard HEVC, two MV determination methods, e.g., advanced affine motion vector determination method and affine model merge, are applied to determine the two MVs more efficiently. More detailed information about the proposed 4-parameter affine motion model can be found.

IV. Experimental results

A large dataset was chosen to validate the various compression schemes being studied. As mentioned above, the Lytro Illum camera only supports lower resolution images of

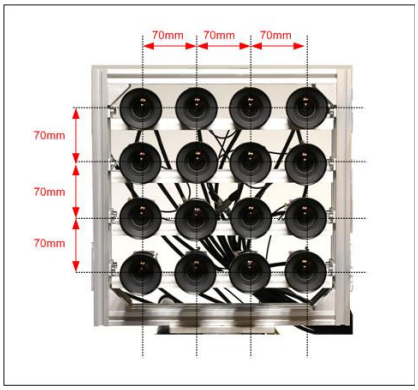

Fig. 8. Dense multi-camera array

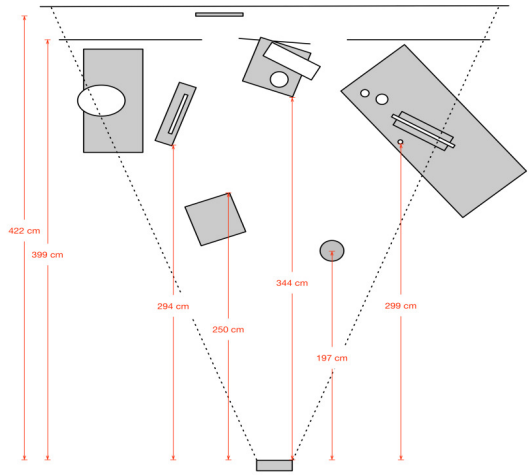

Fig. 9. Technicolor painter scene composition

434×625 pixels. A multi-camera array is able to simulate a light field camera capture on a larger scale and can provide a higher resolution, so the "Technicolor Painter" dataset was used [10]. This dataset was taken with sixteen synchronized cameras, each approximately $70mm$ apart, and arranged in a 4×4 array as shown in Fig. 8. The camera rig provided a 2048×1088 resolution with a bit depth of 24. The cameras acquired 372 captures at 30 frames per second. This equated to a total of 5952 frames and duration of twelve seconds. Also, the intrinsic and extrinsic parameters were provided for each camera in the form of Eqs. 1 and 2 for rectification purposes [11].

The 3D scene geometry that was captured consisted of a spatially robust layout that is depicted in Fig. 9. The scene was dissected into multiple depth planes by placing objects at various distances. This allowed for focal adjustments to be made using the light field focal slices. Also, the geometry added the complexity of salient objects that were very rich in color data. The dataset was abundant in features and for this reason, was a great subject to use for researching feasible compression while still attaining important detail.

In order to compare the performance of each method studied, the proposed algorithms are implemented in the HEVC reference software HM16.7 [12], and average PSNR on the Luma values, as well as bit rate (kbps) were calculated. The PSNR operation was used to compare the original lossless video to the coded videos in decibels (dB). This allowed a confidence level to be established among different video codecs. Also, in order to properly reference current codec performances, intra prediction coding was used to compare the results. The performance is summarized in table II.

TABLE II
THE PERFORMANCE OF EACH COMPONENT

bitrate	intra	2-D hierarchical	global	local	All
265	39.02	39.96	39.99	39.98	39.99
119	37.54	38.12	38.30	38.30	38.34
62	35.41	35.94	36.33	36.31	36.34
33	33.33	33.64	33.97	33.97	33.98

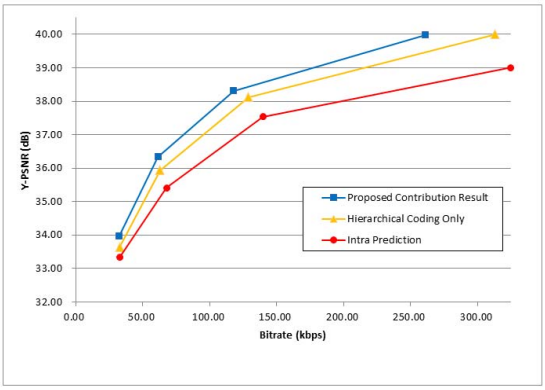

Fig. 10. The R-D curve of the proposed algorithm

The compression performance results were plotted to compare the GOP hierarchical method and intra prediction versus the proposed combined contribution method as shown in Fig. 10. By adjusting the quantization parameter, a rate given in kilobits per second could be understood with corresponding PSNR of the Luma channel. The higher the PSNR value, the higher the quality of the images. It can be seen that the proposed method outperformed the 1-D hierarchical coding and intra prediction regardless of bit rate. It should be noted that the performance of the combined methods averaged a gain of 0.5dB across all bit rates. Also from Fig. 10, we can see that the proposed method will bring better R-D performance in high bitrate case compared with the low bitrate case. The suggested method which leveraged relative frame positions could potentially play a large role in future light field compression standards.

To further analyze and compare the R-D curves above, the Bjontegaard-Delta bitrate (BD-rate) method [13] was used. This allowed for a percentage of bit savings to be calculated when referencing the various methods spanning the entire curve. Basically, an average distance between the curves was calculated. The results of the BD rates are provided in Table III In Table III, the negative values mean the R-D performance

TABLE III
BD RATE COMPARISONS BETWEEN DIFFERENT METHODS

Reference method	Tested method	BD rate
1-D hierarchical	2-D hierarchical	–10.3%
2-D hierarchical	local affine model	–2.9%
2-D hierarchical	global perspective model	–1.9%
2-D hierarchical	local and global	–4.0%
1-D hierarchical	all methods	–13.9%
intra prediction	all methods	–37.8%

improvement.

The 1-D hierarchical coding structure defined in the random access main configuration was used as a base reference method to demonstrate the results when comparing to HEVC. This codec contains the ability to implement temporal hierarchical coding. The tested methods involving the local and global motion models were used to validate the bit savings compared to HEVC. The results of the studied methods displayed vast bit reduction with each method used. The global method showed a 1.9% bitrate saving, while the local model presented a 2.9% bitrate saving. The combined global and local methods showed a 4.0% R-D performance improvement which is less than the sum of local and global models since there are some performance overlaps between the global and local motion models. All methods together uncovered a 13.9% bit saving over the 1-D hierarchical approach. In order to further demonstrate the gains accomplished, all methods proposed in this paper were combined to show a 37.8% bit saving when compared to intra prediction.

V. CONCLUSION

In summary, the light field media compression methods studied in this paper gives an insight into various considerations when transmitting, storing, and retrieving large image datasets. The proposed 2-D hierarchical coding structure, combined global and local high-order motion models provided the best solution for motion estimation prediction of light field images. A significant bitrate saving as high as 13.9% is achieved compared with the 1-D hierarchical coding structure in HEVC. Further studies should involve comparing the results of a multi-camera array versus a light field camera that uses a lenslet aperture. PSNR gains and other performance parameters should be studied in this comparison. Another method for compression for further studies is depth reconstruction and minimizing redundancy of lenslet images. This could aid in scene reconstruction for view synthesis and frame prediction. Also, 3D reconstruction could aid in the compression by taking advantage of the dimensional projection and correlation between the various focal planes.

REFERENCES

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '96. New York, NY, USA: ACM, 1996, pp. 31–42.

[2] D. Dansereau and U. of Sydney., *Plenoptic Signal Processing for Robust Vision in Field Robotics*. University of Sydney, 2014.

[3] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical b pictures and mctf," in *2006 IEEE International Conference on Multimedia and Expo*, July 2006, pp. 1929–1932.

[4] H. Yu, Z. Lin, and F. Teo, "An efficient coding scheme based on image alignment for H.264/AVC," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, May 2009, pp. 629–632.

[5] T. Wiegand, E. Steinbach, and B. Girod, "Affine multipicture motion-compensated prediction," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 15, no. 2, pp. 197–209, Feb. 2005.

[6] H. Huang, J. Woods, Y. Zhao, and H. Bai, "Control-point representation and differential coding affine-motion compensation," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 10, pp. 1651–1660, Oct. 2013.

[7] G. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[8] L. Li, H. Li, D. Liu, Z. Li, H. Yang, L. Sixin, H. Chen, and F. Wu, "An efficient four-parameter affine motion model for video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. PP, no. 99, pp. 1–1, 2017.

[9] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded-up robust features (surf)," *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346 – 359, 2008, similarity Matching in Computer Vision and Multimedia. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1077314207001555

[10] D. Doyen, T. Langlois, B. Vandame, F. Babon, G. Boisson, N. Sabater, R. Gendrot, and A. Schubert, "[Light-field AHG] light field content from 16-camera rig," Document ISO/IEC JTC1/SC29/WG11 MPEG2017/m40010, Jan. 2017.

[11] D. Doyen, N. Sabater, G. Boisson, and B. Vandame, "[light-field ahg] pre-processing for light field camera rig," Document ISO/IEC JTC1/SC29/WG11 MPEG2017/m40011, Jan. 2017.

[12] HM, HEVC test Model. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/

[13] G. Bjontegaard, "Calculation of average PSNR differences between RD-curves," Document VCEG-M33, Austin, Texas, USA, Apr. 2001.