ADVANCED LIGHT FIELD FRAME PREDICTION

FOR OPTIMIZED COMPRESSION

A Thesis

in

Electrical Engineering

Presented to the Faculty of the University
of Missouri-Kansas City in partial fulfillment of
the requirements for the degree

MASTER OF SCIENCE

by

ERIC CORNWELL

B.S., University of Missouri-Kansas City, 2013

Kansas City, Missouri

2017

ADVANCED LIGHT FIELD FRAME PREDICTION FOR OPTIMIZED COMPRESSION

Eric Cornwell, Candidate for the Master of Science Degree

University of Missouri–Kansas City, 2017

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 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.

APPROVAL PAGE

The faculty listed below, appointed by the Dean of the School of Computing and Engineering, have examined a thesis titled "Advanced Light Field Frame Prediction For Optimized Compression," presented by Eric Cornwell, candidate for the Master of Science degree, and certify that in their opinion it is worthy of acceptance.

Supervisory Committee

Zhu Li, Ph.D., Committee Chair

Department of Computer Science & Electrical Engineering

Reza Derakhshani, Ph.D.

Department of Computer Science & Electrical Engineering

Cory Beard, Ph.D.

Department of Computer Science & Electrical Engineering

CONTENTS

ILLUSTRATIONS

TABLES

Table Page

ACKNOWLEDGEMENTS

I would like to thank my academic adviser and professor Dr. Zhu Li and Dr. Li Li for providing me an opportunity and helping along the way to complete my thesis work.

I would like to thank University of Missouri - Kansas City for providing me an opportunity to work on this thesis report.

CHAPTER 1

INTRODUCTION

The purpose of this paper is to investigate a viable 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 will use a suggested work flow that starts with capturing the light field images and ends with a compressed light field sequence using various 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. In order to provide light field synthesis analysis for this paper, a light field dataset will be created using a Lytro Illum camera and further analysis will be done on a multi-camera array provided by Technicolor Research Lab [3].

The layout of this paper is as follows. Chapter 2 provides a brief overview and background of light field technology and a high level outline of compression methods. Chapter 3 describes related work that this paper has referenced and found to be a stepping stone into the suggested compression methods. The next section 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. A conclusion section will summarize the results of this paper and offer an emphasis on future work in light field compression methodology.

CHAPTER 2

BACKGROUND

Light field photography is an emerging computational imaging technology that can be used to rectify lost data not captured by traditional 2-D photography [1]. 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 multi-camera arrays. A single image capture with these imaging devices produces multiple sub-image angular samples called sub-aperture images [1]. 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 figure 1 [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 [figure 2]. The initial media presented in this paper will be captured from a Lytro Illum camera in order to demonstrate the data packaging and to get familiar with the calibration procedure. Further 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.

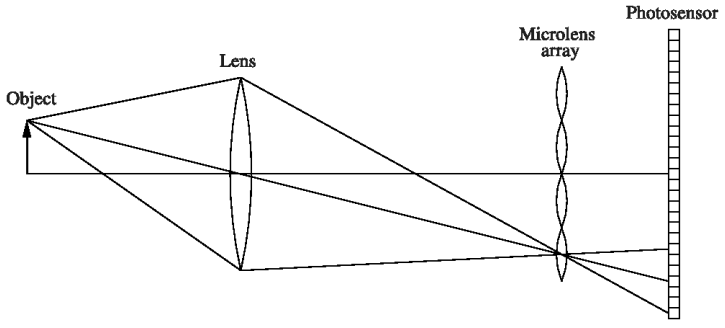

Figure 1 Conceptual schematic of a light field camera [1]

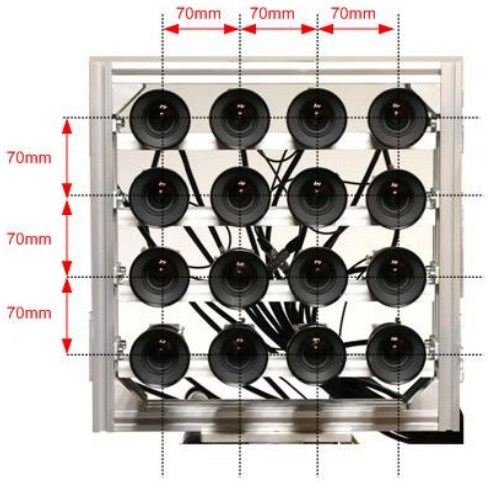

Figure 2 Multi-camera array [3]

A consequence of this 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. 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. The purpose of this paper is to investigate a viable compression method for light field media and to provide a demonstration and analysis of a robust compression scheme specifically aimed at light field datasets. 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. Also, by adjusting the coding pipeline for optical distortion adaptation, compression can be optimized. This paper offers multiple contributions for light field video synthesis and compression for

dense camera array datasets that have shown to output a bit-rate reduction over hierarchical codec performance while improving power signal-to-noise ratio (PSNR).

CHAPTER 3

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 has been presented by Schwarz et al. that uses a structured temporal approach for frame coding [8]. 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 [8]. 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 which is shown in figure 3.

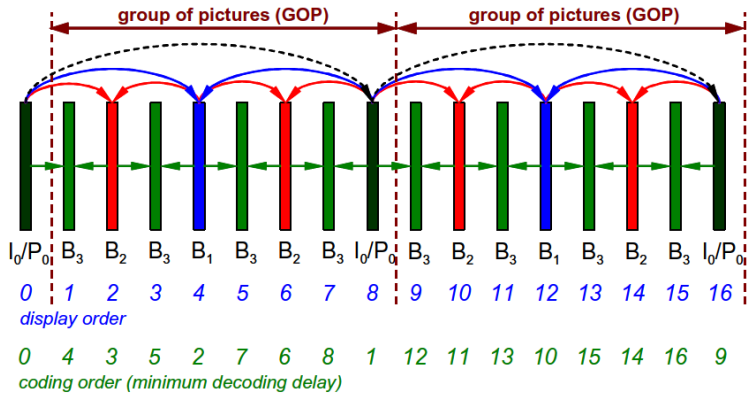

Figure 3 Hierarchical Coding Structure [8]

Previous work has also been done to investigate practical compression methods for efficient block motion estimation on a local scale, as well as global affine transformation to encompass frame warping and translation. A direct transformation that uses translation, rotation, and scaling due to perspective shifts is called an affine transformation [7]. The global affine motion model has been studied by Yu et al. in which a set of motion vectors

(MVs) characterize the motion between an original and reference frame [11]. The output of this algorithm generates a warped reference frame [11]. It should be noted that this method lacks accuracy due to the naïve nature of the algorithm to only focus on the global space and not capture local motion regions. This is mostly due in part by the complexity involved with the addition of multi-parameter affine motion estimation (ME). In order to solve the local affine model complexity issue, Li et al. 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 [7]. This method frees up the left over bits in the encoding process and further compresses the media. These techniques can be used to fine-tune the coding pipeline for specific light field compression.

CHAPTER 4

EXPERIMENTAL AND COMPUTATIONAL DETAILS

4.1 Initial Experiment on Calibration and Synthesis on a Test Dataset

After the various compression techniques were studied, the next task for this project was to obtain hardware and software for the light field video compression synthesis and calibration in order to analyze the data packaging of a light field source. The hardware consisted of a Lytro Illum camera 40 megaray light field camera. The software used to synthesize the light field images was MATLAB with an open-source Light Field Toolbox [2]. All of these software components were ran on a quad-core Intel processor and 8GB of RAM.

The Lytro Illum camera captures a raw uncompressed lenslet image of 5368 x 7728 pixels. This raw file is saved as a Light Field Raw (LFR) image and includes metadata such as focus and depth parameters, as well as the raw image without demosaicing with 10 bits per pixel. This works out to each LFR file being roughly 53MB each.

Like with any imaging device, a good rule of thumb is to calibrate the imaging sensor and lens. This is good practice to compensate for lens distortion and warping by calculating an intrinsic parameter input for the camera. The light field camera introduces one more step to the calibration procedure in order to align the 2D grid of the lenslet to the center of each lenslet image. This grid, as shown in figure 4, is called a white image and is very important during the decoding process of the light field. The Lytro Illum is composed of a 15 x 15 grid pattern, with each grid element representing a camera view.

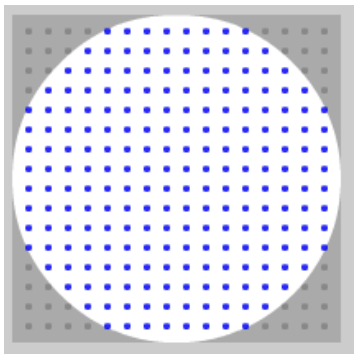

Figure 4 Angular grid sample locations on a microlens photosensor

Once the grid pattern was calibrated for the camera, the lens distortion compensation calibration was implemented. For this paper, an 8x6 checkerboard target was used with a 23mm spacing. It should be noted that smaller spacing and more abundant squares on the target yield a better calibration result for a light field camera due to its lenslet aperture.

Figure 5 Lytro camera calibration setup

The calibration process consisted of capturing images of the checkerboard at various poses as shown in figure 5. For this paper, ten poses were used. Once the captures were

made, the Light Field Toolbox enabled the projection of each checkerboard corner to be mapped for intrinsic and extrinsic calculations. The intrinsic matrix provides a transformation from camera coordinates to image coordinates using translation, scaling, and shearing [10].

$$K = \begin{pmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

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 [9].

$$Q = \begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \end{pmatrix} \quad (2)$$

The extrinsic matrix, Q, includes rotation (r) and translation (t) about the x, y, and z axis. After the calibration was complete, the pose of each capture was estimated iteratively to depict the margin of compensation.

The LFR files were loaded into MATLAB to begin the synthesis portion of the analysis. Each raw light field image was loaded and decoded using the Light Field Toolbox [2]. The decoding process transformed the 2D grid of lenslet images into a 5D array. This process consisted of demosaicing, color correction, histogram equalization, and rectification. The rectification step leveraged the calibration intrinsic and extrinsic parameters, while the demosaicing process used the grid calibration discussed earlier in the paper. Both of these processes were crucial in transforming the raw image into a robust matrix of 2D images. The output of the decoding produced a five dimensional array of size 15x15x434x625x4. The number of views is captured by the 15x15 array, the resolution of the views was 434x625,

and the last dimension represented the three color channels (red/blue/green) and a weighting factor. The weight channel was used during the histogram equalization step to negate any zero weighting during the calculation. The last step of the synthesis process was to export each horizontal 2D image view as a lossless portable network graphic (PNG) image on the various focal planes. At this stage, each image was uncompressed at 434x625 pixels and approximately 1.25MB each.

4.2 Proposed Light Field Compression Analysis and Methods

4.2.1 Dataset Used for Compression Analysis

Once the synthesis and calibration method was understood using the light field camera, a larger dataset was chosen to validate the various compression schemes being studied. As mentioned above, the Lytro Illum camera only supported lower resolution images of 434x625. 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 [3]. This dataset was taken with sixteen synchronized cameras, each approximately 70mm apart, and arranged in a 4x4 array [figure 2]. The synchronization was very crucial in order to provide exact timing triggering for each camera capture. This was achieved by using a global clock mechanism that was referenced by a global trigger. The global trigger sent out a control signal to each camera in parallel to snap the photo thirty times second. The physical camera array layout relative to the scene is depicted in figure 6. The camera naming convention and order becomes important later in the paper when a recommended sequence is described. The camera rig provided a 2048x1088 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 equation 1 and 2 for rectification purposes [4].

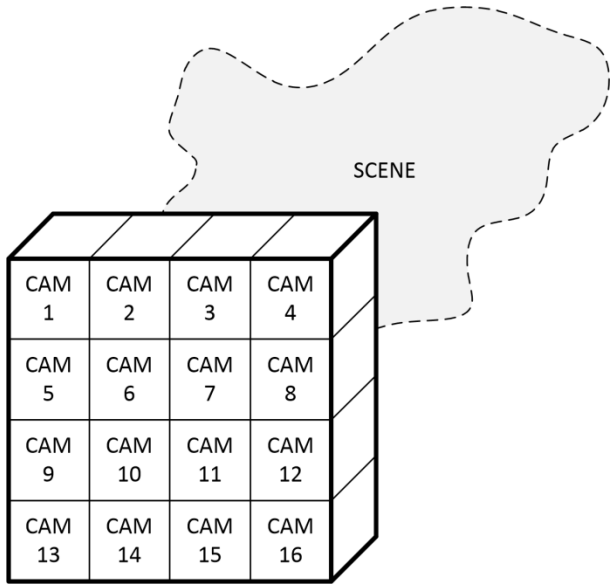

Figure 6 Multi-camera array layout for Technicolor Painter

The 3D scene geometry that was captured consisted of a spatially robust layout that is depicted in figure 7. 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 rich in features and for this reason, was a great subject to use for researching feasible compression while still attaining important detail [figure 8].

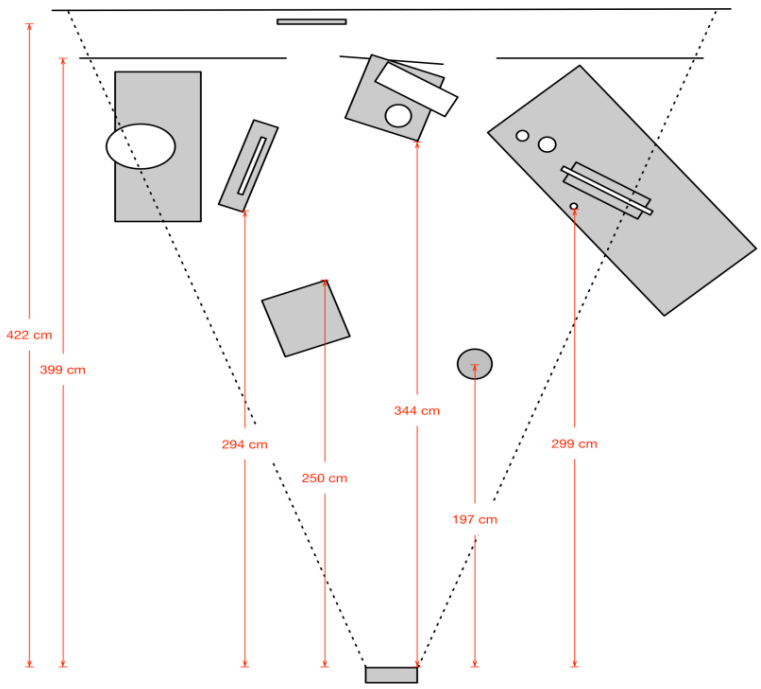

Figure 7 Technicolor painter scene composition [3]

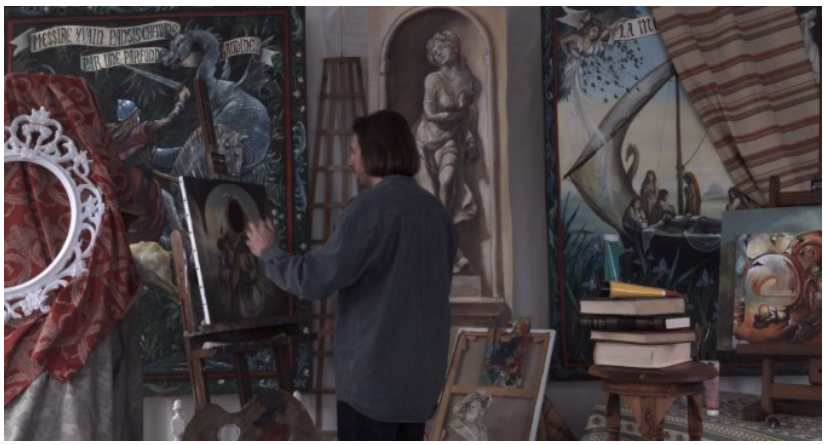

Figure 8 "Technicolor Painter" light field dataset [3]

4.2.2 Contribution I: Hierarchical Temporal Model and Frame Encoding Order

The first contribution to the proposed compression scheme is the hierarchical temporal model. The frame encoding order followed 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 [figure 9]. 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 where 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 affine motion model results are displayed.

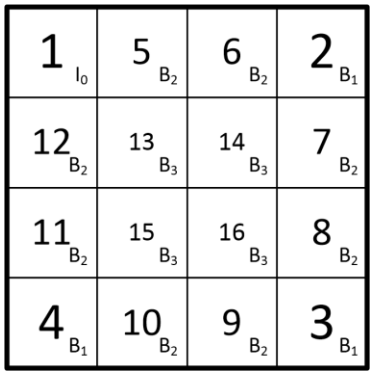

Figure 9 Frame encoding order of light field capture

4.2.3 Contribution II: Global Motion Model

The second method for compression analysis exploited the nature of the light field sequence by using a global motion model. 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. This affine transformation, which is called homography, is a computer vision method that uses geometric properties to relate planar surfaces to one another. It is essentially a perspective transformation and the homography matrix can be used to transform world coordinate pixels 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 [6].

$$\begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} = sH \begin{pmatrix} x_w \\ y_w \\ z_w \\ 1 \end{pmatrix} = sKQ \begin{pmatrix} x_w \\ y_w \\ 0 \\ 1 \end{pmatrix} = s \begin{pmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ 1 \end{pmatrix} \quad (3)$$

In equation 3, x_i and y_i are the perspective shifted image, x_w, y_w, and z_w are the world coordinates, H is the homography matrix, s is a scaling factor, K is the intrinsic matrix, Q is the extrinsic matrix.

Equation 3 allows a transformation from world points in 3D space to a perspective shift in 2D. This is a projection using translation, rotation, and scaling parameters that are contained in a previously calculated camera intrinsic and extrinsic matrix from calibration [6]. For the frame prediction, the algorithm predicted frame x_n from perspective shifts given by frame x_{n-1} and frame x_{n+1}. The algorithm recovered the two perspectives and stitched the

images together to form a completed view. A comparison of the predicted view versus the actual frame was achieved by visualizing the residual and calculating the PSNR of the two images [figure 10]. The residual produced a PSNR value of 29dB 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.

Figure 10 Residual of the stitched images using homography projection

In order to validate the direct calculated homography projection using the intrinsic and extrinsic values, the homography matrix estimation was calculated using random sample consensus (RANSAC) and speeded up robust features (SURF) of the two known frames. This is an iterative feature matching process that provides an estimated result. The SURF method allowed features to be extracted from the two reference frames and then matched using keypoints which can be observed in figure 11. Because of the abundance of keypoints yielding erroneous matches, RANSAC was used to filter out any outliers. Once the

appropriate keypoints are mapped, the resulting projection can be achieved by averaging the keypoint vectors to approximate the translation, rotation, and scaling. The residual of the RANSAC/SURF method yielded a PSNR of 31dB. Although the RANSAC/SURF estimation yielded slightly better results by 2dB, 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.

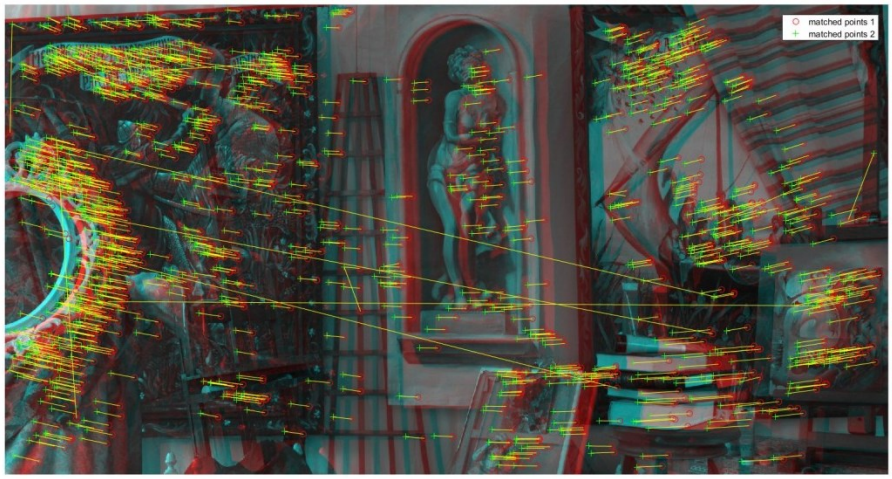

Figure 11 SURF feature extraction and matching

A histogram of the comparison between the homography view synthesis methods is shown in figure 11. The RANSAC/SURF method contains slightly more zero level luma pixels than the camera parameter homography method, but the two methods mirror each other accordingly. A calculation of mean squared error (MSE) was used to further compare the predicted image with the actual. Both the direct calculation and RANSAC/SURF methods obtained an MSE of 461.6 and 857.9 respectively.

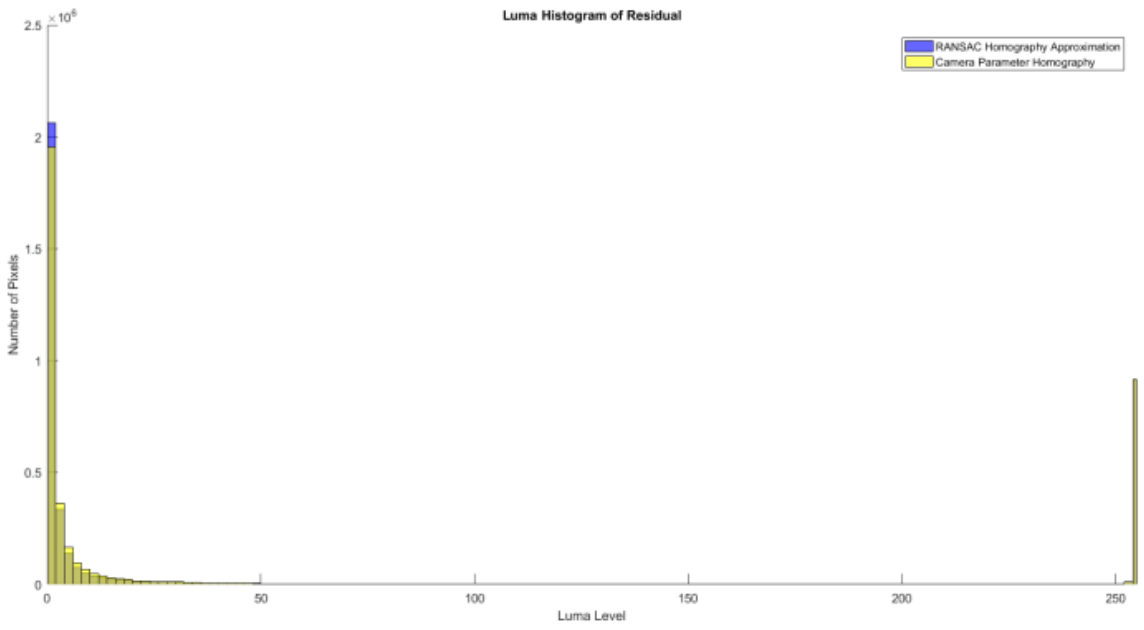

Figure 12 Histogram comparison of the homography methods

4.2.4 Contribution III: Local Affine Model

The third and final contribution to the proposed combined light field compression involved the local affine model described by Li et al. [7]. Originally, the local affine model included the six degrees of freedom to solve camera motions such as camera track, boom, pan, tilt, zoom, and roll [7]. In reality, the light field camera movement can be summarized by translation, rotation, and zoom. This is due to the fact that the camera lens is calibrated and stationary relative to the camera sensor. As a consequence of this assumption, the six parameter system of equations can be eliminated to four parameters [7]. This allows the complexity of the local affine model to decrease substantially by reducing the calculations by 1/3. Once the parameters needed for the local transformation, two motion vectors where used within a given block of pixels [figure 13]. These vectors were used to interpolate the reference image to the encoded image. Similar to the global model, this method took little processing time due to the direct calculation.

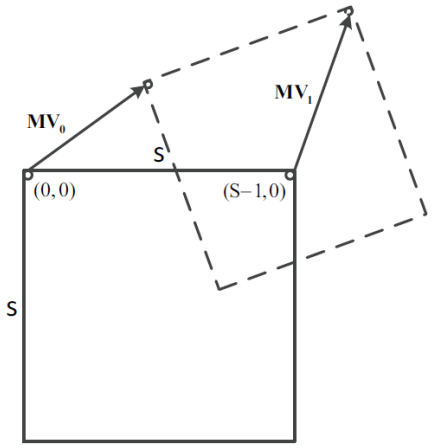

Figure 13 Affine transformation motion vectors [7]

4.2.5 Results

In order to compare the performance of each method studied, average PSNR on the luma (Y) values, as well as bit rate (kbps) were calculated. The PSNR operation was used to compare the original lossless video to the codec videos in decibels (dB). This allowed a confidence level to be established between 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 1.

Table 1 Performance of Contributions

kbps (approx.)	**Intra Prediction Coding**	**GOP Hierarchical Y-PSNR (dB)**	**Global Model Y-PSNR (dB)**	**Local Model Y-PSNR (dB)**	**All Y-PSNR (dB)**
265	39.02	39.96	39.99	39.98	39.99
119	37.54	38.12	38.31	38.30	38.31
62	35.41	35.94	36.33	36.31	36.34
33	33.33	33.64	33.97	33.97	33.98

The compression performance results were plotted to compare the GOP hierarchical method and intra prediction versus the proposed combined contribution method. 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 hierarchical coding and intra predication 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. The suggested method which leveraged relative frame positions could potentially play a large role in future light field compression standards. To further describe the bit savings, the PSNR was plotted against bits per pixel (bpp). This allowed a direct correlation to the bits being conserved with each method.

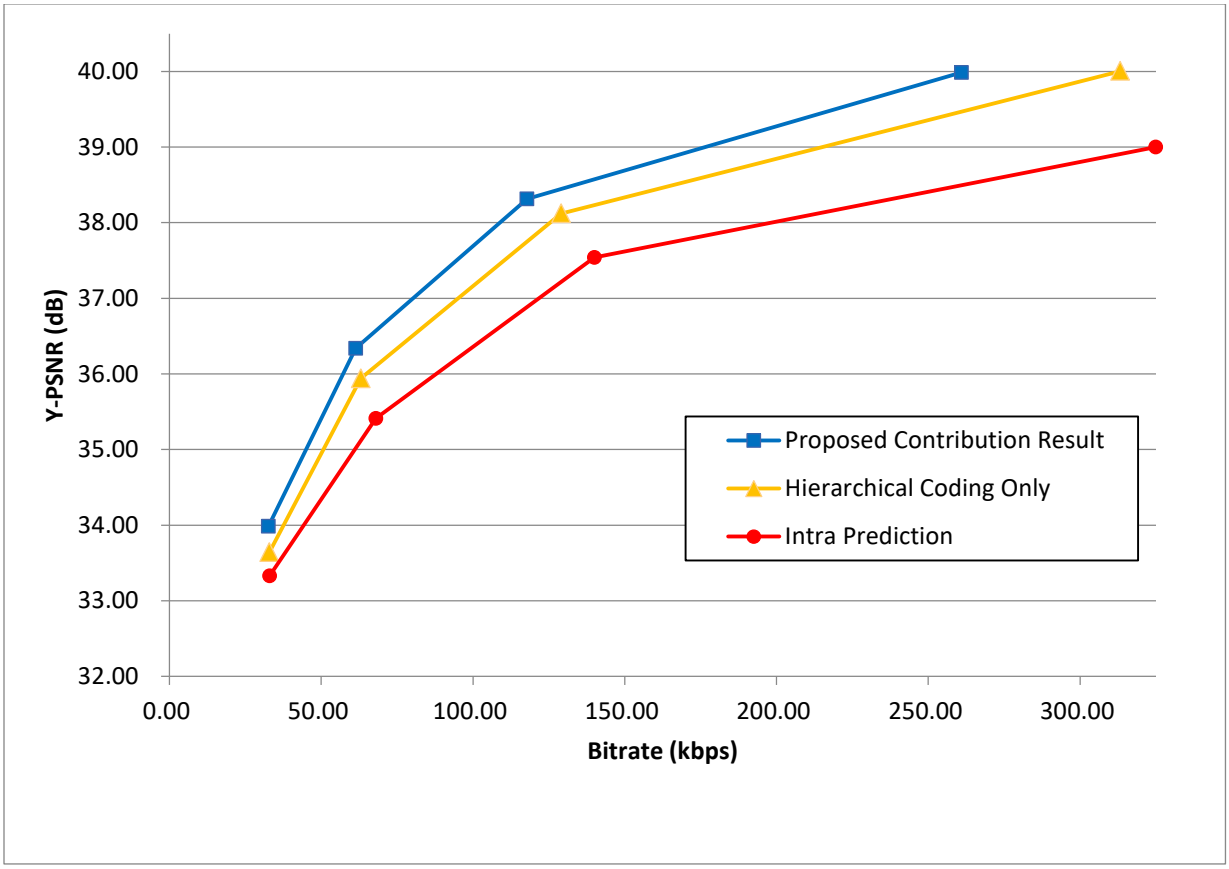

Figure 14 RD Plots of motion estimation compression analysis using bitrate

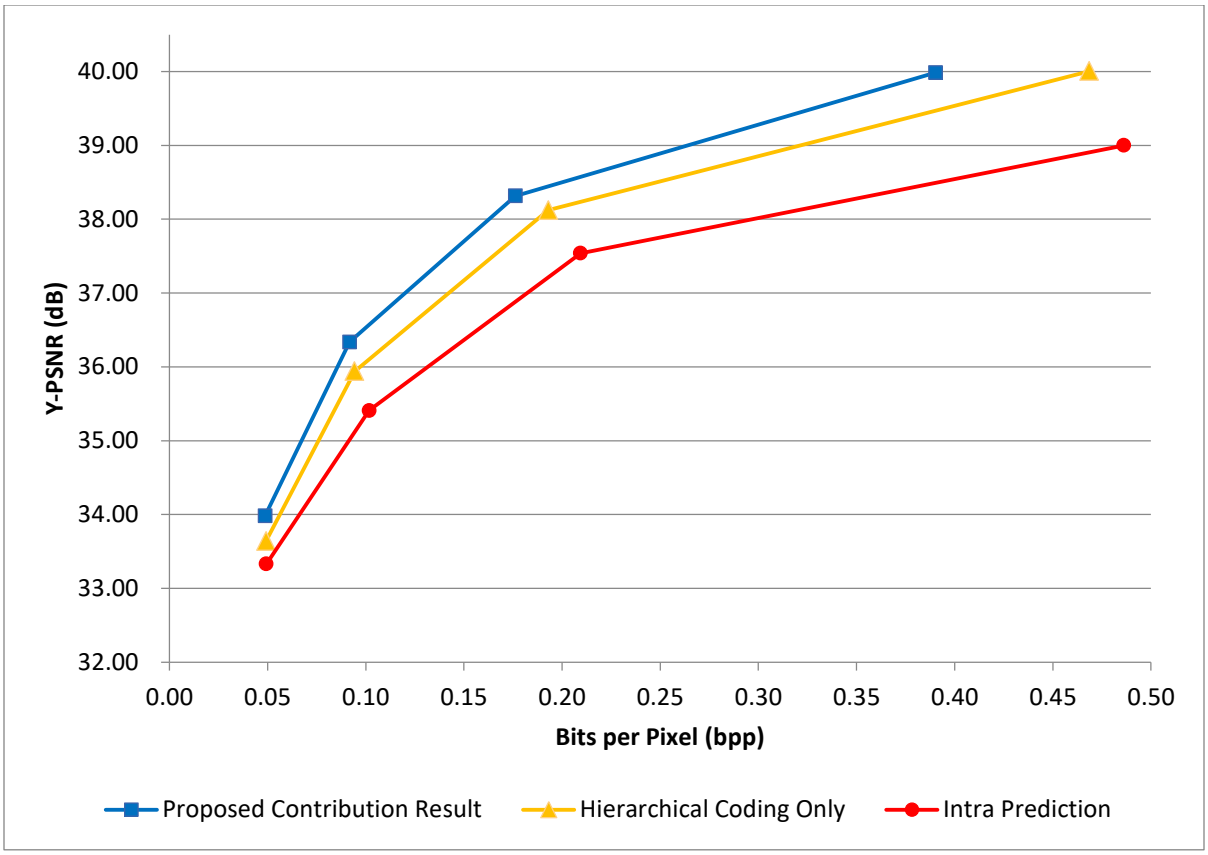

Figure 15 RD Plots of motion estimation compression analysis using bits per pixel

To further analyze and compare the rate-distortion (RD) curves above, the Bjontegaard (BD) method 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 2.

Table 2 BD Rate Comparison

Reference Method	Tested Method	BD Rate (%)
Hierarchical	Local	-2.9
Hierarchical	Global	-1.9
Hierarchical	All	-13.9
Intra Prediction	All	-37.8

The hierarchical method was used as a base reference method to demonstrate the results when comparing to the latest codec known as high efficiency video coding (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. In order to further demonstrate the gains accomplished, all methods proposed in this paper were combined to show a 37.8% bit savings when compared to intra prediction.

CHAPTER 5

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. As seen from figure 14 and 15, the proposed contribution prediction scheme provided the best solution for motion estimation prediction of light field images. An interesting point to consider in this analysis is computational time between frame predictions. Although the view synthesis using homography produced a lower PSNR, the operation was very fast, especially when comparing it to the RANSAC/SURF method. This situation would be ideal for fast light field transmission since the media is inherently large and there is a need for quick rendering at the cost of retaining quality. By using image analysis it can be seen that the view synthesis performed better on objects further from the camera source. This is a consequence of the sparse view of the combined camera array. Further studies should involve comparing the results of a multi-camera array versus a light field camera that uses a lenslet aperture. 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.

REFERENCE LIST

[1] D. G. Dansereau, "Plenoptic Signal Processing for Robust Vision in Field Robotics". Thesis Submission, University of Sydney. January 2014.

[2] D. G. Dansereau, "Light-Field Toolbox for MATLAB". December 2015. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/49683-light-field-toolbox-v0-4

[3] D. Doyen et al., "Light Field Content from 16 camera rig". ISO/IEC JTC1/SC29 WG11 Doc. m40010.Geneva, January 2017.

[4] D. Doyen et al., "Pre-processing for light-field camera rig". ISO/IEC JTC1/SC29 WG11 Doc. m40011, Geneva, CH, January 2017.

[5] FFmpeg. [Online]. Available: https://ffmpeg.org/

[6] D. J. Lee, "Camera Calibration – Homography". March 2012. [Online]. Available: http://ece631web.groups.et.byu.net/Lectures/ECEn631%2010%20-%20Homography%20Camera%20Calibration.pdf

[7] L. Li et al., "An Efficient Four-Parameter Affine Motion Model for Video Coding". IEEE Transactions on Circuits and Systems for Video Technology. May 2017.

[8] H. Schwarz et al., "Analysis of Hierarchical B Pictures and MCTF". Multimedia and Expo, 2006. IEEE International Conference. July 2006.

[9] K. Simek, "Dissecting the Camera Matrix, Part 2: The Extrinsic Matrix". August 2012. [Online]. Available: http://ksimek.github.io/2012/08/22/extrinsic/

[10] K. Simek, "Dissecting the Camera Matrix, Part 3: The Intrinsic Matrix". August 2013. [Online]. Available: http://ksimek.github.io/2013/08/13/intrinsic

[11] H. Yu, “An efficient coding scheme based on image alignment for H.264/AVC”. Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium. May 2009, pp. 629–632.

VITA

Eric Cornwell was born in 1981 in Kansas City, Missouri, USA. He was educated in local public schools and graduated from the University of Missouri-Kansas City in 2013 with B.E. degree in Electrical and Computer Engineering.