RD-OPTIMIZED 3D PLANAR MODEL RECONSTRUCTION & ENCODING FOR VIDEO COMPRESSION

Cheng Yang [†]*, Gene Cheung* [†]*, Seishi Takamura* [‡]

[†] National Institute of Informatics [‡] NTT

ABSTRACT

Conventional video coding approaches follow a hybrid motion prediction / residual transform coding paradigm, which limits the discovery of redundancy to individual pairs of video frames. On the other hand, computer vision techniques like structure-from-motion (SfM) have long exploited redundancy across a large group of frames to estimate a rigid 3D object structure. In this paper, leveraging on previous SfM techniques, we construct a rate-distortion (RD) optimized 3D planar model from a target spatial region in a frame group as a unified signal predictor for these frames. The prediction accuracy of the model is optimally traded off with the cost of coding such representation as side information (SI). Specifically, we approximate a roughly flat spatial region in the video as a 2D plane in 3D space, and project pixels from the frame group to a 2D grid on the plane with appropriate density. The boundary of the irregularly shaped pixel body on the plane is first coded using arithmetic edge coding (AEC), and then the body is encapsulated into a tight-fitting rectangular region, which is encoded as an intra-frame using HEVC. The pixels inside the rectangle but outside the pixel body—called don't care region (DCR)—are filled optimally by minimizing an l_1-norm of the transform coefficients using linear programming. Experimental results show that the RD-optimized planar model improves coding performance over native HEVC implementation.

***Index Terms*—** structure-from-motion, video coding, convex optimization

1. INTRODUCTION

It is estimated[1] that IP video will reach 82% of all consumer Internet traffic by 2021, so while long-studied, video compression will continue to be a research topic of practical interest. Traditional video coding approaches—including video coding standards like H.264 [1] and HEVC [2]—follow a hybrid motion prediction / residual transform coding paradigm: prediction residues computed from a motion model between two video frames are transformed and entropy coded. Thus, with a few exceptions like background extraction and 3D video coding [3–5], only redundancy between pairs of video frames are discovered and exploited for compression[2].

Concurrently, computer vision techniques like structure-from-motion (SfM) have long exploited redundancy across a large group of frames to construct a rigid 3D object model [7–9]. Recent 3D reconstruction methods from 2D video have shown good model accuracy for relatively simple object shapes. Instead of transmitting motion information per inter-coded frame, one approach [10] is to reconstruct a single 3D model of an rigid object *a priori* for back-projecting texture information to the corresponding spatial locations in the video frames as signal prediction (with appropriate camera parameters per frame). Then only remaining prediction residuals need to be coded for transmission per frame, potentially resulting in high coding performance.

The crux of this 3D-model-based approach is the efficient coding of the 3D model as *side information* (SI). In this paper, we reconstruct and code a 3D planar model from a group of frames in a rate-distortion (RD) optimized manner towards higher video compression efficiency. Specifically, we first reconstruct a 2D planar model from a pre-selected spatial region within a video using state-of-the-art SfM techniques. Given the planar model, we then map pixels in the frame group to a 2D grid on the plane with a density that trades off the quality of signal prediction using this model and the cost of coding the grid pixels. The boundary of the irregularly shaped pixel body on the plane is coded using *arithmetic edge coding* (AEC) [11], and then the body is encapsulated into a tight-fitting rectangular region, which is encoded as an intra-frame using HEVC [2]. The pixels that are inside the rectangle but outside the pixel body—called *don't care region* (DCR)—are filled optimally by minimizing an l_1-norm of the transform coefficients via a linear programming (LP) formulation [12]. Experimental results show that: i) optimal filling of DCR pixels improves coding efficiency of pixels on the 2D plane, and ii) the RD-optimized planar model improves compression performance over native HEVC implementation.

The outline of this paper is as follows. We discuss related works in Section 2. We overview our proposed video coding system in Section 3, including scene reconstruction, planar modeling, model optimization and coding. We discuss details of planar model optimization and coding in Section 4. Finally, results and conclusion are presented in Section 5 and 6, respectively.

2. RELATED WORK

Although model-based video coding [3–5, 13–15] has been studied for over a decade, the trade-off between model accuracy and coding cost remains a challenge. In [13], a 3D surface is estimated for encoding depth videos. However, coding cost for a general surface in [13] is much more expensive than our proposed planar model. A 3D planar approximation scheme for color-plus-depth videos is proposed in [15], and an RD-efficient piecewise planar scene modeling approach using 2D color images is proposed in [14]. However, [15] cannot be used for video sequences with no depth information. [14] constructs a scene using a mesh, where the number of vertices in the mesh can be very large and wasteful if the spatial region of interest is roughly a flat plane, such as the side of a building. Unlike mesh-based methods [13–15] that typically have high coding rates for scenes with roughly flat regions, our approach focuses on an appropriate planar modeling scheme to avoid large coding cost.

[1] https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

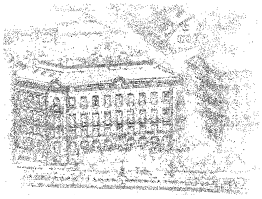
[2] Multiple reference frame approaches [6] still exploit only pairwise redundancy, but at a block level.

For practical planar-based video coding, an appropriate grid density setting is required on the planar model for optimal coding gain. An object tracking based video coding scheme with rigid object assumption is proposed in [10]. However, such model-based video coding scheme does not optimize the planar grid density. We differ from [10] and [13–15] as follows: i) we trade off optimally between reconstruction quality and SI cost that includes planar grid density, and ii) we optimize coding of the 2D grid pixels by exploiting freedom in choosing DCR pixels.

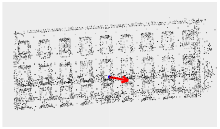
SfM techniques [7–9] return accurate scene reconstruction and have been widely used in both off-line image rendering [8] and Simultaneous Localization and Mapping (SLAM) [16–18]. We stress that we apply state-of-the-art SfM [19] methods just for scene reconstruction (the first step) in our proposed video coding system. Specifically, we use the SfM scene reconstruction results, *i.e.*, a set of sparse point cloud and a set of camera parameters that correspond to each of the input video frames, for planar modeling.

3. SYSTEM OVERVIEW

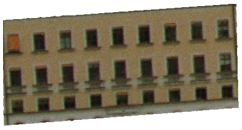
(a) Sample frame.



(b) Point cloud.



(c) Fitted plane.



(d) Planar model.

Fig. 1. Example of model reconstruction using `old_town_cross`: (a) Sample original frame, (b) COLMAP SfM-derived point cloud, (c) a fitted plane shown in grey with its normal indicated by a red arrow; points that are behind the fitted plane are rendered with lighter grey, and (d) a sample planar model.

Our proposed video coding system includes the following modules: scene reconstruction, planar modeling, model optimization and coding. We first model the scene as a 2D plane in 3D space. We accomplish this by using a state-of-the-art SfM algorithm called COLMAP [7, 20] that has demonstrated superior performance against competing methods [19]. Since the input video frames are distorted and their intrinsic camera parameters are unknown, we choose a pinhole camera model with radial distortion for COLMAP, where COLMAP simplifies the corresponding opencv [21] pinhole camera model and models the radial distortion effect with only one radial distortion coefficient. COLMAP performs the SfM process by: 1) correspondence search, including SIFT [22] feature detection and exhaustive feature matching with geometric verification, and 2) incremental sparse reconstruction, which is implemented by bundle adjustment [23, 24] with Ceres Solver [25].

Taking a video sequence, *e.g.*, `old_town_cross`, as the input with sample frame shown in Fig. 1(a), COLMAP returns a sparse point cloud representing the reconstructed 3D scene, shown in Fig. 1(b), and a set of camera parameters that correspond to each of the input video frames, where each point in the point cloud corresponds to a feature point in the input video frames.

For planar modeling, we first designate a roughly flat spatial region of interest (ROI) in the point cloud, *e.g.*, the front part of the yellow house, to be plane-fitted. To do this, we first select four points in the point cloud that cover the ROI, then find the corresponding ROI's within each of the input video frames. We can now extract a subset of point cloud that belongs to the ROI, fit a 2D plane in the 3D space by using RANSAC [26], shown in Fig. 1 (c), and find the homography [27] between the fitted plane and each of the input video frames, again, using RANSAC. Next, we project the pixels in the ROI's of all input video frames to the fitted plane based on the previously computed homography, shown in Fig. 1(d).

The system workflow is shown in Fig. 2. We discuss planar model and video coding in details in the next section.

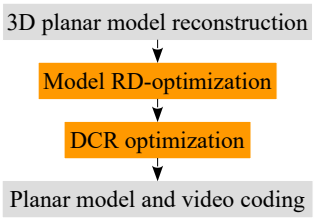


Fig. 2. Proposed 3D model-based video coding scheme.

4. PLANAR MODEL AND VIDEO CODING

For efficient coding of the planar model, we first set an appropriate grid density of the fitted plane, and map pixels from the input video frames to the fitted plane (pixels rounded to the same 2D grid locations are averaged). We then code the boundary of the irregularly shaped pixel body on the plane using AEC [11] and encapsulate the planar model into a tight-fitting rectangular region; pixels inside the rectangle but outside the body are called *DCR pixels*. The rectangular region is encoded as an intra-frame using HEVC. We complete the DCR pixels by minimizing an l_1-norm of the transform codffi-cients via an LP formulation. Finally, we take the AEC-coded plane boundary, HEVC-encoded plane texture and frame camera parameters as SI, and project all pixels within the HEVC-encoded planar model back to corresponding spatial regions in the video frames as prediction, so that prediction residual can be coded again using HEVC. We discuss our planar model RD-optimization and DCR optimization scheme, planar model coding, and video coding in detail next.

4.1. Model RD-Optimization

Over 40 million pixels from all original frames mapped to the fitted plane for the tested region in `old_town_cross`—it is impractical to encode the planar model without setting up an appropriate grid density even for a small video sequence. Thus, an RD-optimal planar model grid density is needed. We find such a grid density using a pre-set increment parameter γ. Specifically, we denote the pixel locations on the planar model before setting up a grid as $(x_i, y_i), i = 1, ..., N$, the ones after as $[(\text{round}(x_i\gamma), \text{round}(y_i\gamma)]$, where N denotes the total pixel number on the planar model before grid setting. Note that $[(\text{round}(x_i\gamma), \text{round}(y_i\gamma)]$ may contain repetitive pixel locations where different pixels are rounded to the same grid locations. In this case, we assign such gridded pixel locations

with the averaged intensity of these overlapped pixels, *i.e.*, the total number of gridded pixel locations $M \leq N$. For RD-optimal grid density, we expect that $M \ll N$ and saturation of the number of occupied pixel locations in reconstructed images as γ increases. As shown in Fig. 3, the resulting mean PSNR of the reconstructed images is consistent with the above saturation trend. In practice, we choose γ (highlighted as squares in Fig. 3) when the percentage of the number of occupied pixel locations in reconstructed images changes within a pre-set small tolerance. In our experiment with `old_town_cross`, the RD-optimal grid setting results in only $M/N \approx 0.5\%$ of total number of mapped pixels.

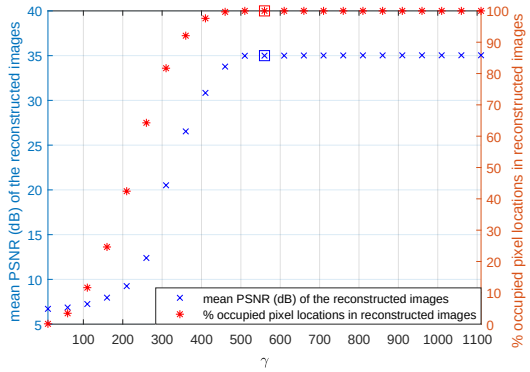


Fig. 3. Grid density increment parameter γ for the tested region in `old_town_cross`.

4.2. DCR Optimization and Planar Model Coding

For a given n-pixel block $\mathbf{x}$ on the boundary of the irregularly shaped pixel body on the 2D plane, let $\mathbf{D}$ be a $m \times n$ *sampling matrix* that extracts m pixels *inside* the body from the block, which is represented by $\mathbf{y}$. While the $n - m$ block pixels outside the body—*i.e.*, DCR pixels—can be freely chosen, we enforce the constraint that the original m pixels must be maintained during optimization.

Denote by $\mathbf{z}$ the intra-prediction signal for the block generated by HEVC, and by $\mathbf{\Phi}$ the transform (*e.g.* discrete cosine transform (DCT) commonly used in coding standards) used for transform coding. In general, a sparse signal representation in the transform domain leads to a low coding cost. We can thus design an l_0-norm objective and formulate the following optimization:

$$\min_{\mathbf{x}} \|\mathbf{\Phi}(\mathbf{x}-\mathbf{z})\|_0 \quad \text{s.t. } \mathbf{D}\mathbf{x} = \mathbf{y} \tag{1}$$

where the objective is the number of non-zero transform coefficients for the prediction residual $\mathbf{x} - \mathbf{z}$.

Because l_0-norm is non-convex, we optimize its convex relaxation l_1-norm instead, resulting in:

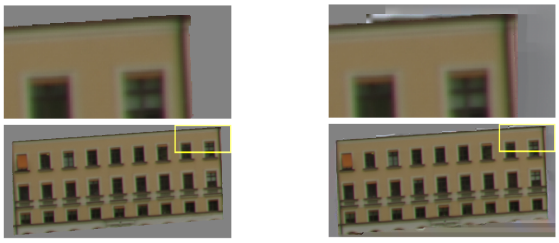
$$\min_{\mathbf{x}} \|\mathbf{\Phi}(\mathbf{x}-\mathbf{z})\|_1 \quad \text{s.t. } \mathbf{D}\mathbf{x} = \mathbf{y} \tag{2}$$

There exist many algorithms in the optimization literature such as proximal gradient [28] to minimize non-smooth l_1-norm in (2). Instead, we reformulate (2) into a simpler linear programming (LP) formulation as follows:

$$\min_{\mathbf{x},\mathbf{u}} \mathbf{1}^\top \mathbf{u} \quad \text{s.t.} \begin{cases} -\mathbf{u} \leq \mathbf{\Phi}(\mathbf{x}-\mathbf{z}) \leq \mathbf{u}, \\ \mathbf{D}\mathbf{x} = \mathbf{y}, \\ \mathbf{u} \geq \mathbf{0}. \end{cases} \tag{3}$$

LP problems like (3) can be solved efficiently using many known methods like the Simplex algorithm [12].

Fig. 4(b) shows a sample DCR-optimized planar model. Fig. 4(a) shows the input of our DCR optimization scheme where DCR pixels are filled with value 128. We show in Section 5.1 that our DCR optimization scheme does lead to planar model coding gain.



(a) All-128. (b) DCR-optimized.

Fig. 4. DCR optimization using `old_town_cross` at QP=2. The upper half of both (a) and (b) show the enlarged upper right portion of the planar model highlighted by yellow rectangles.

4.3. Video Coding

For our proposed planar model for video coding, SI includes AEC-coded [11, 29] plane boundary I_b, HEVC-encoded plane texture I_t and frame camera parameters. Specifically, the camera parameters of each video frame contain two parts: 1) intrinsic parameters that is shared for all frames in the same video, including a single focal length, two-parameter principal point coordinates and a single radial distortion coefficient; and 2) extrinsic parameters, including a four-parameter quaternion [20] that includes the camera rotation information and a three-parameter camera translation vector. Hence, the total SI cost T in bits is given by: $T = I_b + I_t + 16(4 + 7L)$, where L is the total frame number. We project all pixels within the HEVC-encoded planar model back to each of the spatial regions in the input video frames as prediction, and the prediction residuals are coded using HEVC.

5. EXPERIMENTATION

We tested two video sequences on a Windows 10 PC with Intel Core i5-7500 and 8GB of RAM: 1) `old_town_cross` with 400 frames, 50 fps, and a 960x768 tested region, and 2) `city` with 300 frames, 60 fps, and a 384x192 tested region. The tested regions are shown in Fig. 5. We use HM-16.4-4432 for planar model coding and original video coding, and a customized HM-16.6-JEM-6.0-based version for our proposed video coding system. We present planar model and video coding results next.

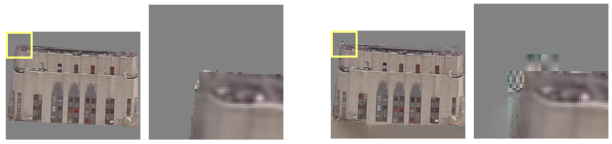
5.1. Planar Model Coding

Before planar model coding, we apply our DCR optimization scheme in (3) in Section 4.2 to the encapsulated planar models of both tested video regions. For implementation of (3), we first obtain a set of HEVC intra modes for all blocks of the input image, and then iterate the process of getting HEVC predictors with the same set of intra modes and (3) until all blocks that are at the boundary of the planar model are DCR-optimized. The DCR optimized planar model for `old_town_cross` is shown in Fig. 4, and `city` in Fig. 6.

(a) old_town_cross. (b) city.

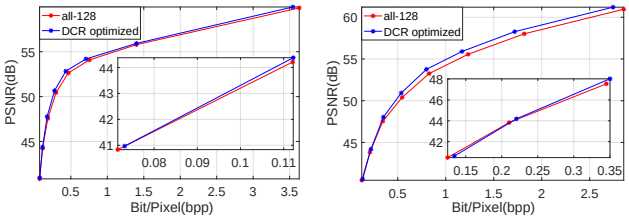
Fig. 5. Tested regions in old_town_cross and city.

The planar model coding RD performance is shown in Fig. 7. The tested QP's are 2, 7, 12, 17, 22, 27, 32 and 37. We also calculate Bjøntegaard Delta Bitrate (BD-rate) reduction [30], the bit-rate reduction of our DCR optimization scheme compared to the all-128 baseline given the same image quality, and Bjøntegaard Delta PSNR (BD-PSNR) gain, both of which are shown in Table. 1. Both Fig. 7 and Table. 1 show consistent performance gain compared to the all-128 baseline, except that our proposed scheme for city has a slight performance drop at QP=37.



(a) All-128. (b) DCR-optimized.

Fig. 6. DCR optimization using city at QP=2. The right half of both (a) and (b) show the enlarged upper left portion of the planar model highlighted by yellow rectangles.



(a) old_town_cross. (b) city.

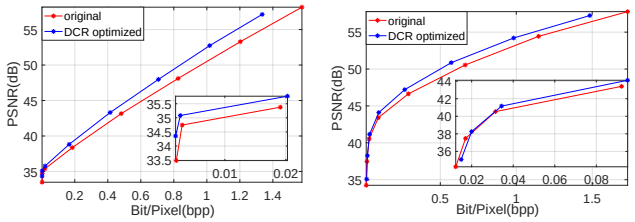
Fig. 7. Planar model coding RD performance.

5.2. Video Coding

For video coding, we compare the performance of original video coding using HM-16.4-4432 and our proposed video coding system using a customized HM-16.6-JEM-6.0-based version with DCR-optimized, HEVC-encoded planar models. The video coding RD performance using the same QP's as planar model coding is shown in Fig. 7, and the corresponding BD-rate is shown in Table. 2. Both Fig. 7 and Table. 2 show consistent performance gain by our proposed coding system compared to the original baseline, except that our proposed system for city has a slight performance drop at QP=37. Note that, the total SI cost T (see Section 4.3) is incorporated into the total bit-rate count for our proposed video coding scheme.

Table 1. Planar coding BD-rate reduction and BD-PSNR gain.

planar	Y	U	V
old_town_cross	-7.45%	-6.84%	-6.12%
	0.38dB	0.29dB	0.26dB
city	-3.65%	-5.01%	-10.58%
	0.31dB	0.32dB	0.68dB



(a) old_town_cross. (b) city.

Fig. 8. Video coding RD performance.

Table 2. Video coding BD-rate reduction and BD-PSNR gain.

video	Y	U	V
old_town_cross	-15.72%	-20.20%	-20.89%
	0.74dB	0.86dB	0.78dB
city	-11.72%	-22.99%	-21.03%
	0.64dB	0.69dB	0.69dB

6. CONCLUSION

RD-optmization in model-based video coding remains a challenge. In this paper, we propose an RD-optimized 2D planar-model-based video coding system that optimally trades off between quality of prediction by the planar model and its coding cost. Specifically, after reconstructing a planar model using existing SfM techniques, we select an appropriate density for a 2D grid on the plane. The irregularly shaped pixel body (whose shape is coded using AEC) is encapsulated inside a tightly-fitted rectangular region, where pixels inside the rectangle but outside the body called DCR pixels are optimized by minimizing an l_1-norm of the transform coefficients using linear programming. Experiments confirm that our proposed DCR optimization scheme improves planar model coding efficiency, and our RD-optimized planar model improves compression performance over a native HEVC implementation. We stress that our proposed single planar modelling scheme in our video coding system mainly handles roughly flat spatial regions in the scene. Future work will focus on RD-optimized multiple-planar modelling that incorporates both local and global scene modelling cost for efficient video coding with smaller distortion.

7. REFERENCES

[1] T. Wiegand et al., "Overview of the H.264/AVC video coding standard," *IEEE TCSVT*, vol. 13, no. 7, pp. 560–576, July 2003.

[2] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," in *IEEE TCSVT*, vol. 22, no.12, December 2012.

[3] H. G. Musmann, M. Htter, and J. Ostermann, "Object-oriented analysis-synthesis coding of moving images," *SPIC*, vol. 1, no. 2, pp. 117 – 138, 1989.

[4] D. E. Pearson, "Developments in model-based video coding," *Proceedings of the IEEE*, vol. 83, no. 6, pp. 892–906, Jun 1995.

[5] F. C. M. Martins and J. M. F. Moura, "Video representation with three-dimensional entities," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 1, pp. 71–85, Jan 1998.

[6] G. Cheung, W.-T. Tan, and C. Chan, "Reference frame optimization for multiple-path video streaming with complexity scaling," in *IEEE TCSVT*, vol. 17, no.6, June 2007, pp. 649–662.

[7] J. L. Schnberger and J. M. Frahm, "Structure-from-motion revisited," in *IEEE CVPR*, Las Vegas, NV, June 2016.

[8] S. N. Sinha et al., "Piecewise planar stereo for image-based rendering," in *IEEE ICCV*, Kyoto, Japan, Sept 2009.

[9] Y. Furukawa et al., "Manhattan-world stereo," in *IEEE CVPR*, Miami, FL, June 2009.

[10] S. Takamura and A. Shimizu, "Efficient video coding using rigid object tracking," in *APSIPA ASC*, Kuala Lumpur, Malaysia, Dec 2017.

[11] I. Daribo et al., "Arbitrarily shaped motion prediction for depth video compression using arithmetic edge coding," *IEEE TIP*, vol. 23, no. 11, pp. 4696–4708, Nov 2014.

[12] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[13] W. Sun et al., "Rate-constrained 3D surface estimation from noise-corrupted multiview depth videos," *IEEE TIP*, vol. 23, no. 7, pp. 3138–3151, July 2014.

[14] E. Imre et al., "Rate-distortion efficient piecewise planar 3-D scene representation from 2-D images," *IEEE TIP*, vol. 18, no. 3, pp. 483–494, March 2009.

[15] B. O. Ozkalayci and A. A. Alatan, "3D planar representation of stereo depth images for 3DTV applications," *IEEE TIP*, vol. 23, no. 12, pp. 5222–5232, Dec 2014.

[16] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics Automation Magazine*, vol. 13, no. 2, pp. 99–110, June 2006.

[17] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): part II," *IEEE Robotics Automation Magazine*, vol. 13, no. 3, pp. 108–117, Sept 2006.

[18] C. Cadena et al., "Simultaneous localization and mapping: Present, future, and the robust-perception age," *CoRR*, vol. abs/1606.05830, 2016.

[19] T. Schops et al., "A multi-view stereo benchmark with high-resolution images and multi-camera videos," in *IEEE CVPR*, Honolulu, HI, July 2017.

[20] J. L. Schoenberger, "COLMAP," http://colmap.github.io/, accessed Feb. 11, 2018.

[21] opencv dev team, "Opencv 2.4.13.5 documentation," https://docs.opencv.org/2.4/, accessed Feb. 11, 2018.

[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[23] D. C. Brown, "The bundle adjustment - progress and prospects," *International Archives of Photogrammetry Paper*, vol. 21, pp. pgs. 29–33, 1976.

[24] B. Triggs, P. F. Mclauchlan, R. I. Hartley, and A. W. Fitzgibbon, "Bundle adjustment - a modern synthesis," *Vision Algorithms Theory & Practice*, 2000.

[25] S. Agarwal, K. Mierle *et al.*, "Ceres solver," http://ceres-solver.org, accessed Feb. 11, 2018.

[26] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[27] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2003.

[28] N. Parikh and S. Boyd, "Proximal algorithms," in *Foundations and Trends in Optimization*, vol. 1, no.3, 2013, pp. 123–231.

[29] A. Zheng, G. Cheung, and D. Florencio, "Context tree-based image contour coding using a geometric prior," *IEEE TIP*, vol. 26, no. 2, pp. 574–589, Feb 2017.

[30] G. Bjontegaard, "Calculation of average psnr differences between r-d curves," in *Proceedings of the 13-th VCEG Meeting*, Austin, TX, April 2001.