Proceeding of the IEEE
International Conference on Robotics and Biomimetics
Dali, China, December 2019

Fast ADMM ℓ_1 minimization by applying SMW formula and multi-row simultaneous estimation for Light Transport Matrix acquisition*

Naoya Chiba
Graduate School of
Information Sciences
Tohoku University
Sendai, Miyagi, Japan
chiba@ic.is.tohoku.ac.jp

Akira Imakura
Faculty of Engineering,
Information and Systems
University of Tsukuba
Tsukuba, Ibaraki, Japan
imakura@cs.tsukuba.ac.jp

Koichi Hashimoto
Graduate School of
Information Sciences
Tohoku University
Sendai, Miyagi, Japan
koichi@tohoku.ac.jp

***Abstract*— The Light Transport Matrix (LTM) is a fundamental expression of the light propagation of the projector-camera system. The matrix includes all the characteristics of light rays transferred from the projector to the camera, and it is used for scene relighting, understanding the light path, and 3D measurement. Especially, LTM enables robust 3D measurement even if the scene includes metallic or semi-transparent objects; thus it is already used for robot vision. The LTM is often estimated by ℓ_1 minimization because the LTM has a huge number of elements. ℓ_1 minimization methods, which utilize the Alternating Direction Method of Multipliers (ADMM), can reduce the number of observations. In addition, a powerful extended ADMM ℓ_1 minimization method named Saturation ADMM, which can estimate the LTM under saturated conditions, also exists. In the study presented in this paper, we reduce the computational cost of ADMM ℓ_1 minimization by applying the Sherman-Morrison-Woodbury (SMW) formula. Furthermore, we propose "multi-row simultaneous LTM estimation," which is a new method to improve the computational efficiency. The contribution of this paper is to propose the use of these two methods to speed up LTM estimation and demonstrate that our methods reduce the computational cost in theory and the calculation time in practice. Experiments indicate that our method accelerates ADMM ℓ_1 minimization by up to 4.64 times, and Saturation ADMM ℓ_1 minimization by up to 2.54 times compared to the original methods.**

***Index Terms*— 3D Measurement, Light Transport Matrix, Alternating Direction Method of Multipliers (ADMM)**

I. Introduction

Light transport refers to the behavior of the light rays propagated between a light source and an optical sensor. In this regard, we assume a Projector-Camera System, which consists of a projector and a camera, as an optical system. Light transport in a projector-camera system can be modeled as the Light Transport Matrix (LTM) when the intensity response function is linearized [23, 24]. This matrix can be considered as representing the impulse responses between the projector and the camera; thus, it includes all information about the projector-camera system that can describe a linear relation. Typical applications of the LTM are scene relighting [8, 12, 13, 24], understanding the light path (e.g. direct/global illumination separation) [1, 15, 18, 19, 20], and 3D measurement [4, 5, 6, 9, 18, 20]. Especially, LTM enables robust 3D measurement even if the scene includes metallic or semi-transparent objects; thus it is already used for robot vision.

The LTM contains a large number of elements because it is a fundamental expression; therefore, measuring the elements of the LTM directly is highly time-consuming. The elements describe the light propagation weight between a projector pixel and a camera pixel; therefore, because almost all LTM elements are zero, the LTM can be assumed to be a sparse matrix. The use of sparse estimation (compressive sensing) techniques [3, 10] enables the LTM to be estimated using fewer observations. LTM estimation techniques have been the subject of various studies [5, 6, 8, 9, 13, 21, 22].

The LTM is often estimated by using Regularized Orthogonal Matching Pursuit (ROMP) [16, 17]. Another well-known and powerful ℓ_1 minimization method is the Alternating Direction Method of Multipliers (ADMM) [2]. Compared to ROMP, this method requires a smaller number of observations to estimate a sparse vector. For example, the number of the projections/captures ranged from 512 to 1990 images in [21], 991 images in [22], and 900 images in [13]. Although these methods require approximately 1,000 images, ADMM can estimate the LTM with 113 projections/captures [9]. (It should be noted that it is difficult to compare these methods in a simple way because the projector-camera systems have different resolutions and color/grayscale characteristics, but ADMM is capable of estimating a sparse LTM with fewer projections/captures than ROMP). Additionally, an extended method of ADMM ℓ_1 minimization, Saturation ADMM [8] can estimate the LTM even if the measurement scene includes saturation. These considerations motivated us to attempt to

* This work was supported by ACT-I, JST Grant Numbers JPMJPR16UH, JPMJPR16U6, and JSPS KAKENHI Grant Numbers JP16H06536, 17K12690, JP18J20111.

978-1-7281-6321-5/19//$31.00 ©2019 IEEE

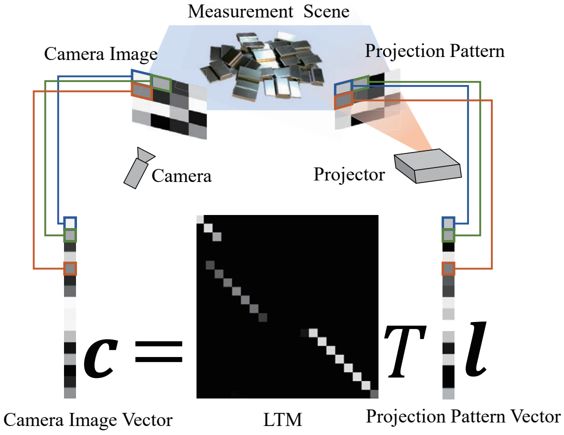


Fig. 1: LTM model adapted from [8]. We define $\boldsymbol{c}$ and $\boldsymbol{l}$ as the vectorized camera intensities and vectorized projector intensities, respectively. The LTM T describes the light ray propagation between $\boldsymbol{c}$ and $\boldsymbol{l}$. In other words, the relationship between these vectors is given by weights, which are elements of the LTM. Once we provide a projection pattern vector, we can easily calculate the camera image vector by using matrix-vector multiplication.

speed up ADMM. Multi-scale LTM estimation [9] utilizes ADMM with a coarse-to-fine approach. This approach significantly reduces the computational cost, and it enables high-resolution (e.g., $R = 256$ in [9], or $R = 128$ in [8] with R being the number of projector/camera pixels.) LTM sparse estimation by using ADMM.

Although the computational cost is reduced by using multi-scale LTM estimation, obtaining the LTM still requires 342 seconds [9]. In this study, we apply the Sherman-Morrison-Woodbury (SMW) formula to ADMM and Saturation ADMM ℓ_1 minimization to further reduce the computational cost. Additionally, we propose a new approach to speed up the computation, multi-row simultaneous estimation, as an extension of multi-scale LTM estimation. These methods reduce the calculation time without any deterioration of the accuracy.

II. LTM ESTIMATION BY USING ℓ_1 MINIMIZATION VIA ADMM

In this section, we describe the Light Transport Matrix (LTM) model as an optical model of the projector-camera system, and the LTM sparse estimation problem. Then, we mention LTM estimation by using ℓ_1 minimization via ADMM. We also introduce Saturation ADMM, which is a powerful variation of ADMM that can estimate the LTM under saturated conditions [8].

A. LTM estimation problem

We first introduce LTM as a model of light transport for a projector-camera system [23, 24]. Let $\boldsymbol{l}$ and $\tilde{\boldsymbol{c}}$ be a vectorized projection pattern and a vectorized camera image, respectively. Here, we assume the resolution of the projector and the camera are $R \times R$; thus, the number of elements of $\boldsymbol{l}$ and $\tilde{\boldsymbol{c}}$ are R^2. When a projection pattern $\boldsymbol{l}$ is projected from the projector, the light rays reflect on the scene, and finally, some of them are captured by the camera as a camera image vector $\tilde{\boldsymbol{c}}$. This light transportation is denoted by

$$\tilde{\boldsymbol{c}} = T\boldsymbol{l} + \boldsymbol{b}, \tag{1}$$

where T is the LTM which is determined by the scene and the projector-camera system, and $\boldsymbol{b}$ is the background lights. The LTM is known to be a sparse matrix [5, 6, 13]. In the following, we consider $\boldsymbol{b}$ to be the constant; thus, we define a camera image vector without the background lights as $\boldsymbol{c} = \tilde{\boldsymbol{c}} - \boldsymbol{b}$, and we simply call $\boldsymbol{c}$ as the camera image vector. Here, (1) is written as

$$\boldsymbol{c} = T\boldsymbol{l} \tag{2}$$

(see Fig. 1).

To apply sparse estimation to obtain the LTM, we introduce multiple projections and captures. We denote N as the number of projections and captures, and n as its index. The n-th projection pattern vector and the n-th camera image vector are given as $\boldsymbol{l}_n$ and $\boldsymbol{c}_n$, respectively. Let a projection pattern matrix L be $L = \begin{bmatrix}\boldsymbol{l}_1 & \boldsymbol{l}_2 & \cdots & \boldsymbol{l}_N\end{bmatrix}$, and let a camera image matrix C be $C = \begin{bmatrix}\boldsymbol{c}_1 & \boldsymbol{c}_2 & \cdots & \boldsymbol{c}_N\end{bmatrix}$. With L and C, all relations between projections and captures are written as

$$C = TL. \tag{3}$$

The LTM estimation problem can be formulated as a general ℓ_1 minimization problem [5, 13]. We follow the notations that were used in the study [8]. The i-th row of C and T are denoted as $\bar{\boldsymbol{c}}_i$ and $\bar{\boldsymbol{t}}_i$, respectively. Here, a row of (3) is written by $\bar{\boldsymbol{c}}_i = \bar{\boldsymbol{t}}_i L$, and its transpose is given by $\bar{\boldsymbol{c}}_i^\top = L^\top \bar{\boldsymbol{t}}_i^\top$. The sparse estimation of $\bar{\boldsymbol{l}}_i^\top$ can be formulated as an ℓ_1 minimization as

$$\min_{\bar{\boldsymbol{t}}_i} \left\{ \left\| \bar{\boldsymbol{t}}_i^\top \right\|_1 + \frac{1}{2\lambda} \left\| \bar{\boldsymbol{c}}_i^\top - L^\top \bar{\boldsymbol{t}}_i^\top \right\|_2^2 \right\}, \tag{4}$$

where λ is a parameter of the ℓ_1 regularizer. (4) can be written as a general ℓ_1 minimization problem

$$\min_{\boldsymbol{x}} \left\{ \|\boldsymbol{x}\|_1 + \frac{1}{2\lambda} \|\boldsymbol{y} - A\boldsymbol{x}\|_2^2 \right\}, \tag{5}$$

with the notations: $\boldsymbol{x} = \bar{\boldsymbol{t}}_i^\top$, $\boldsymbol{y} = \bar{\boldsymbol{c}}_i^\top$, and $A = L^\top$. Here, the dimensions of these notations are $\boldsymbol{x} \in \mathbb{R}^{R^2}$, $\boldsymbol{y} \in \mathbb{R}^N$, and $A \in \mathbb{R}^{N \times R^2}$. Finally, it was previously concluded [5] that the LTM estimation problem can be solved by repeating this ℓ_1 minimization problem for each row of (3).

B. ℓ_1 minimization by using ADMM

The Alternating Direction Method of Multipliers (ADMM) [2] is a powerful optimization method that is applicable to the ℓ_1 minimization problem. For the constraints $\boldsymbol{x} - \boldsymbol{z} = \mathbf{0}$, a Lagrangian multiplier $\boldsymbol{h}$ and ℓ_2 penalty weight μ are introduced.

ADMM ℓ_1 minimization is separated into two parts: Pre-calculation and Post-calculation. Pre-calculation depends only on the matrix A; therefore, Pre-calculation is accomplished offline, and it does not require any observation. Post-calculation requires $\boldsymbol{x}$ and $\boldsymbol{y}$; thus, it should be calculated after the observations. Additionally, Post-calculation includes J iterative computations when processing itself.

For the following formulation, we introduce the Soft Thresholding Function $\mathit{SoftThr}(x;\theta)$ as

$$\mathit{SoftThr}(x;\theta) = \begin{cases} x-\theta & (\theta \le x) \\ 0 & (-\theta \le x < \theta) \\ x+\theta & (x \le -\theta) \end{cases}. \tag{6}$$

When the Soft Thresholding Function is applied to a vector or a matrix, it is applied to each element, respectively. We also denote a vector $\boldsymbol{u}$ as $\boldsymbol{u} = \boldsymbol{h}/\mu$ for simplicity.

The Pre-calculation of ADMM ℓ_1 minimization calculates

$$M_1 \leftarrow \left(I + \frac{1}{\mu\lambda} A^\top A\right)^{-1}, \tag{7}$$

and

$$M_2 \leftarrow \frac{1}{\mu\lambda} M_1 A^\top, \tag{8}$$

where I is the identity matrix. Then, the Post-calculation computes

$$\boldsymbol{M}_3 \leftarrow M_2 \boldsymbol{y}, \tag{9}$$

and the following equations are computed T times iteratively:

$$\begin{cases} \boldsymbol{x} \leftarrow \boldsymbol{M}_3 + M_1(\boldsymbol{z} - \boldsymbol{u}) \\ \boldsymbol{z} \leftarrow \mathit{SoftThr}\left(\boldsymbol{u} + \boldsymbol{x}; \frac{1}{\mu}\right) \\ \boldsymbol{u} \leftarrow \boldsymbol{u} + \boldsymbol{x} - \boldsymbol{z} \end{cases}. \tag{10}$$

In all of the above calculations, the computational cost of ADMM ℓ_1 minimization is $\mathcal{O}(R^6 + R^4 N + R^4 J)$.

C. ℓ_1 minimization by using Saturation ADMM

The LTM model assumes that the relationship between the camera intensities and the projector intensities is linear; however, the real relationship includes nonlinearities because of the saturation [8]. Hence, the ℓ_1 minimization should be solved by considering the saturation of the observed camera images. To describe the nonlinear behavior, the Clipping Function $\mathit{Clip}(y;Y)$ is introduced as

$$\mathit{Clip}(y;Y) = \begin{cases} y & (y \le Y) \\ Y & (y > Y) \end{cases}. \tag{11}$$

The Clipping Function can be applied to a vector or a matrix in the same way as the Soft Thresholding Function. When using the Clipping Function, the ℓ_1 minimization problem under saturated conditions is described as

$$\min_{\boldsymbol{x}} \left\{ \|\boldsymbol{x}\|_1 + \frac{1}{2\lambda} \|\boldsymbol{y} - \mathit{Clip}(A\boldsymbol{x}; Y\mathbf{1})\|_2^2 \right\}. \tag{12}$$

Saturation ADMM, which is an extension of ADMM ℓ_1 minimization, and which also consists of Pre-calculation and Post-calculation, was recently introduced [8].

For the Pre-calculation,

$$M_1 \leftarrow \left(I + \frac{\nu}{\mu} A^\top A\right)^{-1} \tag{13}$$

and

$$M_2 \leftarrow \frac{\nu}{\mu} M_1 A^\top \tag{14}$$

are calculated, where, μ and ν are the penalty weights. Then, for the Post-calculation, the following equations are computed by using J iterative cycles:

$$\begin{cases} \boldsymbol{x} \leftarrow M_1(\boldsymbol{z} - \boldsymbol{u}) + M_2(\boldsymbol{\zeta} - \boldsymbol{v}) \\ \boldsymbol{z} \leftarrow \mathit{SoftThr}\left(\boldsymbol{u} + \boldsymbol{x}; \frac{1}{\mu}\right) \\ \boldsymbol{\xi} \leftarrow A\boldsymbol{x} + \boldsymbol{v} \\ \boldsymbol{\zeta} \leftarrow D\left(\boldsymbol{\xi}; \left(\sqrt{\frac{1+\nu\lambda}{\nu\lambda}} - 1\right)\boldsymbol{y} + \sqrt{\frac{1+\nu\lambda}{\nu\lambda}} Y\mathbf{1}, \right. \\ \qquad \left. \frac{1}{1+\lambda\nu}\boldsymbol{y} + \frac{\lambda\nu}{1+\lambda\nu}\boldsymbol{\xi}\right) \\ \boldsymbol{v} \leftarrow \boldsymbol{\xi} - \boldsymbol{\zeta} \\ \boldsymbol{u} \leftarrow \boldsymbol{u} + \boldsymbol{x} - \boldsymbol{z} \end{cases}, \tag{15}$$

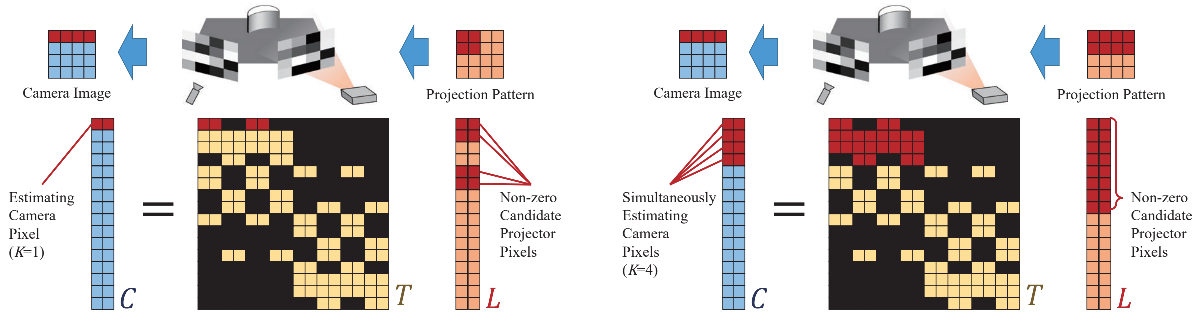
where $\mathbf{1}$ is a vector of which the elements are all 1, and a function $D(x;\theta,\phi)$ that is defined by

$$D(x;\theta,\phi) = \begin{cases} \phi & (x \le \theta) \\ x & (x > \theta) \end{cases}. \tag{16}$$

The function $D(x;\theta,\phi)$ can be applied to a vector or a matrix in a similar way as the Soft Thresholding Function and the Clipping Function. The computational cost of Saturation ADMM ℓ_1 minimization is $\mathcal{O}(R^6 + R^4 N + R^4 J)$, which is the same as ADMM ℓ_1 minimization.

III. Application of SMW formula for ADMM/Saturation ADMM

In this section, we explain the Sherman-Morrison-Woodbury (SMW) formula and its application to ADMM and Saturation ADMM. We also estimate the extent to which the computational cost decreases by applying this formula.


Camera Image
Projection Pattern
Estimating Camera Pixel (K=1)
=
Non-zero Candidate Projector Pixels
C
T
L
Camera Image
Projection Pattern
Simultaneously Estimating Camera Pixels (K=4)
=
Non-zero Candidate Projector Pixels
C
T
L

(a) Original multi-scale LTM estimation ($K = 1$)

(b) Multi-row simultaneous estimation ($K = 4$)

Fig. 2: Illustrative representation of multi-row simultaneous estimation. Black elements on LTM T indicate non-zero elements that are estimated in the previous resolution. (a) When we focus on a camera pixel, some projector pixels can be considered as non-zero candidate projector pixels. (b) Multi-row simultaneous estimation estimates some camera pixels together; thus, the number of non-zero candidate projector pixels often increases. However, we can reduce the number of computations of the Pre-calculation such that it becomes as small as $1/K$ times.

A. SMW-ADMM

The SMW formula is a well-known formula in linear algebra, and is named after J. Sherman, W. J. Morrison, and M. A. Woodbury [11]. This formula can reduce the computational cost of the inverse of a matrix.

The SMW formula is written as

$$
\begin{aligned}
&(X_1 + X_2X_3X_4)^{-1} \\
=&X_1^{-1} - X_1^{-1}X_2X_3\left(X_3 + X_3X_4X_1^{-1}X_2X_3\right)^{-1}X_3X_4X_1^{-1}.
\end{aligned} \tag{17}
$$

By applying the SMW formula, (7) is

$$
\left(I + \frac{1}{\mu\lambda}A^\top A\right)^{-1} = I - \frac{1}{\mu\lambda}A^\top\left(I + \frac{1}{\mu\lambda}AA^\top\right)^{-1}A. \tag{18}
$$

By using this knowledge, we propose ADMM ℓ_1 minimization via the SMW formula (SMW-ADMM) in this subsection.

The Pre-calculation of SMW-ADMM computes

$$
M_1 = \frac{1}{\mu\lambda}AA^\top, \tag{19}
$$

$$
M_2 = \frac{1}{\mu\lambda}\left(I + M_1\right)^{-1}, \tag{20}
$$

and

$$
M_3 = \frac{1}{\mu\lambda}I - M_2M_1. \tag{21}
$$

Then, the Post-calculation of SMW-ADMM calculates

$$
\boldsymbol{M}_4 \leftarrow A^\top\left(M_3\boldsymbol{y}\right), \tag{22}
$$

and computes the following J times:

$$
\begin{cases}
\boldsymbol{x} \leftarrow \boldsymbol{M}_4 + \boldsymbol{z} - \boldsymbol{u} - A^\top(M_2(A(\boldsymbol{z}-\boldsymbol{u}))) \\
\boldsymbol{z} \leftarrow \mathit{SoftThr}\left(\boldsymbol{u} + \boldsymbol{x}; \dfrac{1}{\mu}\right) \\
\boldsymbol{u} \leftarrow \boldsymbol{u} + \boldsymbol{x} - \boldsymbol{z}
\end{cases}. \tag{23}
$$

Here, the important idea is that we do not calculate $A^\top M_2 A$ in the Pre-calculation because the computational cost increases when we include this calculation in the Pre-calculation. The result of $\boldsymbol{z} - \boldsymbol{u}$ is a vector; therefore, we can reduce the computational cost by computing $\boldsymbol{x}$ by the order of (23). The computational cost of SMW-ADMM is $\mathcal{O}(R^2N^2 + N^3 + R^2NJ)$.

For example, $R = 32$, $N = 32$, and $J = 100$ were used in previous studies [5, 6]. In this example, the computational cost of SMW-ADMM is roughly estimated as $\mathcal{O}(R^{4.33})$, and that of the original ADMM ℓ_1 minimization is $\mathcal{O}(R^6)$. Consequently, we conclude that SMW-ADMM reduces the computational cost of the ADMM ℓ_1 minimization.

B. Saturation SMW-ADMM

The idea of SMW-ADMM can be applied to the Saturation ADMM ℓ_1 minimization. We named the application of the SMW formula to Saturation ADMM ℓ_1 minimization the Saturation SMW-ADMM.

The Pre-calculation of the Saturation SMW-ADMM is expressed as

$$
M_1 \leftarrow \frac{\nu}{\mu}AA^\top \tag{24}
$$

and

$$
M_2 \leftarrow \frac{\nu}{\mu}\left(I + M_1\right)^{-1}M_1. \tag{25}
$$

The Post-calculation of the Saturation SMW-ADMM computes the following J times:

$$\begin{cases} \boldsymbol{x} \leftarrow A^\top \left(M_2\left(\boldsymbol{z}-\boldsymbol{u}\right)\right) + A\left(\boldsymbol{\zeta}-\boldsymbol{v}\right) \\ \boldsymbol{z} \leftarrow \mathit{SoftThr}\left(\boldsymbol{u}+\boldsymbol{x}; \dfrac{1}{\mu}\right) \\ \boldsymbol{\xi} \leftarrow A\boldsymbol{x}+\boldsymbol{v} \\ \boldsymbol{\zeta} \leftarrow D\left(\boldsymbol{\xi}; \left(\sqrt{\dfrac{1+\nu\lambda}{\nu\lambda}}-1\right)\boldsymbol{y}+\sqrt{\dfrac{1+\nu\lambda}{\nu\lambda}}Y\mathbf{1}, \right. \\ \qquad\qquad \left. \dfrac{1}{1+\lambda\nu}\boldsymbol{y}+\dfrac{\lambda\nu}{1+\lambda\nu}\boldsymbol{\xi}\right) \\ \boldsymbol{v} \leftarrow \boldsymbol{\xi}-\boldsymbol{\zeta} \\ \boldsymbol{u} \leftarrow \boldsymbol{u}+\boldsymbol{x}-\boldsymbol{z} \end{cases} . \quad (26)$$

This computation reduces the computational cost to the same extent as SMW-ADMM.

IV. Multi-row simultaneous estimation

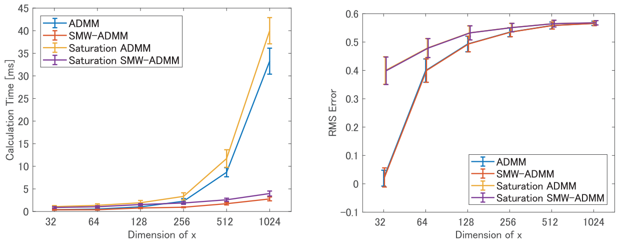
In this section, we describe the proposed approach to improve the efficiency of the computation. We named our approach the "Multi-row simultaneous estimation" for multi-scale LTM estimation, which was proposed in another paper [9]. First, we introduce the multi-scale LTM estimation. Then, we propose our multi-row simultaneous estimation for the multi-scale LTM estimation.

A. Brief introduction of Multi-scale LTM estimation

The main objective of multi-scale LTM estimation is to reduce the number of non-zero elements by using course-to-fine estimation. This estimation first measures a low-resolution LTM, which reduces the resolution of the projector but retains the camera resolution. Once the low-resolution LTM is observed, the zero elements of the low-resolution LTM propagates it to high resolution. Therefore, the method can reduce the size of the estimating vector of the sparse estimation. The method is highly effective such that it reduces 84.28-99.95% elements of the estimating vector [9]. However, the multi-scale LTM estimation needs to perform the Pre-calculation adaptively, which means that the Pre-calculation depends on the observation. Note that, the Pre-calculation for the first resolution can be conducted offline. The problem occurs from the second resolution and increases the computational cost. Therefore, we propose a new solution to reduce the Pre-calculation of the multi-scale LTM estimation.

B. Multi-row simultaneous estimation

Because of the multi-scale LTM estimation, the Pre-calculation from the second resolution should be calculated after the LTM estimation of the previous resolution. This enables us to only reduce the number of computations to reduce the computational cost of this part. Our solution to the problem is simple: to combine some rows of the LTM and estimate them together. When some rows of the LTM are estimated together, the number of elements that need to be included for a sparse estimation increases (see Fig. 2). However, we can reduce the number of computations of the Pre-calculation. Accordingly, there is a trade-off between the number of rows that can be computed simultaneously.



(a) Calculation Time

(b) RMS Error

Fig. 3: The number of iterative calculations and the Root Mean Square (RMS) Error for ADMM, SMW-ADMM, Saturation ADMM, and Saturation SMW-ADMM for ℓ_1 minimization by numerical simulation. Compared with ADMM or Saturation ADMM, the time required to perform the calculations for methods to which the SMW formula was applied was always little. In contrast, the RMS Error does not change even if we apply the SMW formula because this formula does not include any approximation; instead, it is simply a transformation of the equations.

We define the number of rows estimated together as K. We also denote the sparseness propagated by the previous LTM as S, where $S = \#(\text{propagated non-zero elements})/R^2$. We assume the LTM non-zero elements are located uniformly distributed. Here, the computational cost of SMW-ADMM can be estimated as $\mathcal{O}\left(S^2R^2N^2 + N^3 + S^2R^2NJ\right)$ .The expected number of elements that should be computed by the sparse estimation is originally R^2S . When K rows are estimated simultaneously, the probability of an element is always propagated as zero for all K is $(1-S)^K$; thus, S for the multi-row simultaneous estimation becomes $S_{\text{simul}} = 1 - (1-S)^K$. On the other hand, the number of computations of the Pre-calculation decreases by $1/K$ times when K rows are estimated simultaneously. Finally, the computational cost can be estimated as $\mathcal{O}\left(\frac{1}{K}\left(S_{\text{simul}}^2R^2N^2 + N^3\right) + S_{\text{simul}}^2R^2NJ\right)$.

V. Experiments

A. General evaluation of SMW-ADMM

We first evaluate the extent to which the SMW formula accelerates the ADMM ℓ_1 minimization by numerically simulating a toy problem. The observation matrix A is given by randomly sampling the normal distribution. The sparse estimating vector $\boldsymbol{x}$ consists of 99 % zero elements and 1 % non-zero elements. The non-zero elements are randomly sampled by following a uniform distribution from 0.0 to 1.0. The indices of the zero and the non-zero elements are randomly selected. The observed vector $\boldsymbol{y}$ is computed by $\boldsymbol{y} = A\boldsymbol{x}$. The dimension of $\boldsymbol{y}$ is fixed at 32. The methods included in the comparison are ADMM, SMW-ADMM, Saturation ADMM, and Saturation SMW-ADMM for ℓ_1 minimization. They were all implemented using MATLAB R2017b and run on a desktop computer (Intel Core i7-6700K 4.00GHz, 64GB RAM). We recorded the calculation times and the Root Mean Square (RMS) Error when the dimension of $\boldsymbol{x}$ changed from 32 to 1024. Note that the dimension of $\boldsymbol{x}$ is R^2 in the context of the LTM estimation.

We ran the numerical simulation 100 times. The result is shown in Fig. 3. We see that ADMM and SMW-ADMM have the same RMS Error, but the time required to calculate SMW-ADMM is less than for ADMM. We also observe the same trends between Saturation ADMM and Saturation SMW-ADMM. Additionally, for the high dimension of $\boldsymbol{x}$, the difference when using the SMW formula application becomes large. On the other hand, the RMS Error is the same between ADMM and SMW-ADMM, and between Saturation ADMM and Saturation SMW-ADMM. This result is obtained because the SMW formula is simply a transformation of the equations and it does not include any approximation. We can conclude that the SMW formula accelerates ADMM and Saturation ADMM ℓ_1 minimization without any deterioration in the accuracy.

B. LTM estimation problem

We then examined the application of ADMM and Saturation ADMM ℓ_1 minimization, to which the SMW formula was applied, to estimate the real LTM. We utilized a projector-camera system consisting of a projector (EPSON, EB-2155W) and a camera (Point Grey Research, Flea3 FL3-U3-88S2C). The projector-camera system was calibrated according to a previous procedure [14], and these researchers also performed an optical calibration by using a published method [7]. The projection patterns and camera images were transformed by homographic matrices, which result in the projection area and capture area becoming the same on the basis paper (this calibration follows that in [8]). We set both the projector resolution and the camera resolution as $R = 128$, and the multi-scale estimation is utilized as: the first resolution is 32, and the second resolution is 128. Each resolution consists of 32 projections/captures and 1 more for background light subtraction; thus, we utilized 65 images.

To demonstrate that the SMW formula does not affect the computational accuracy, we utilized the estimated LTM for scene relighting. When a projection pattern is projected from the projector, the camera image can be estimated by (1) with the LTM and the background lights. The relighting results are shown in Fig. 4. Because the SMW formula does not affect the accuracy, the relighting results between ADMM and SMW-ADMM, and between Saturation ADMM and Saturation SMW-ADMM are similar observations. Note that, the number of projections/captures is not large; thus, the absolute accuracy of the estimated LTM is not high. This causes the difference between the ground truth camera image and the camera images obtained by relighting. We focus on the difference between the result obtained before applying the SMW formula and that obtained afterwards.

1) Calculation time: We evaluate the extent to which the SMW formula and the multi-row simultaneous estimation accelerate ADMM and Saturation ADMM ℓ_1 minimization. The ADMM, SMW-ADMM, and SMW-ADMM are compared with the multi-row simultaneous estimation, Saturation ADMM, Saturation SMW-ADMM, and Saturation SMW-ADMM with the multi-row simultaneous estimation. We specify the number of rows that are estimated together as $K = 8$ because the following example indicates $K = 8$ to be the fastest in practice.

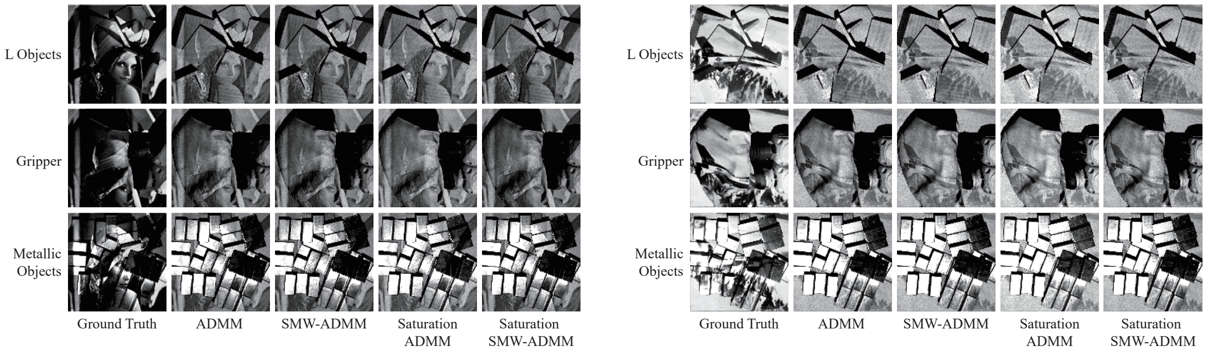
The results are shown in Fig. 5. The SMW formula significantly speed-ups both the ADMM and Saturation ADMM ℓ_1 minimization. The multi-row simultaneous estimation does not affect the first resolution; therefore, it accelerates the calculation time from the estimation of the second resolution onward. Compared with ADMM, SMW-ADMM is 3.08-3.92 times faster, and SMW-ADMM with multi-row simultaneous estimation is 4.06-4.64 times faster, and then compared with Saturation ADMM, Saturation SMW-ADMM is 1.98-2.54 times faster, and SMW-ADMM with multi-row simultaneous estimation is 2.79-3.36 times faster.

2) Evaluation of multi-row simultaneous estimation: Last, we examined the relationship between the calculation time and the number of rows estimated together K. We utilized the same experimental settings as in the above experiments. The controlled parameter is K, and it is set to 1, 2, 4, 8, 16, 32, and 64.

The results are shown in Fig. 6 and Fig. 7. According to the results, to estimate 4-8 rows together is the fastest. For our experimental setting, the sparseness S is 0.0040, 0.0041, and 0.0081 for the scenes with the L objects, the gripper, and the metallic objects, respectively. Here, we can estimate the best value for parameter K at the lowest computational cost, and they are 2.6734, 2.6245, and 1.6315 for the three respective scenes. These values are smaller than the actual result in Fig. 6 and Fig. 7. In our consideration, this is because we only estimates the computational costs; however, the larger the number of rows computed together, the more the actual computational efficiency becomes. Regardless of the scene, the sparseness is larger than 1 in all cases; thus, we can estimate the multi-row simultaneous estimation made by the SMW-ADMM ℓ_1 minimization theoretically.

VI. Conclusion

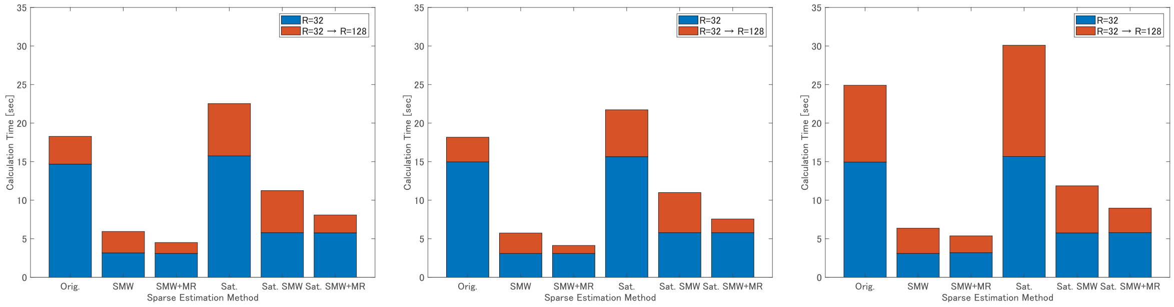
In this study, we focused on reducing the calculation time of the LTM estimation. For this purpose, we proposed two ideas: (1) to apply the SMW formula to ADMM and Saturation ADMM ℓ_1 minimization, and (2) multi-row simultaneous estimation, which is an extension of the multi-scale



(a) Lena

(b) Airplane

Fig. 4: Results of relighting experiment. The ground truth images were captured by camera.The four images on the right-hand side are estimated camera images $\boldsymbol{c}$, which is given by (2), where T is the estimated LTM and $\boldsymbol{l}$ is the images.

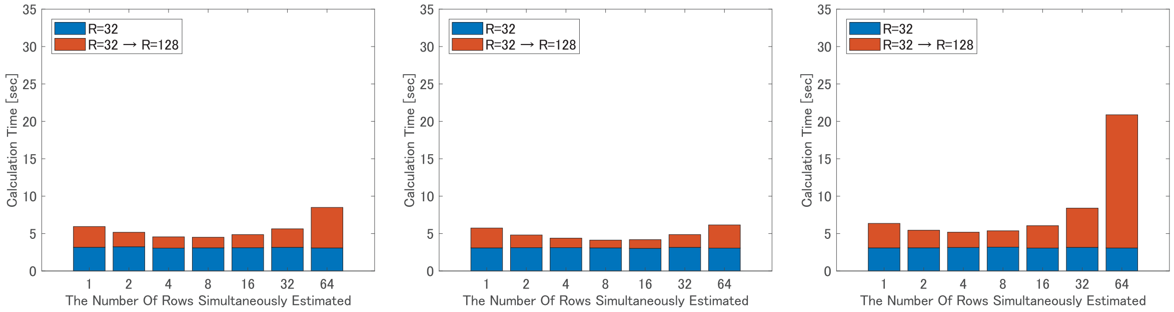


(a) L Objects

(b) Gripper

(c) Metallic Objects

Fig. 5: The time required to calculate the Original ADMM (Orig.), SMW-ADMM (SMW), and SMW-ADMM with the multi-row simultaneous estimation (SMW+MR), Saturation ADMM (Sat.), Saturation SMW-ADMM (Sat. SMW), and Saturation SMW-ADMM with the multi-row simultaneous estimation (Sat. SMW+MR). The multi-row simultaneous estimation binds 8 rows together because it is the fastest for almost all scenes (see Fig. 6 and Fig. 7). The first and second resolutions for the multi-scale LTM estimation are shown in blue and red, respectively.



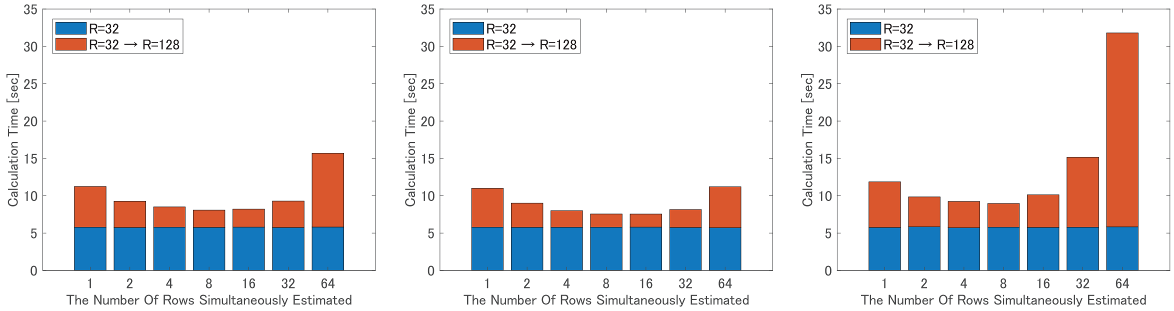
(a) L Objects

(b) Gripper

(c) Metallic Objects

Fig. 6: The times required for the SMW-ADMM calculation. We compare the effects of the number of rows simultaneously estimated. According to the results, to estimate 4-8 rows together is the fastest.

LTM estimation. We also clarified these reductions of the computational cost theoretically in Sec. III and Sec. IV. Additionally, we examined their effects in practice via numerical simulation and by solving a real LTM estimation problem.



(a) L Objects (b) Gripper (c) Metallic Objects

Fig. 7: The times required for the Saturation SMW-ADMM calculation. We also compare the effects of the number of rows simultaneously estimated. These results indicate that to estimate 8 rows together is the fastest for these conditions.

We finally demonstrated the sufficiency of our method to reduce the computational cost of the LTM estimation both in theory and in practice.

References

[1] J. Bai, M. Chandraker, T.-T. Ng, and R. Ramamoorthi. A Dual Theory of Inverse and Forward Light Transport. In *Proceedings of European Conference on Computer Vision*, pages 1–8, 2010.

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundation and Trends in Machine Learning*, 3(1):1–122, 2011.

[3] E. J. Candes and M. B. Wakin. An introduction to compressive sampling. *IEEE Signal Processing Magazine*, 25(2):21–30, 2008.

[4] N. Chiba, S. Arai, and K. Hashimoto. Feedback Projection for 3D Measurements Under Complex Lighting Conditions. In *Proc. American Control Conf.*, pages 4649–4656, 2017.

[5] N. Chiba and K. Hashimoto. 3D Measurement by Estimating Homogeneous Light Transport (HLT) Matrix. In *Proc. IEEE Int. Conf. on Mechatronics and Automation*, pages 1763–1768, 2017.

[6] N. Chiba and K. Hashimoto. Homogeneous light transport matrix estimation based 3D shape measurement. *Int. J. Mechatronics and Automation*, 6(2–3):63–70, 2017.

[7] N. Chiba and K. Hashimoto. Development of Calibration Tool for Projector-Camera System Considering Saturation And Gamma Correction. In *Proc. Robotics and Mechatronics Conf.*, pages 2A2–O09, 2018.

[8] N. Chiba and K. Hashimoto. Sparse Estimation of Light Transport Matrix Under Saturated Condition. In *Proc. British Machine Vision Conf.*, 2018.

[9] N. Chiba and K. Hashimoto. Ultra-Fast Multi-Scale Shape Estimation of Light Transport Matrix for Complex Light Reflection Objects. In *Proc. IEEE Int. Conf. on Robotics and Automation*, pages 1763–1768, 2018.

[10] M. Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. 2010.

[11] G. H. Golub and C. F. van Loan. *Matrix Computations*. JHU Press, fourth edition, 2013.

[12] I. Miyagawa and T. Kinebuchi. Compressive Inverse Light Transport for Radiometric Compensation in Projection-based Displays. *ITE Transactions on Media Technology and Applications*, 5(3):96–109, 2017.

[13] I. Miyagawa, Y. Taniguchi, and T. Kinebuchi. Radiometric Compensation Using Color-Mixing Matrix Reformed from Light Transport Matrix. *ITE Trans. on Media Technology and Applications*, 4(2):155–168, 2016.

[14] D. Moreno and G. Taubin. Simple, Accurate, and Robust Projector-Camera Calibration. In *Proc. Int. Conf. on 3D Imaging, Modeling, Processing, Visualization Transmission*, pages 464–471, 2012.

[15] O. Nasu, S. Hiura, and K. Sato. Analysis of Light Transport based on the Separation of Direct and Indirect Components. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–2, 2007.

[16] D. Needell and R. Vershynin. Uniform Uncertainty Principle and Signal Recovery via Regularized Orthogonal Matching Pursuit. *Foundations of Computational Mathematics*, 9(3):317–334, 2009.

[17] D. Needell and R. Vershynin. Signal recovery from incomplete and inaccurate measurements via regularized orthogonal matching pursuit. *IEEE J. of Selected Topics in Signal Processing*, 4(2):310–316, 2010.

[18] M. O'Toole, J. Mather, and K. N. Kutulakos. 3D Shape and Indirect Appearance by Structured Light Transport. In *Proceedings of Conference on Computer Vision and Pattern Recognition*, pages 3246–3253, 2014.

[19] M. O'Toole, R. Raskar, and K. N. Kutulakos. Primal-dual Coding to Probe Light Transport. *ACM Trans. on Graphics*, 31(4):pp. 39:1–39:11, 2012.

[20] M. O'Toole, R. Raskar, and K. N. Kutulakos. 3D Shape and Indirect Appearance by Structured Light Transport. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 38(7):1298–1312, 2016.

[21] P. Sen and S. Darabi. Compressive Dual Photography. *Computer Graphics Forum*, 28(2):609–618, 2009.

[22] P. Peers, D. K. Mahajan, B. Lamond, A. Ghosh, W. Matusik, R. Ramamoorthi, and P. Debevec. Compressive Light Transport Sensing. *ACM Trans. on Graphics*, 28(1):pp. 3:1–3:18, 2009.

[23] S. M. Seitz, Y. Matsushita, and K. N. Kutulakos. A Theory of Inverse Light Transport. In *Proc. IEEE Int. Conf. on Computer Vision*, pages 1440–1447, 2005.

[24] P. Sen, B. Chen, G. Garg, S. R. Marschner, M. Horowitz, M. Levoy, and H. P. A. Lensch. Dual Photography. *ACM Trans. on Graphics*, 24(3):745–755, 2005.