

Figure 1: Gigapixel 3D camera. We equipped Microsoft Research Asia’s 1.6 gigapixel camera with 16 LEDs to realize nearby light photometric stereo.

variations of light directions.

Light calibration. The light direction/intensity is usually calibrated from a mirror/diffuse sphere by observing the peak intensity of the reflection on the sphere in traditional photometric stereo scenario. However, because our lights are placed near to each other, projections of them onto a mirror sphere are near, and then obtaining accurate light directions becomes difficult; meanwhile, our requirement of accuracy is higher than traditional photometric stereo because the nearer the lights are, the more important efficiently differentiating them is. Readers are directed to [Xu and Wallace 2008] and [Wong et al. 2008] for broad overviews of existing light calibration methods. These methods either need complicated steps, or set a goal that accuracy requirement is not high. In contrast, our method uses only one checkerboard and simple operations but maintains high accuracy. Though the method is developed for our gigapixel camera case, it can be easily generalized for common cameras.

3 Imaging setup

We first describe the imaging setup. Our system consists of Apsara [Microsoft 2010b] and 16 controllable Cree XLamp XM-L LEDs. Fig. 1 shows images of the Apsara equipped with the LED frame.

The lens of Apsara is Schneider Optics Apo Symmar L 480/8.4 whose distortion is small (distortion at 90% image range is 0.1% barrel). “480/8.4” means that focal length of the lens is 480[mm] and the minimum F -number of the aperture is 8.4. Apsara uses a CCD sensor back of Lumenera Corp. The CCD sensor’s resolution is 4008×2672 and the pixel size is $9[\mu\text{m}] \times 9[\mu\text{m}]$. Apsara moves the sensor back in the imaging plane so that it records a *single perspective* gigapixel image by stitching the recorded images. By assuming a thin lens model, the distance u from the lens to the object (object distance) and the distance v from the lens to the image (image distance) relate to the focal length of the lens f_l as

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f_l}, \quad (1)$$

which is known as the thin lens formula (or Gaussian lens formula) [Steinhaus 1999]. If the ratio u/v is known, both u and v can be calculated. From the object distance u and image distance v , we can compute the Depth of Field (DOF) and Field of View (FOV) of

the camera system. The near limit of DOF D_N and the far limit of DOF D_F are computed [Jacobson et al. 2000] by

$$D_N = \frac{u f_l^2}{f_l^2 + N c (u - f_l)} \quad (2)$$

and

$$D_F = \frac{u f_l^2}{f_l^2 - N c (u - f_l)}, \quad (3)$$

where N is the F -number of the aperture, and c is the diameter of Circle of Confusion (CoC) that equals to 0.024[mm] in our system. Actually, DOF of Apsara is very shallow, *e.g.*, when object distance $u = 1600[\text{mm}]$, $(D_N - D_F)@F8.4 = 3[\text{mm}]$, $(D_N - D_F)@F16 = 6[\text{mm}]$, and $(D_N - D_F)@F64 = 24[\text{mm}]$. Apsara has a capability of focus stacking for recording a scene that has a depth variation greater than the DOF, *i.e.*, DOF can be extended by focus stacking. For further details of the Apsara camera, readers are referred to [Ben-Ezra 2010; Microsoft 2010b].

Our system precisely controls LEDs in synchronization with sensor exposures. For each sensor back location, our system records the target scene under different LED lightings (can be multiplexed illumination [Schechner et al. 2003]). The light intensity of LEDs is not isotropic, as with any real-world LEDs, which requires extra care; the light intensity is strongest in one direction and decreases in other directions. In addition, to achieve a compact lighting frame which is smaller than the target scene, it is needed to consider a perspective illumination effect where under illumination of one LED scene points receive lights from different directions.

4 Shape recording

Let us denote a 3D light source position as $\mathbf{s} \in \mathbb{R}^3$ and location of a scene point P as $\mathbf{x} \in \mathbb{R}^3$. The light vector from point P to the light source becomes $\mathbf{l} = \mathbf{s} - \mathbf{x}$. With a Lambertian assumption, observed intensity i at point P can be described as following with including a light fall-off factor $\|\mathbf{l}\|^2$:

$$i = E \frac{\mathbf{l}^\top (\rho \mathbf{n})}{\|\mathbf{l}\|^3}, \quad (4)$$

where E is the light source intensity at a unit distance at this particular direction. From a set of n observations ($n \geq 3$) under varying lightings $\mathbf{l}_1, \dots, \mathbf{l}_n$ emitted from different light sources $\mathbf{s}_1, \dots, \mathbf{s}_n$, the albedo-scaled surface normal at point P can be computed as

$$\rho \mathbf{n} = (\mathbf{L}^\top)^\dagger \mathbf{i}, \quad (5)$$

where \dagger denotes a Moore-Penrose pseudo inverse operator,

$$\mathbf{L} = \begin{bmatrix} E_1 \frac{\mathbf{l}_1}{\|\mathbf{l}_1\|^3} & E_2 \frac{\mathbf{l}_2}{\|\mathbf{l}_2\|^3} & \dots & E_n \frac{\mathbf{l}_n}{\|\mathbf{l}_n\|^3} \end{bmatrix}, \quad (6)$$

$$\text{and } \mathbf{i} = [i_1 \quad i_2 \quad \dots \quad i_n]^\top. \quad (7)$$

With calibrated \mathbf{L} , it gives a unique solution; however, knowing accurate light vectors is difficult because light intensities and the relative positions of each scene point and the lights are required. In the following, we describe a method to calibrate the lightings in an assumed depth range and assess its accuracy.

4.1 Light calibration

The problem of light calibration is non-trivial in our case. As described above, because of the perspective lighting model and non-uniform distribution of lightings, lighting direction and intensity

need to be calibrated with respect to a scene point location. In other words, the light configurations (incident angle and strength) vary with the scene point. It is unrealistic to perform calibration for the whole 3D space in the depth range where the target will be placed. This section describes a practical approach of light calibration in this setting and next section assesses its performance.

Our light calibration uses a diffuse checkerboard pattern. By observing the checkerboard pattern at a certain 3D location under a certain lighting condition, our method associates the observed intensity with the surface normal \mathbf{n}^g that is geometrically computed from the checkerboard pattern. More specifically, we solve for the lighting vector using Eq. (4) with the known surface normal, given m surface normal directions ($m \geq 3$, corresponding to varying poses of the checkerboard pattern). For computing the lighting vectors \mathbf{L} at a scene point P , we use the pseudo inverse of the surface normal matrix $[\mathbf{n}_1^{(g)} \dots \mathbf{n}_m^{(g)}]$ as

$$\rho \mathbf{L}^\top = [\mathbf{i}_1 \dots \mathbf{i}_m] [\mathbf{n}_1^{(g)} \dots \mathbf{n}_m^{(g)}]^\dagger, \quad (8)$$

where $[\mathbf{i}_1 \dots \mathbf{i}_m]$ is $n \times m$ and each column consists of observations of the checkerboard at one pose under n lights, $[\mathbf{n}_1^{(g)} \dots \mathbf{n}_m^{(g)}]$ is $3 \times m$ containing surface normal of the checkerboard at m poses, and \mathbf{L} is $3 \times n$ and each column is lighting information of one light. Note that here we assume that the sensor response is linear. In case the sensor response is non-linear, there exist a number of methods for performing the radiometric calibration [Debevec and Malik 2008]. We cannot separate the diffuse albedo ρ from the light matrix \mathbf{L} ; however, this ambiguity is not an issue because all the light matrices are scaled by the same diffuse albedo ρ of the checkerboard pattern. For creating the observation matrix $[\mathbf{i}_1 \dots \mathbf{i}_m]$, we select pixels corresponding to white grids on the checkerboard pattern.

To obtain the geometric surface normals \mathbf{n}^g , calibration of intrinsic parameters of the camera is needed. We use the Camera Calibration Toolbox for Matlab [Bouguet 2007] because of its simplicity. By observing the known checkerboard pattern from the calibrated camera, we can compute the surface normal of the checkerboard from the rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ of the extrinsics by

$$\mathbf{n}^g = \mathbf{R}[0 \ 0 \ 1]^\top. \quad (9)$$

Since it is not practical to perform the light calibration for all possible scene points in the target area, we use an approximate calibration approach; calibration is performed at sub-sampled locations in the target area as depicted in Fig. 2, and the lighting matrix \mathbf{L} is interpolated for the rest of scene points. The accuracy of this calibration approach relates to the density of sampled calibration locations, interpolation method, the size and location of the target, and intensity of light emissions towards different directions. We perform simulations to study the effect of these factors to the calibration accuracy.

4.2 Error analysis

To assess the performance of the light calibration procedure, we conduct simulations using the setup depicted in Fig. 2. In the simulation, we place the light sources equally distributed on a circle which is perpendicular to the optical axis and whose center is at the optical center of the camera. In Table 1, we summarize the fixed settings and variables that are used in the simulation. To simulate the quantization effect using the quantization bits γ , we let the brightest intensity observation be $2^\gamma - 1$. We also simulate the non-uniform directional distribution of the light emittance using a

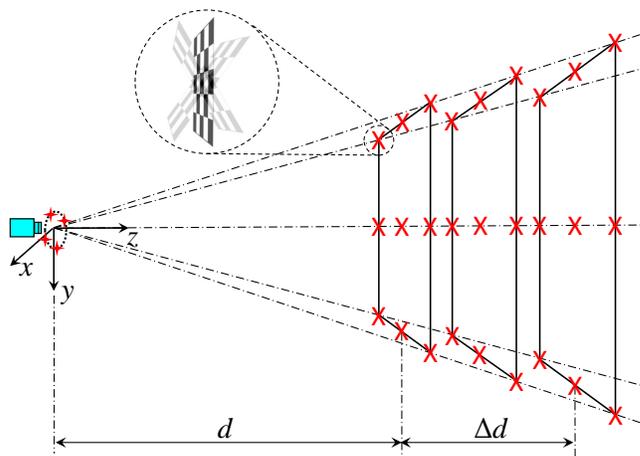


Figure 2: Illustration of light calibration. In photometric stereo, we vary light directions to recover surface normal; in our light calibration, we vary surface normal to obtain light direction. By observing the checkerboard at a certain 3D location under a certain lighting condition, we can recover light intensity and direction by varying the orientation of the checkerboard. Calibration is performed at sub-sampled locations in the target area denoted as \times and is interpolated at the left locations.

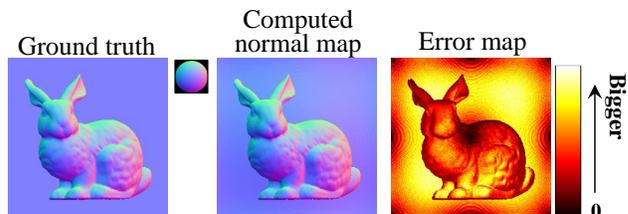


Figure 3: A typical angular error map when sampling mode is 3×3 in the xy -slice.

quadratic curve where the emittance becomes $a\%$ at 30° off from the z -axis (the peak emittance is aligned to the z -axis).

We use as variables the number of sub-sampled calibration locations $\{n_w, n_h, n_d\}$, target depth d and thickness Δd , and interpolation method \mathcal{I} for the light matrix interpolation (each element of light matrix is interpolated independently). By varying these variables, we observe the error bounds of the surface normal sensing.

Fig. 3 shows a typical error map when sampling mode, $n_w \times n_h \times n_d$, is $3 \times 3 \times$ some number. In the simulation, the error is computed as the angular difference in degree between the computed surface normal and the ground truth. Naturally, it has a tendency that the error of pixels near the calibrated locations becomes smaller as the figure depicts.

Fig. 4 shows the effect of interpolation methods when $d = 2000[\text{mm}]$ (Δd is the thickness of the head) and sampling mode is $5 \times 5 \times 3$. By selecting an appropriate interpolation method, we can reduce the error. The nearest neighbor interpolation ‘nearest’ performs the worst, and ‘tricubic’ interpolation improves the result compared with the ‘trilinear’ interpolation. The difference between the ‘tricubic’ and ‘tri-cubic-spline’ is minor in terms of accuracy; however, we found that ‘tri-cubic-spline’ consistently performs slightly better than ‘tricubic’ interpolation with a similar computational cost.

Table 1: List of constants and variables in our simulation.

Constant	Description	Value
N	Number of lights	3
r	Radius of light frame	100mm
a	Relative light emittance at 30° off from z -axis	90%
$FOV@D$	Field of view at depth D	$2000 \times 2000\text{mm}^2@2000\text{mm}$
γ	Quantization bits	8
Variable	Description	Value
d	The nearest distance from target area to camera	(0, 5000mm]
Δd	Thickness of target scene along z -axis	[0, 1000mm]
$n_w \times n_h \times n_d$	Number of sub-sampled locations for calibration along x -, y -, and z -axis	$\{3 \times 3 \times 3, 5 \times 5 \times 3, 9 \times 9 \times 3\}$
\mathcal{I}	Interpolation method	{‘nearest’, ‘trilinear’, ‘tricubic’, ‘tri-cubic-spline’ ¹ }

¹ ‘tri-cubic-spline’ means cubic spline interpolation in three dimensional space.

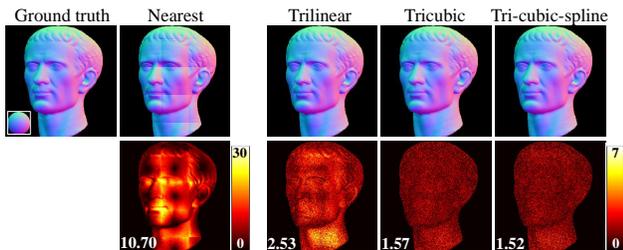


Figure 4: Comparison of interpolation methods. Below each normal map except ground truth are angular error maps. The numbers on the error maps show mean angular errors of normal estimates (unit: degree).

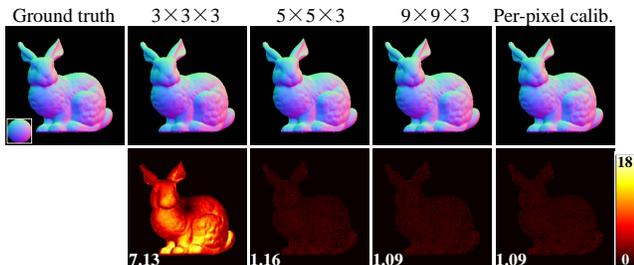


Figure 5: Comparison of different sampling modes, $n_w \times n_h \times n_d = \{3 \times 3 \times 3, 5 \times 5 \times 3, 9 \times 9 \times 3, \text{ and per-pixel calibration}\}$. The numbers on the error maps show mean angular errors of normal estimates (unit: degree).

Fig. 5 shows the comparison of applying different sampling modes when $d = 2000[\text{mm}]$ (Δd is the thickness of the bunny) and \mathcal{I} is ‘tri-cubic-spline’. The rightmost column ‘per-pixel calib.’ indicates the ideal configuration where the calibration is performed at the pixel density; *i.e.*, the error source is only quantization. From the results, we can observe that denser calibration sampling yields a better result and that sparse sampling can achieve reasonably accurate result, *e.g.* $5 \times 5 \times 3$.

Fig. 6 shows the error variations with different d and Δd when sampling mode is set to $3 \times 3 \times 3$ and \mathcal{I} is set to ‘tri-cubic-spline’. The vertical axis is the mean angular error of 300000 points with visible surface normals randomly distributed in the depth range $[d, d + \Delta d]$. Understanding the expected error variations is useful for the choice of the target distance d and target thickness Δd . As we can see from the plots, one tendency is that the error becomes smaller with

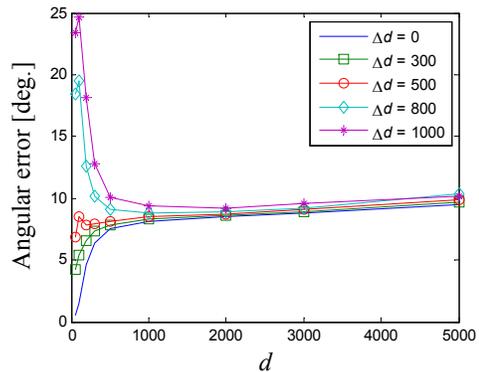


Figure 6: Angular error changes with d and Δd , unit: mm.

a smaller Δd , which is because when Δd is larger, sampling density becomes sparser and the light fall-off matters more; this becomes particularly significant when the target distance d is small. Another general tendency is that when the target distance d is large (*e.g.*, when $d > 1000[\text{mm}]$ in the plots), the error becomes larger with a larger d , which is because the relative baseline of the light frame becomes smaller when d is larger, then the shading variations under different lights become smaller, and then error caused by pixel intensity’s quantization matters more (readers are referred to [Wang et al. 2015] where small baseline photometric stereo is analytically analyzed); on the other hand, when d is small (*e.g.*, when $d < 1000[\text{mm}]$ and $\Delta d \geq 500[\text{mm}]$ in the plots), error becomes larger with smaller d , which is because light fall-off matters strongly.

4.3 Light calibration of gigapixel 3D camera

This subsection describes technical details of the light calibration of our gigapixel 3D camera. Shallow DOF makes our camera calibration a non-trivial problem because locations of checkerboard can only have small variations in depth which degenerates the calibration problem. We instead calibrate image distance and camera center individually and then use Camera Calibration Toolbox for Matlab [Bouquet 2007] to calibrate other intrinsic parameters.

Shallow DOF can be used for obtaining the ratio u/v (*cf.* (1)): set the aperture to $F8.4$ (DOF is 3mm when object distance u is 1600[mm]), put a flat paper on which there is a line in focus (extremely shallow DOF makes sure that the focused line is parallel to the imaging plane), and then because the lens distortion is small,

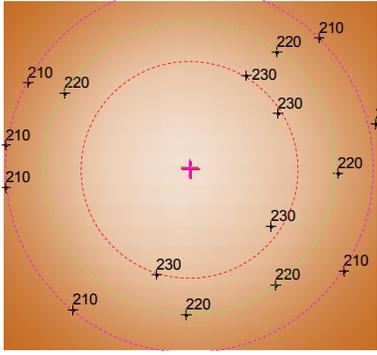


Figure 7: Vignetting when aperture is set to $F8.4$ and the quadratic fitting results in red channel with estimated image center denoted as $+$.

u/v approximately equals to the ratio between the length of the line in the paper and the length of the line in the imaging plane which is the number of pixels $\times 9[\mu\text{m}]$. After knowing u/v , combining Eq. (1), we can solve u and v simultaneously. We obtain the focal length of the camera f_{cam} by setting $f_{cam} = v$ (note that focal length of our lens is $480[\text{mm}]$ and focal length defined in Camera Calibration Toolbox for Matlab is image distance from effective thin lens's center to image plane).

The image center (C_x, C_y) is estimated by locating the center of an optical effect, vignetting [Lin and Low 1990]. We put a uniformly illuminated light box in front of the lens and fit a quadratic model to the vignetting effect as

$$I(x, y) = a_0 + a_1y + a_2x + a_3xy + a_4y^2 + a_5x^2. \quad (10)$$

The image center is then computed as

$$\begin{aligned} C_x &= \frac{a_1a_3 - 2a_2a_5}{4a_5a_4 - a_3^2}, \\ C_y &= \frac{a_1a_3 - 2a_1a_4}{4a_5a_4 - a_3^2}. \end{aligned} \quad (11)$$

Fig. 7 shows an image of recorded vignetting at $F8.4$ and the quadratic fitting results in red channel. Other intrinsic parameters of the camera is calibrated using Camera Calibration Toolbox for Matlab by setting the estimates of f_{cam} and (C_x, C_y) as known parameters. We choose to use $F64$ ($\text{DOF} = 24[\text{mm}]$) when capturing checkerboard at various locations.

Finally, light calibration is performed by capturing a checkerboard at a location but varying poses for each LED lighting condition. We obtain surface normal of the checkerboard by recovering extrinsic parameters (see Eq. (9)). We choose to use $F16$ for a just sufficient DOF ($= 6[\text{mm}]$) with a short exposure time. Fig. 8 shows an image of a checkerboard when the aperture is $F16$. Though the checkerboard is out of focus, the corner extractions still performs reliably.

5 Real-world experiments

Fig. 9 shows a typical imaging setup. Target is placed $1.6[\text{m}]$ (object distance $u = 1.6[\text{m}]$) away from the camera, when Pixels Per Inch (PPI) is $\frac{46092}{930/25.4} = 1259$ pixels/inch. The full-frame is divided into 15×16 tiles (Fig. 10), and the sensor back is moved to capture the target tile by tile. After calibration, we first tried photometric stereo in some tiles (Fig. 10, 11 and 12). Because of high PPI, in Fig. 12, we can see the brush-strokes and the texture

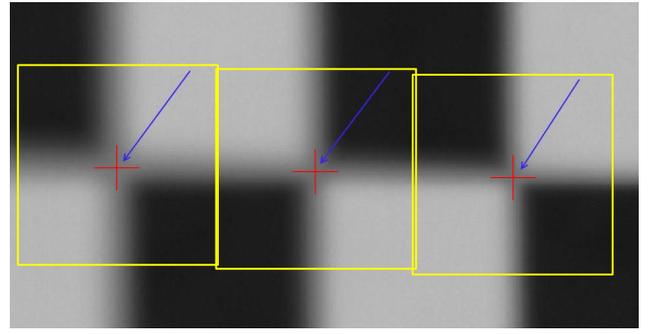


Figure 8: Corner extraction when the measured checkerboard is out of focus under $F16$, $\text{DOF} = 6[\text{mm}]$.

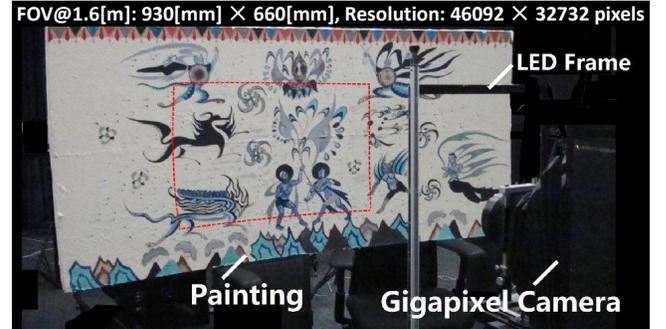


Figure 9: Imaging setup. Gigapixel camera equipped with an LED frame is used to capture a painting $1.6[\text{m}]$ in front.

of the watercolor painting clearly, which cannot be seen from a 2D photo. Then we tried gigapixel 3D imaging, and the albedo map and surface normal map are shown in Fig. 13 and Fig. 14 respectively. Comparing both figures, we see that fine 3D structures are captured which are unobvious in traditional 2D image. Because of ultra-high resolution, even small convex protuberance with diameter $0.59[\text{mm}]$ can be recorded. The time cost for a full-frame 3D imaging is: there are two parallel work threads, a capturing thread and a saving thread; in the capturing thread, the time of one capture is $t + 0.55[\text{s}]$, where t is exposure time; in the saving thread, if only RAW file is saved when it is the quickest, the time is $0.15[\text{s}]$; therefore, the cost time T is dependent on the capturing thread; the time cost in all is

$$T = (t + 0.55) \times (15 \times 16) \times N \times S [\text{s}], \quad (12)$$

where N is number of lights and S is number of focal stacks. After obtaining the surface normal, we can do relighting using virtual lights (see Fig. 15).

6 Discussions and future work

Our prototype shows a capability of recording fine details of the scene geometry at a gigapixel resolution. It is enabled by a photometric stereo approach that gives pixel-wise estimate of scene surface normal. We believe that this setup is useful for applications where precise shape recording is needed, e.g., digital archive of cultural heritages. We can obtain depth map from the focus stacking via edges/textures sharpness analysis, which is depth from focus [Jarvis 1983]. The proposed system is complementary to it and therefore they can be naturally put together for achieving high-quality sensing. Namely, depth from focus gives coarse shape in-



Figure 11: Photometric stereo in a tile: a buddha statue (58[mm]×44[mm]) and its normal map.

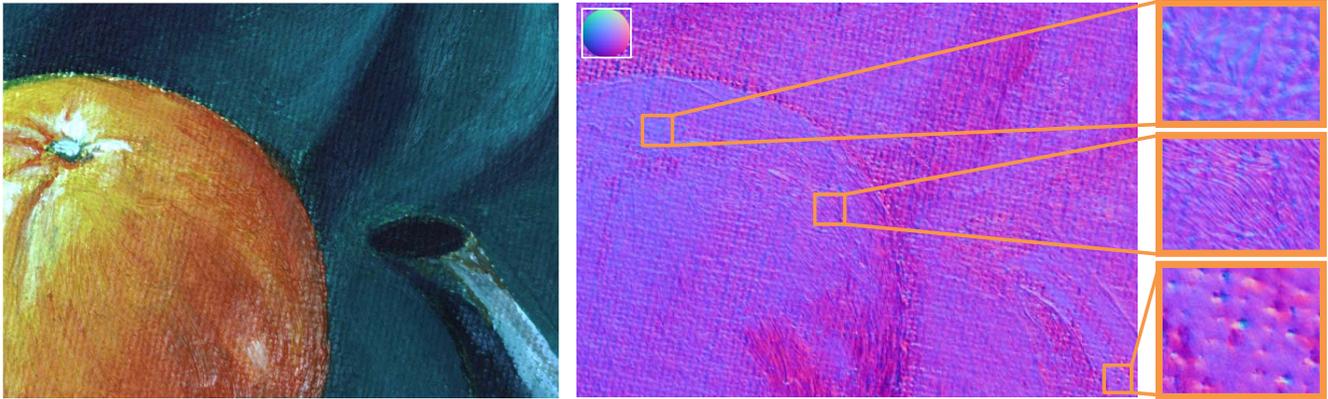


Figure 12: Photometric stereo in a tile: a watercolor painting (55[mm]×40[mm]) and its normal map. Notice that we can see the 3D structure of the brush-strokes and texture clearly.

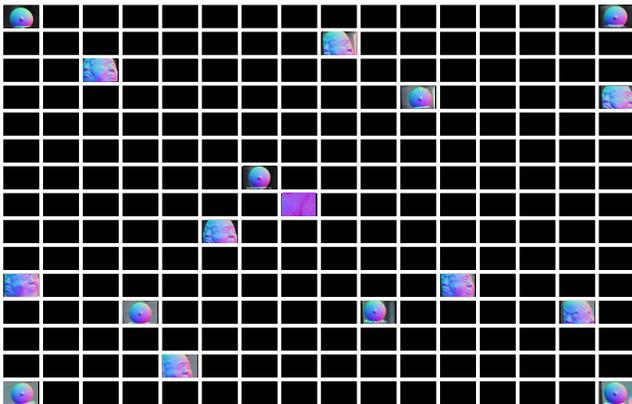


Figure 10: Photometric stereo results in some tiles in full-frame which is composed of 15×16 tiles in total.

formation, and it can be augmented by a *per-pixel* and gigapixel resolution surface normal map.

Acknowledgements

We thank Xin Ma, Gang Chen and Xiao Liang for supporting our project. Our setup is built upon the Apsara gigapixel camera, which is developed by Moshe Ben-Ezra.

References

- BEN-EZRA, M. 2010. High resolution large format tile-scan camera: Design, calibration, and extended depth of field. In *IEEE International Conference on Computational Photography (ICCP)*.
- BOUGUET, J.-Y., 2007. Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/.
- BRADY, D., AND HAGEN, N. 2009. Multiscale lens design. *Optics express* 17, 13, 10659–10674.
- CMU, NASA, AND GOOGLE, 2006. Gigapan epic. <http://gigapan.com/cms/shop/epic>.
- COSSAIRT, O., MIAU, D., AND NAYAR, S. K. 2011. Gigapixel computational imaging. In *IEEE International Conference on Computational Photography (ICCP)*.
- DEBEVEC, P., AND MALIK, J. 2008. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*, 31.
- FLINT, G., 2000. Gigapxl project. <http://www.gigapxl.org/>.
- FURUKAWA, Y., AND PONCE, J. 2010. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 32, 8, 1362–1376.



Figure 13: Gigapixel albedo map of a painting.

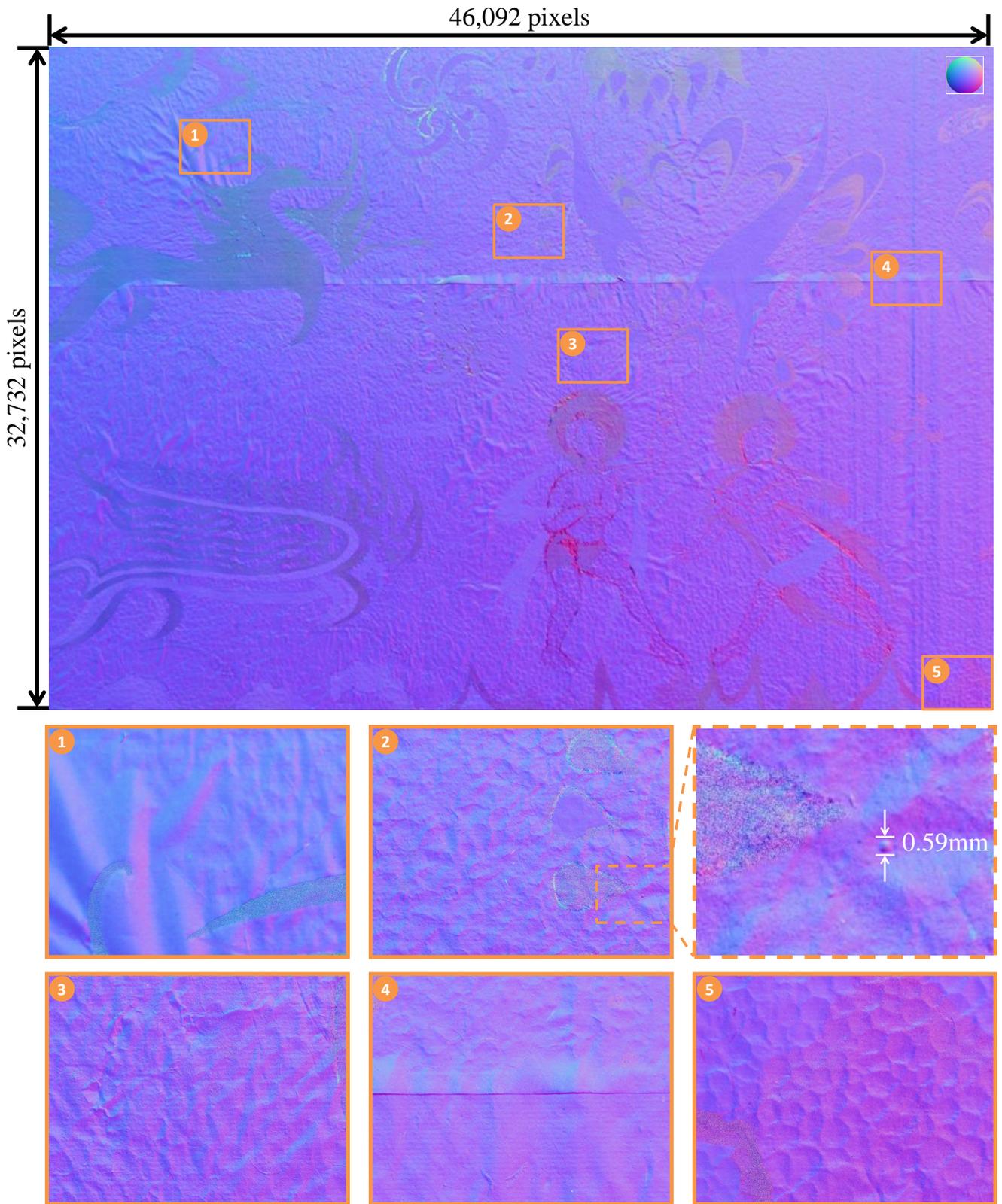


Figure 14: Gigapixel surface normal map of the painting.

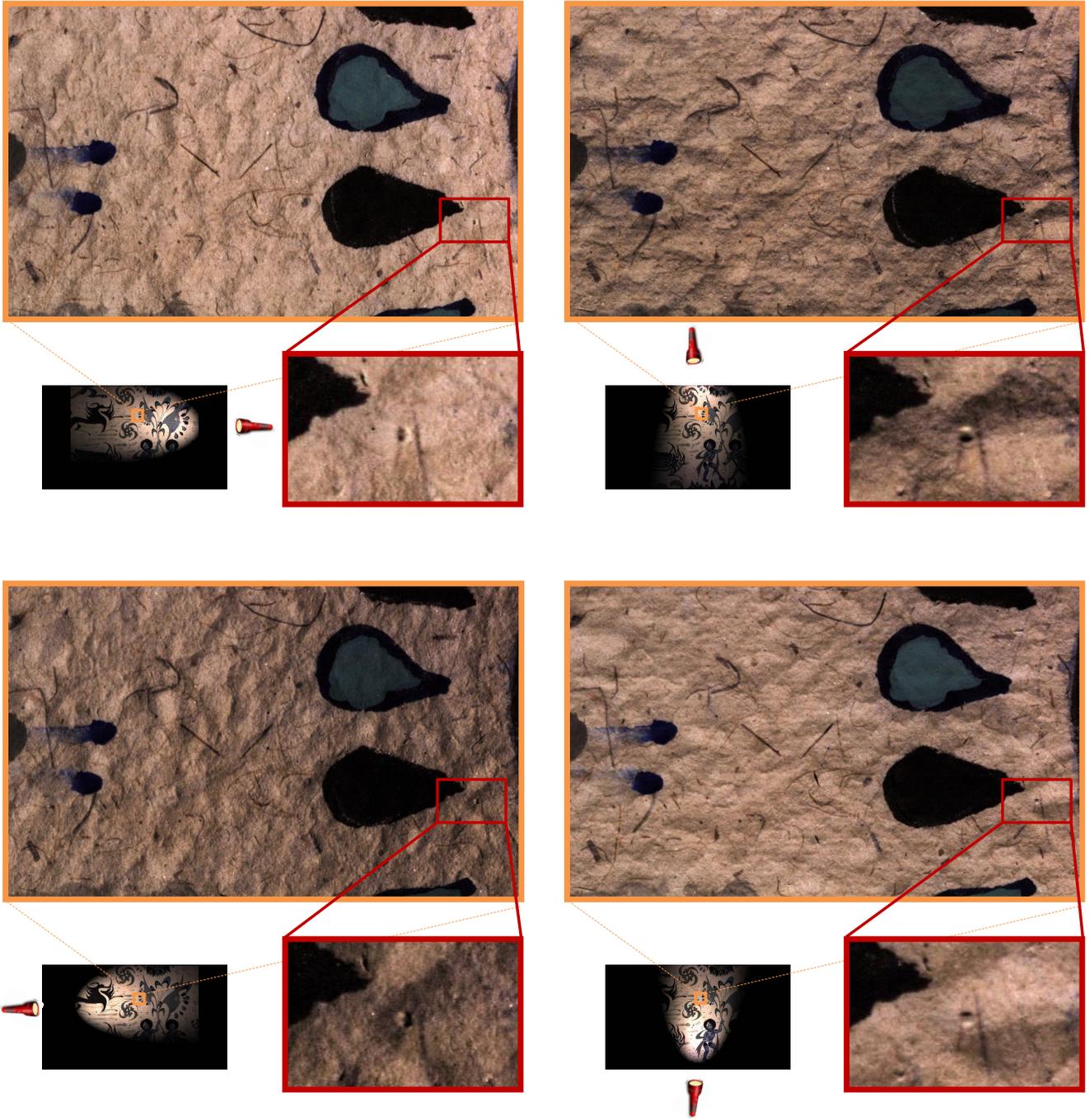


Figure 15: Relighting results. Images are rendered by virtual lighting based on recovered surface normal and albedo information.

- GOOGLE, 2011. Art project. www.googleartproject.com.
- JACOBSON, R., RAY, S., ATTRIDGE, G., AND AXFORD, N. 2000. *The manual of photography: photographic and digital imaging*. Focal Press.
- JARVIS, R. 1983. A perspective on range finding techniques for computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 5, 2, 122–139.
- LIN, Y., AND LOW, T., 1990. Device and method for optical center detection. US Patent 7,307,709.
- LU, Z., LUO, W., SUN, Z., BEN-EZRA, M., AND BROWN, M. S. 2012. Imaging buddhist art with a digital large-format camera: A field study report from the dunhuang caves. *Journal on Computing and Cultural Heritage (JOCCH)* 5, 3 (Oct.), 9:1–9:10.
- LU, Z., TAI, Y., DENG, F., BEN-EZRA, M., AND BROWN, M. 2013. A 3d imaging framework based on high-resolution photometric-stereo and low-resolution depth. *International Journal of Computer Vision (IJCV)* 102, 1, 18–32.
- MICROSOFT, 2010. Kinect. <http://www.xbox.com/en-US/KINECT>.
- MICROSOFT, 2010. Microsoft apsara 1.6 gigapixel camera. <http://www.dgcam.org/dgCam/Home.html>.
- NOMURA, Y., ZHANG, L., AND NAYAR, S. 2007. Scene collages and flexible camera arrays. In *Proceedings of Eurographics Symposium on Rendering*, vol. 2.
- SCHECHNER, Y. Y., NAYAR, S. K., AND BELHUMEUR, P. N. 2003. A theory of multiplexed illumination. In *IEEE International Conference on Computer Vision (ICCV)*.
- SILVER, W. 1980. Determining shape and reflectance using multiple images. *Master's thesis, MIT*.
- STEINHAUS, H. 1999. *Mathematical Snapshots*, 3rd ed. Dover, New York.
- WANG, S., AND HEIDRICH, W. 2004. The design of an inexpensive very high resolution scan camera system. In *Computer Graphics Forum*.
- WANG, J., MATSUSHITA, Y., SHI, B., AND SANKARANARAYANAN, A. C. 2015. Photometric stereo with small angular variations. In *IEEE International Conference on Computer Vision (ICCV)*.
- WANG, J., SANKARANARAYANAN, A. C., GUPTA, M., AND NARASIMHAN, S. G. 2016. Dual structured light 3d using a 1d sensor. In *European Conference on Computer Vision (ECCV)*.
- WHLER, C. 2009. *3D computer vision: efficient methods and applications*. Springer Publishing Company, Incorporated.
- WILBURN, B., JOSHI, N., VAISH, V., TALVALA, E., ANTUNEZ, E., BARTH, A., ADAMS, A., HOROWITZ, M., AND LEVOY, M. 2005. High performance imaging using large camera arrays. *ACM Transactions on Graphics (TOG)* 24, 3, 765–776.
- WONG, K., SCHNIEDERS, D., AND LI, S. 2008. Recovering light directions and camera poses from a single sphere. In *European Conference on Computer Vision (ECCV)*.
- WOODHAM, R. 1980. Photometric method for determining surface orientation from multiple images. *Optical engineering* 19, 1, 191139–191139.
- XU, S., AND WALLACE, A. 2008. Recovering surface reflectance and multiple light locations and intensities from image data. *Pattern Recognition Letters* 29, 11, 1639–1647.
- ZHANG, Q., YE, M., YANG, R., MATSUSHITA, Y., WILBURN, B., AND YU, H. 2012. Edge-preserving photometric stereo via depth fusion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.