RGBD-Fusion: Real-Time High Precision Depth Recovery

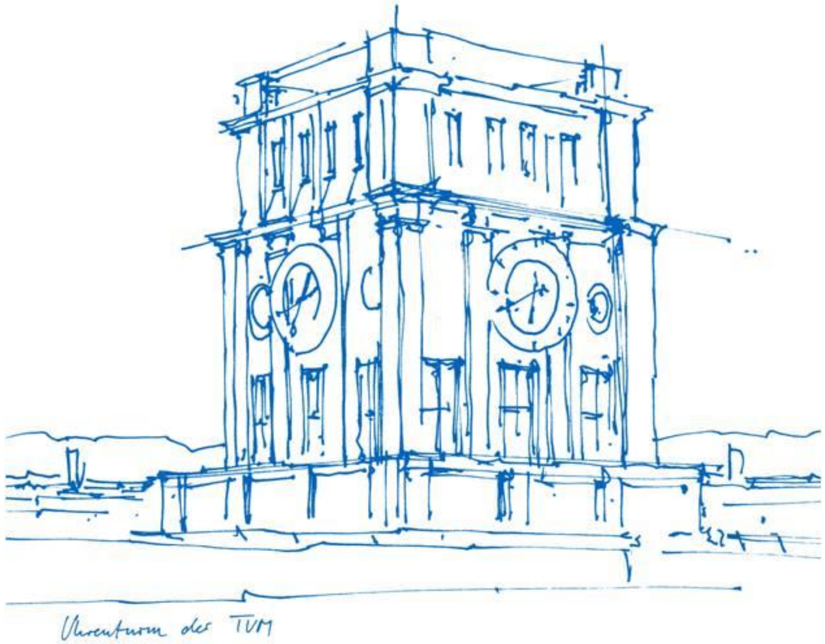
Vinzenz Männig

Technische Universität München

Department of Informatics

Chair of Computer Vision & Artificial Intelligence

Garching, 06. October 2020



Motivation

Example

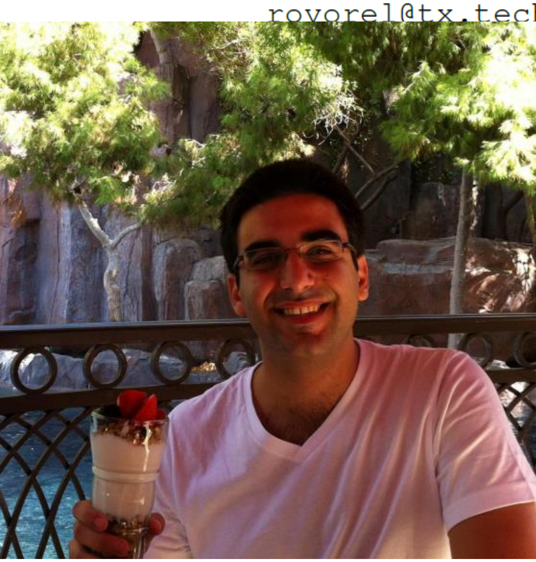
Baseball Cap[1]

Paper

Conference: IEEE Computer Vision and Pattern Recognition (cvpr) 2015

RGBD-Fusion: Real-Time High Precision Depth Recovery

Roy Or - El[1] Guy Rosman[2] Aaron Wetzler[1] Ron Kimmel[1] Alfred M. Bruckstein[1]
[1]Technion, Israel Institute of Technology
[2]Computer Science and Artificial Intelligence Lab, MIT
royorel@tx.technion.ac.il rosman@csail.mit.edu twerd@cs.technion.ac.il
on@cs.technion.ac.il freddy@cs.technion.ac.il

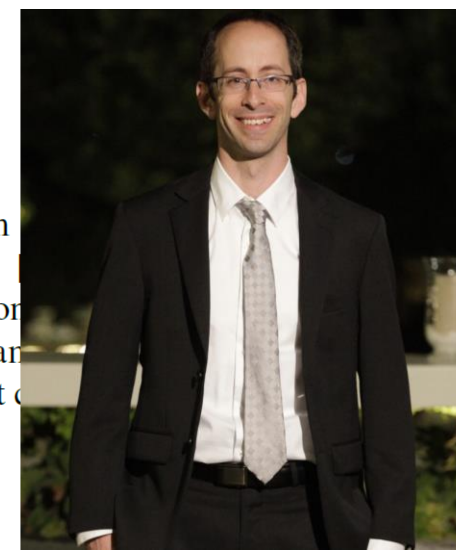


act

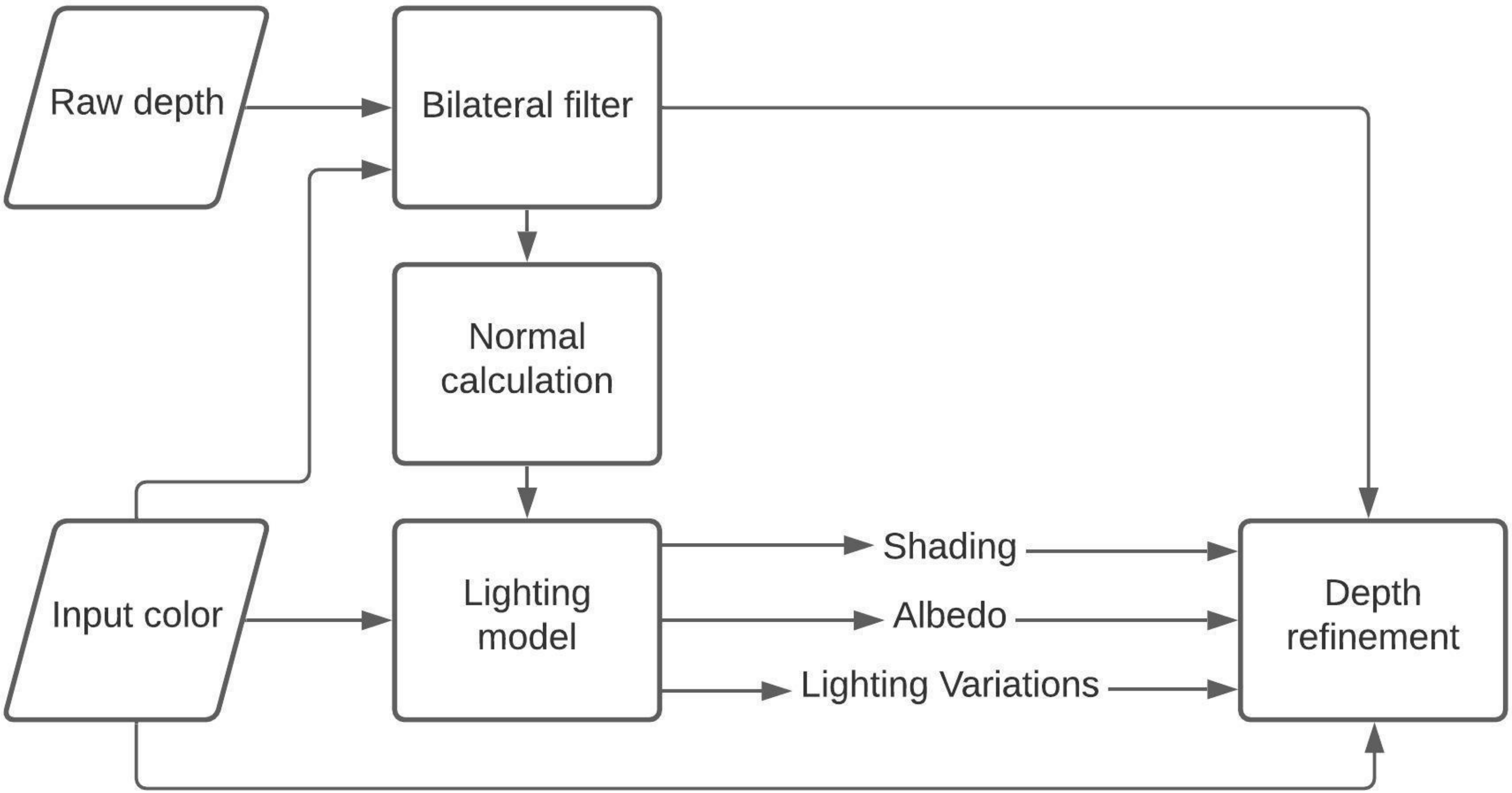
GB-D scanners is increas-
eless, existing scanners of-
ls in the environment. We
nce the depth map by fus-

even richer RGB-D-I fusion.

Reconstructing a shape from
known as shape from shading
searched area in computer vision
problems usually suffer from an
be several possible surfaces that



Outline


Raw depth
Bilateral filter
Normal calculation
Input color
Lighting model
Shading
Albedo
Lighting Variations
Depth refinement

Assumptions

Calibrated fixed system

Intrinsic and extrinsic parameters of depth and color images are known

Not too many different albedos

Bilateral Filter

Depth maps are subject to noise and outliers

Calculate smooth estimate, retain sharp edges for further processing

$$z^{filtered}(i,j) = \frac{1}{W_p} \sum_{k \in \Omega(i,j)} z(i_k, j_k) * f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

Bilateral Filter 2

$$z^{filtered}(i,j) = \frac{1}{W_p} \sum_{k \in \Omega(i,j)} z(i_k, j_k) * f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

Weight

$$W_p = \sum_{k \in \Omega(i,j)} f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

Bilateral Filter 3

$$z^{filtered}(i,j) = \frac{1}{W_p} \sum_{k \in \Omega(i,j)} z(i_k, j_k) * f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

Weight

$$W_p = \sum_{k \in \Omega(i,j)} f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

Spatial kernel

$$f_s(\|(i_k, j_k) - (i,j)\|) = \exp(-\frac{(i_k - i)^2 + (j_k - j)^2}{2\sigma_d^2})$$

Bilateral Filter 4

$$z^{filtered}(i,j) = \frac{1}{W_p} \sum_{k \in \Omega(i,j)} z(i_k, j_k) * f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

Weight

$$W_p = \sum_{k \in \Omega(i,j)} f_s(\|(i_k, j_k) - (i,j)\|) * g_r(\|I(i_k, j_k) - I(i,j)\|)$$

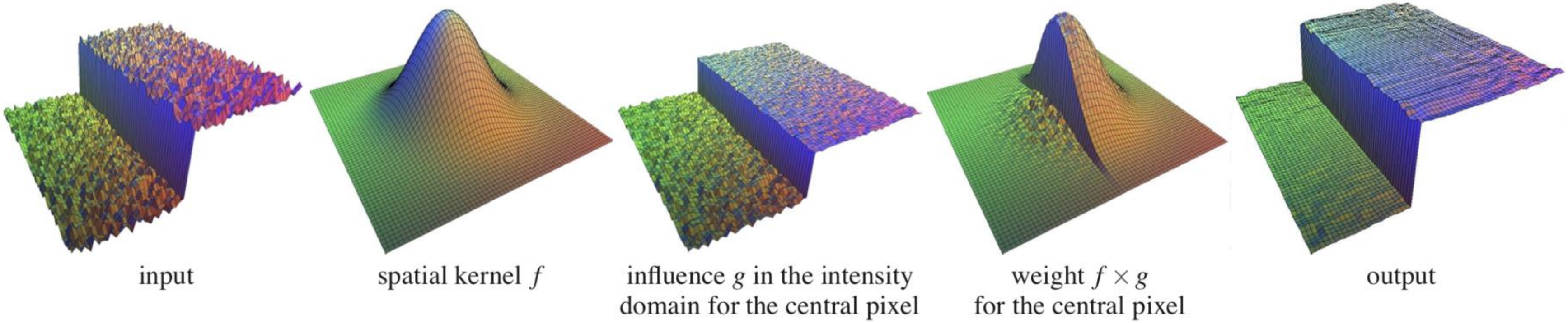
Spatial kernel

$$f_s(\|(i_k, j_k) - (i,j)\|) = \exp(-\frac{(i_k - i)^2 + (j_k - j)^2}{2\sigma_d^2})$$

Intensity kernel

$$g_r(\|I(i_k, j_k) - I(i,j)\|) = \exp(-\frac{\|I(i_k, j_k) - I(i,j)\|}{2\sigma_r^2})$$

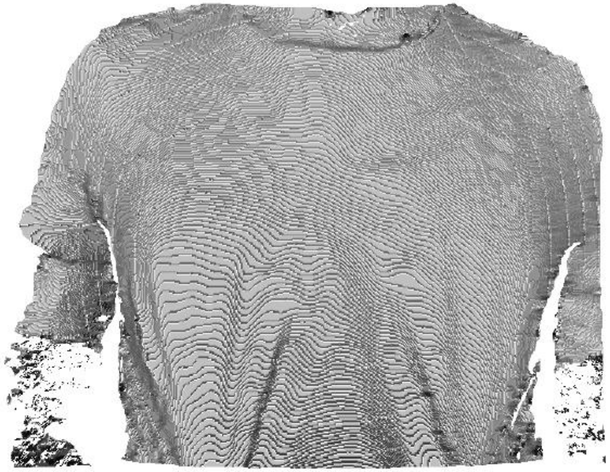
Bilateral Filter 5



Bilateral Filter (https://cs.brown.edu/courses/cs129/labs/lab_bilateral/img/bilateral_filtering.png)

Bilateral Filter 6

Shirt[1]



Surface Normals from Depth Map

Find best fitting plane in a least squares sence

$$\min_{\vec{n}} \sum_{k \in \Omega(\vec{x})} \left(\vec{p}_k^T \vec{n} - d\right)^2$$

$$subject\ to\ \|\vec{n}\| = 1\ and\ n_x x + n_y y + n_z z - d = 0$$

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix}^T \vec{n} = d$$

Surface Normals from Depth Map 2

$$\min_{\vec{n}} \sum_{k \in \Omega(\vec{x})} \left[\vec{p}_k^T \vec{n} - \begin{pmatrix} x \\ y \\ z \end{pmatrix}^T \vec{n} \right]^2$$

$$subject\ to\ \|\vec{n}\| = 1$$

Analytic solution

$$M = \frac{1}{|k|} \sum_{k \in \Omega(\vec{x})} (\overrightarrow{p_k} - \bar{p})(\overrightarrow{p_k} - \bar{p})^T$$

$$\bar{p} = \frac{1}{|k|} \sum_{k \in \Omega} \overrightarrow{p_k}$$

$\vec{n}$ is given by the smallest eigenvector of M

Surface Normals from Depth Map 3

$$\min_{\vec{n}} \sum_{k \in \Omega(\vec{x})} \left[\vec{p}_k^T \vec{n} - \begin{pmatrix} x \\ y \\ z \end{pmatrix}^T \vec{n} \right]^2$$

$$subject\ to\ \|\vec{n}\| = 1$$

$$\min_{\vec{n}} \sum_{k \in \Omega(\vec{x})} \left[\begin{pmatrix} i_k \\ j_k \\ z_k \end{pmatrix}^T \vec{n} - \begin{pmatrix} i \\ j \\ z \end{pmatrix}^T \vec{n} \right]^2 =$$

$$\min_{\vec{n}} \sum_{k \in \Omega(\vec{x})} \left[\begin{pmatrix} i_k - i \\ j_k - j \\ z_k - z \end{pmatrix}^T \vec{n} \right]^2 = \min_{\vec{n}} \sum_{k \in \Omega(\vec{x})} \left[\begin{pmatrix} dx \\ dy \\ dz \end{pmatrix}^T \vec{n} \right]^2$$

Surface Normals from Depth Map 4

$$z = z(x, y)$$

$$\vec{r}(x, y) = \big(x, y, z(x, y)\big)$$

$$\vec{n} = \frac{\partial \vec{r}}{\partial x} \times \frac{\partial \vec{r}}{\partial y} = \begin{pmatrix} 1 \\ 0 \\ \frac{\partial z}{\partial x} \end{pmatrix} \times \begin{pmatrix} 0 \\ 1 \\ \frac{\partial z}{\partial y} \end{pmatrix} = \begin{pmatrix} -\frac{\partial z}{\partial x} \\ -\frac{\partial z}{\partial y} \\ 1 \end{pmatrix}$$

Normalize

$$\vec{n} = \frac{1}{\sqrt{1 + \|\nabla z\|^2}} \begin{pmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \\ -1 \end{pmatrix}$$

Surface Normals from Depth Map 5

$$\vec{n} = \frac{1}{\sqrt{1 + \|\nabla z\|^2}} \begin{pmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \\ -1 \end{pmatrix}$$

cheap calculation e.g. by central differences

$$\frac{\partial z}{\partial x} \approx \frac{z(i+1,j) - z(i-1,j)}{2} \approx \frac{z(i-2,j) - 8z(i-1,j) + 8z(i+1,j) - z(i+2,j)}{12}$$

$$\frac{\partial z}{\partial y} \approx \frac{z(i,j+1) - z(i,j-1)}{2} \approx \frac{z(i,j-2) - 8z(i,j-1) + 8z(i,j+1) - z(i,j+2)}{12}$$

Lighting Estimation

Extended intrinsic image decomposition model (non-Lambertian)[2]

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$

L image lighting, ρ albedos, S shading, β local lighting variations (interreflections, specularities)

Ill-posed problem, use depth map to deal with ambiguities

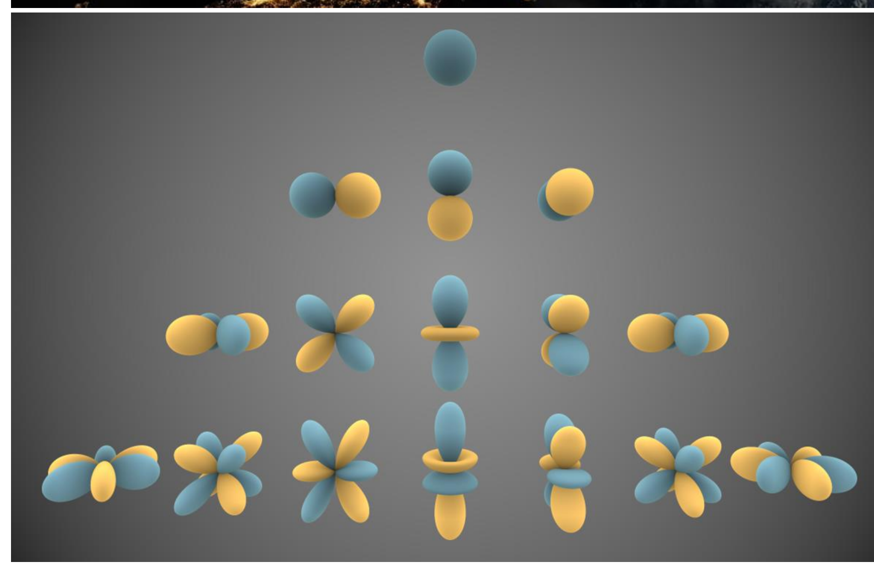
Spherical harmonics

The intensity of isotropic light can be modeled as a function of direction

Function in shere

Represent by spherical harmonics

In cartesian coordinates spherical harmonics can be defined as polynomials



Spherical Harmonics (https://en.wikipedia.org/wiki/File:Spherical_Harmonics.png)

Shading

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$

Assume Lambertian scene and uniform light source to recover shading

$$\rho(i,j) = 1$$
$$\beta(i,j) = 0$$

Irradiance of diffuse objects in natural illumination can be well described by low order sperical harmonic components

For simplicity only use zero and first order harmonics, linear polynomials of surface normals, independent on pixel locations

$$S(\vec{n}) = \vec{m}^T \tilde{n}$$
$$\tilde{n} = (\tilde{n}, 1)^T$$

Shading 2

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$

$$\rho(i,j) = 1, \qquad \beta(i,j) = 0, \qquad S(\vec{n}) = \vec{m}^T\tilde{n}, \qquad \tilde{n} = (\vec{n},1)^T$$

Overdetermined least squares parameter estimation problem

$$\underset{\vec{m}}{\operatorname{argmin}} \|\vec{m}^T\tilde{n} - I\|_2^2$$

Least squares model is not sensitive to high frequency changes and subtle shape changes

Background can also be robustly handled

Explains mostly distant ambient light sources

Albedo Recovery

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$
$$\beta(i,j) = 0$$

Now include different albedos

$$\min_{\rho} \|\rho S(\vec{n}) - I\|_2^2$$

Prone to overfitting $\rho = \frac{I}{S}$

Regularization term to prevent ρ from changing to rapidly required

Assume albedo map piecewise smooth, low number of albedos in picture

Albedo Recovery 2

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$
$$\beta(i,j) = 0$$

Assume albedo map piecewise smooth, low number of albedos in picture

Use weighted Laplacian instead of Gaussian

$$\min_{\rho} \|\rho S(\vec{n}) - I\|_2^2 + \lambda_\rho \left\| \sum_{k \in \Omega} w_k^I w_k^d (\rho - \rho_k) \right\|_2^2$$

Albedo Recovery 3

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$

$$\beta(i,j) = 0$$

$$\min_{\rho} \|\rho S(\vec{n}) - I\|_2^2 + \lambda_\rho \left\| \sum_{k\in\Omega} w_k^I w_k^d (\rho - \rho_k) \right\|_2^2$$

$$w_k^I = \begin{cases} 0, & \|I_k - I(i,j)\|_2^2 > \tau \\ \exp\left(-\frac{\|I_k - I(i,j)\|}{2\sigma_I^2}\right), & otherwise \end{cases}$$

$$w_k^d = \exp\left(-\frac{\|z_k - z(i,j)\|}{2\sigma_d^2}\right)$$

Lighting Variations Recovery

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$

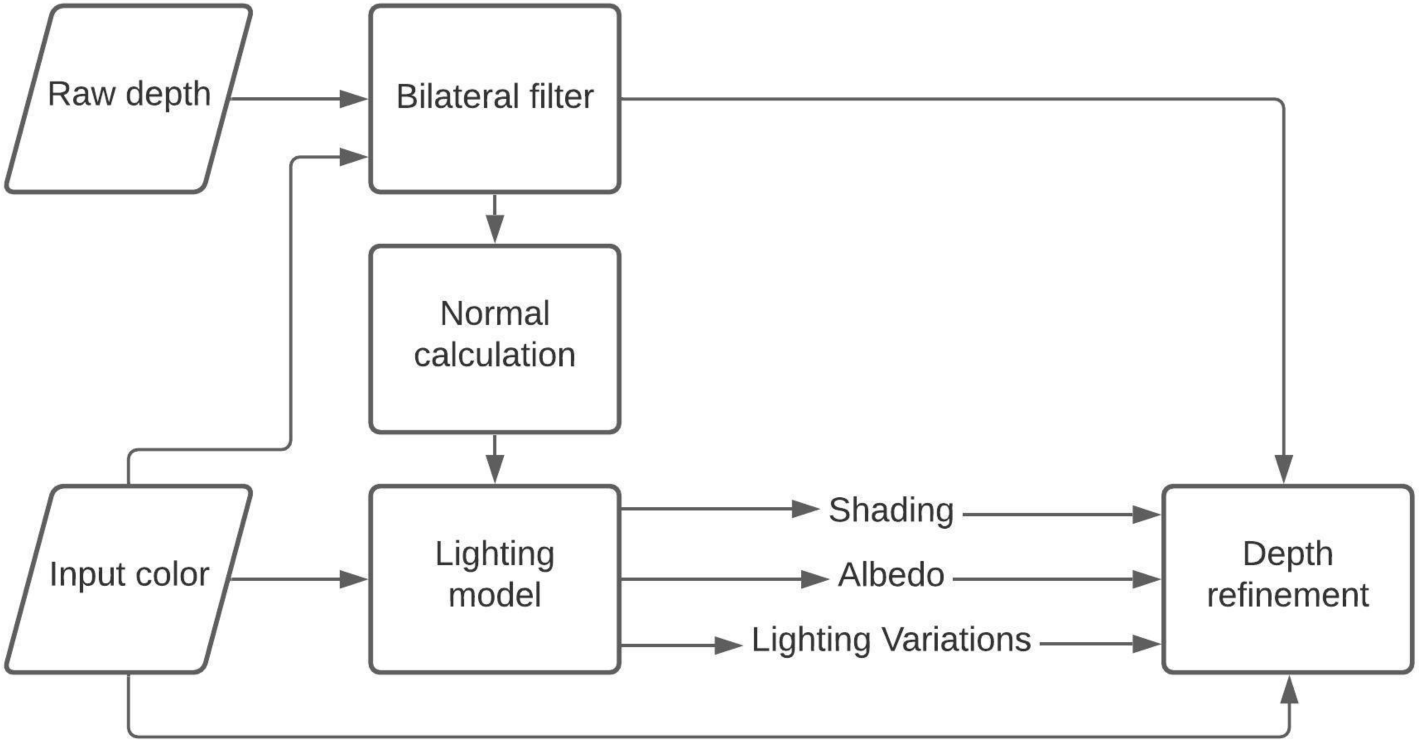
Similar approach for specularities and local light sources, maintain smooth variations

First order harmonics only account for 87,5% of scene lighting, regularization required in order to be consistent with shading model

$$\min_{\beta} \|\beta - (I - \rho S(\vec{n}))\|_2^2 + \lambda_\beta \left\| \sum_{k \in \Omega} w_k^I w_k^d (\beta - \beta_k) \right\|_2^2 + \lambda_\beta^2 \|\beta\|_2^2$$

Orientation

$$L(i,j,\vec{n}) = \rho(i,j)S(\vec{n}) + \beta(i,j)$$



Surface Refinement

Typical Shape from Shading (SFS) method

$$\min_{\vec{n}} \|L(i, j, \vec{n}) - I\|_2^2 + regularization/constraints$$

Surfaces tend to tilt away from viewing direction, aggravated by errors in lighting model (e.g caused by normal outliers from background)

Surface Refinement 2

Typical Shape from Shading (SFS) method

$$\min_{\vec{n}} \|L(i,j,\vec{n}) - I\|_2^2 + regularization/constraints$$

Surfaces tend to tilt away from viewing direction, aggravated by errors in lighting model (e.g caused by normal outliers from background)

Rewrite problem as functional of z to take advantage of existing depth map
Force surface to only change in depth direction

$$\vec{n} = \frac{1}{\sqrt{1 + \|\nabla z\|^2}} \begin{pmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \\ -1 \end{pmatrix}$$

Surface Refinement 3

Fix lighting model, allow the surface gradient to vary

$$\min_z \|L(\nabla z) - I\|_2^2$$

Simplifies numerical scheme, reduces ambiguities

Regularize with simple fidelity and smoothness terms

$$\min_z \|L(\nabla z) - I\|_2^2 + \lambda_z \|z - z_0\|_2^2 + \lambda_z^2 \|\Delta z\|_2^2$$

Surface Refinement 4

Fix lighting model, allow the surface gradient to vary

$$\min_z \|L(\nabla z) - I\|_2^2$$

Simplifies numerical scheme, reduces ambiguities

Regularize with simple fidelity and smoothness terms

$$\min_z \|L(\nabla z) - I\|_2^2 + \lambda_z \|z - z_0\|_2^2 + \lambda_z^2 \|\Delta z\|_2^2$$

Use depth based numerical scheme to increase robustness against lighting model errors (Levenberg-Marquadt algorithm or Trust-Region methods)

$$\Delta = -inv\left(J^T J + \lambda diag(J^T J)\right) J^T r$$

Convergence is slow, not suitable for real-time applications

Surface Refinement 5

$$\min_z \|L(\nabla z) - I\|_2^2 + \lambda_z \|z - z_0\|_2^2 + \lambda_z^2 \|\Delta z\|_2^2$$

Use approach similar to iteratively reweighted least squares

$$\vec{n}^k = \omega^k \begin{pmatrix} \frac{\partial z}{\partial x} \\ \frac{\partial z}{\partial y} \\ -1 \end{pmatrix}, \qquad \omega^k = \frac{1}{\sqrt{1 + \|\nabla z^{k-1}\|^2}}$$

$$L(i, j, \nabla z) = \rho(i, j) * (\vec{m}^T \tilde{n}^k) + \beta(i, j)$$

$$\min_z \|\rho(\vec{m}^T \tilde{n}^k) - (I - \beta)\|_2^2 + \lambda_z \|z^k - z_0\|_2^2 + \lambda_z^2 \|\Delta z^k\|_2^2$$

Surface Refinement 6

$$\min_{z} \|\rho(\vec{m}^T \tilde{n}^k) - (I - \beta)\|_2^2 + \lambda_z \|z^k - z_0\|_2^2 + \lambda_z^2 \|\Delta z k\|_2^2$$

Algorithm 1: Accelerated Surface Enhancement

Input: $z_0, \vec{m}, \rho, \beta$ - initial surface, lighting parameters

while $f(z^{k-1}) - f(z^k) > 0$ **do**

  Update $\tilde{n}^k = (\vec{n}^k, 1)^T$

  Update $L(\nabla z^k) = \rho(\vec{m}^T \tilde{n}^k) + \beta$

  Update z^k to be the minimizer of $f(z^k)$

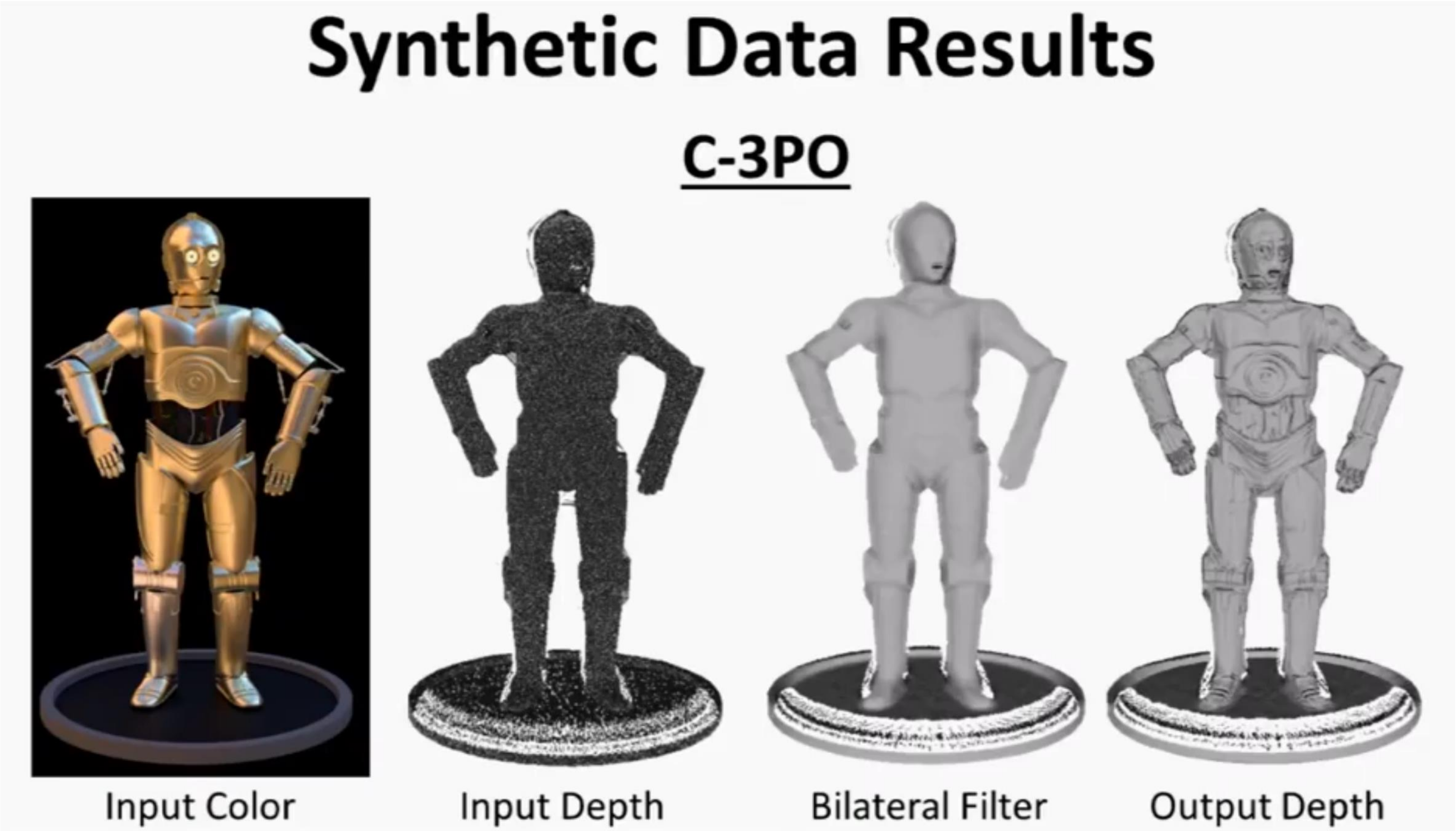
end

Surface Refinement Algorithm[1]

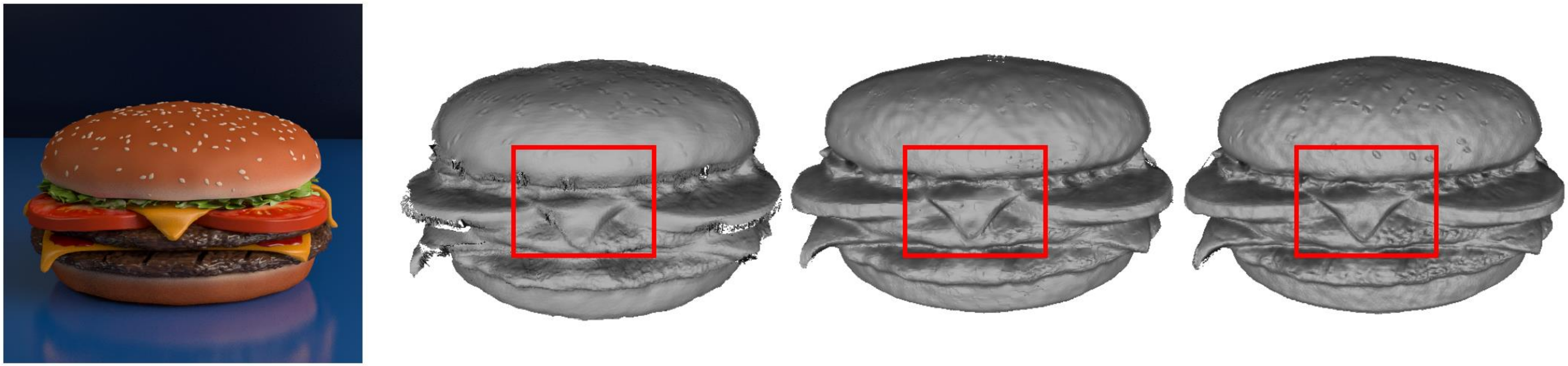
Examples

Shirt[1]

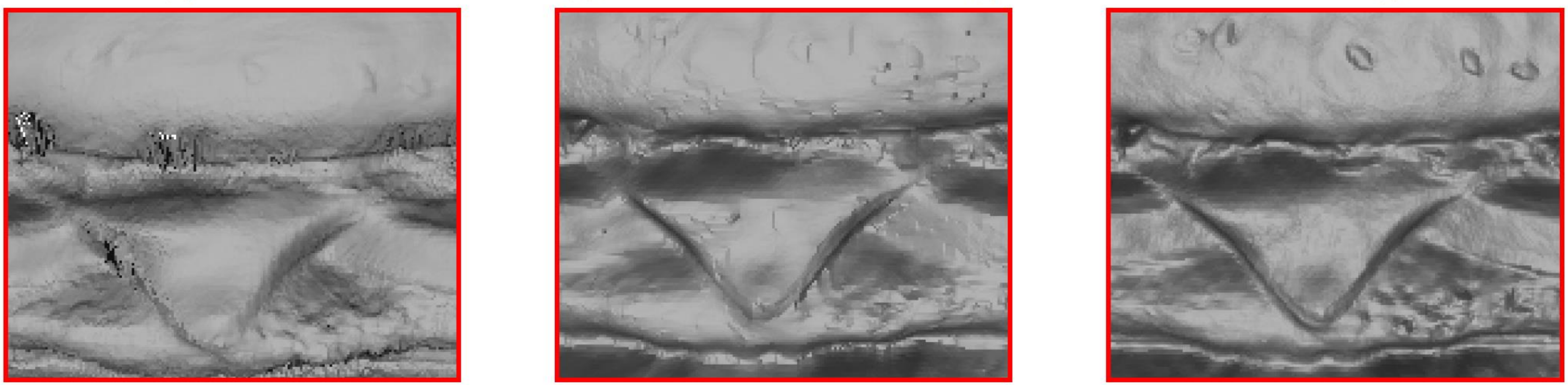
Examples 2


Synthetic Data Results
C-3PO
Input Color
Input Depth
Bilateral Filter
Output Depth

Examples 3



(a) (b) (c) (d)



Burger[1] (e) (f) (g)

Results

Real-Time capable (10fps, 640x480)

In 2015 more accurate than reported state of the art, runs 20000 times faster

I7 3.4GHz
16GB RAM
Nvidia Geforce GTX TITAN GPU

Section	Time
Bilateral Filter	3.8ms
Image alignment	31.1ms
Normal Estimation	5.3ms
Lighting Recovery	40.3ms
Surface Refinement	22.6ms
Total Runtime	103.1ms

Runtime table[1]

References

[1] Or-El, Roy; Rosman, Guy; Wetzler, Aaron; Kimmel, Ron; Bruckstein, Alfred M. RGBD-fusion: Real-time high precision depth recovery. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston, MA, USA. pages 5407–5416, 2015.
[2] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman. Ground-truth dataset and baseline evaluations for intrinsic image algorithms. In International Conference on Computer Vision, pages 2335–2342, 2009.
[3] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(2):218–233, 2003.
[4] E. H. Land and J. J. McCann. Lightness and retinex theory. J. Opt. Soc. Am., 61(1):1–11, Jan 1971.
[5] R. Ramamoorthi and P. Hanrahan. An efficient representation for irradiance environment maps. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pages 497–500. ACM, 2001.