

Depth Restoration via Joint Training of a Global Regression Model and CNNs

Gernot Riegler*¹
 riegler@icg.tugraz.at
 René Ranftl*¹
 ranftl@icg.tugraz.at
 Matthias Rütther¹
 ruether@icg.tugraz.at
 Thomas Pock^{1,2}
 pock@icg.tugraz.at
 Horst Bischof¹
 bischof@icg.tugraz.at

¹ Institute for Computer Graphics and Vision
 Graz University of Technology
 Graz, Austria
² Digital Safety & Security Department
 Austrian Institute of Technology
 Vienna, Austria

1 Overview

The supplemental material of our BMVC 2015 submission provides remaining information to our method and additional quantitative and qualitative results. In Section 2 we visualize the Huber approximation used in our regularization term and compare it to the ℓ_1 and Huber norm, respectively. Section 3 gives the proof to Proposition 1 that we used in the paper. In Section 4 we show how to compute the gradient of our Global Regression Model (GRM) in matrix-vector notation. Finally, we show some additional evaluations of our method in Section 5.

2 Smooth Huber Approximation

In Figure 1 we compare the smooth Huber approximation

$$n_{\varepsilon}(t) = [|t| \leq \varepsilon] \cdot \left(-\frac{1}{8\varepsilon^3}t^4 + \frac{3}{4\varepsilon}t^2 + \frac{3\varepsilon}{8} \right) + [|t| > \varepsilon] \cdot |t|, \quad (1)$$

that we use in the regularization term of the GRM with the standard Huber norm

$$n(t) = [|t| \leq \varepsilon] \cdot \left(\frac{t^2}{2\varepsilon} + \frac{\varepsilon}{2} \right) + [|t| > \varepsilon] \cdot |t|, \quad (2)$$

and the ℓ_1 norm

$$\ell_1(t) = |t|. \quad (3)$$

Note that the Huber approximation is faithful to the Huber function.

*Authors contributed equally

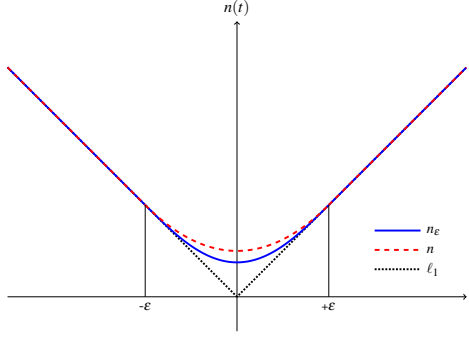


Figure 1: Proposed smooth potential function compared to Huber function and the ℓ_1 norm.

3 Proof to Proposition 1

Proposition 1. *Let $E(u; f(w, I_k))$ be strongly convex and twice differentiable with respect to u . Further, let $E(u; f(w, I_k))$ be differentiable with respect to f and let $f(w, I_k)$ be differentiable with respect to w . Then the gradient of a differentiable loss L with respect to the parameters w is well-defined and is given by*

$$\frac{\partial L}{\partial w} = - \sum_{k=1}^K \left(\left[(\nabla_u^2 E)^{-1} \frac{\partial L}{\partial u_k} \right]^T \frac{\partial^2 E}{\partial u \partial w} \right) \Big|_{u_k = u_k^*}. \quad (4)$$

Proof. Problem HL-LL can equivalently be rewritten in terms of the optimality conditions of the lower-level problem:

$$\min_{w \in W, u_k \in \mathbb{R}^N} \frac{1}{K} \sum_{k=1}^K L(u_k, v_k) \quad \text{s.t.} \quad \nabla_u E(u_k; f(w, I_k)) = 0. \quad (5)$$

Note that we will omit the explicit dependence of $E(u_k; f(w, I_k))$ on the parametrization $f(w, I_k)$ for the rest of the proof in order to facilitate an uncluttered notation.

Problem (5) is an optimization problem with non-linear equality constraints. The Lagrangian of this function is given by

$$\mathcal{L}(u, w, \gamma) = \sum_{k=1}^K \frac{1}{K} L(u_k, v_k) + (\nabla_u E(u_k)) \gamma_k. \quad (6)$$

The stationary points of the Lagrangian are characterized by the optimality conditions:

$$\underbrace{\frac{\partial \mathcal{L}}{\partial u_k} = \frac{\partial L}{\partial u_k} + (\nabla_u^2 E(u_k)) \gamma_k = 0}_{C_1}, \quad \underbrace{\frac{\partial \mathcal{L}}{\partial w} = \sum_{k=1}^K \gamma_k^T \frac{\partial^2 E(u_k)}{\partial u_k \partial w} = 0}_{C_2}, \quad \underbrace{\frac{\partial \mathcal{L}}{\partial \gamma_k} = \nabla_u E(u_k) = 0}_{C_3} \quad (7)$$

By substituting the minimizer of the lower-level problem $u_k^*(f(w, I_k))$ for u_k , condition C_3 can be eliminated, since it is full-filled by definition. From strong convexity of the energy $E(u)$, it follows that $\nabla_u^2 E(u) \succ 0$. Thus the Lagrange multipliers γ_k can be explicitly computed from C_1 , which results in

$$\gamma_k = -(\nabla_u^2 E)^{-1} \frac{\partial L}{\partial u_k}. \quad (8)$$

Substituting (8) into C_2 finally yields the gradient (4). \square

4 Gradient Computations

In order to allow for convenient gradient computations, we reformulate the regularizer using matrix-vector notation

$$R(u) = \sum_{i=1}^N \sum_{j>i}^N n_{\varepsilon}(h_{ij}(w_h, I_k)(u_i - u_j)) = \rho(WBu), \quad (9)$$

where $B \in \mathbb{R}^{M \times N}$ is a sparse matrix, which for each of the M edges (i, j) has a row with an entry -1 at position i and 1 at position j . The matrix $W = W(h_{ij}(w_h, I_k)) \in \mathbb{R}^{M \times M}$ is a diagonal matrix that facilitates the weighting of each edge, i.e.. we have

$$(WBu)_{ij} = h_{ij}(w_h, I_k)(u_i - u_j). \quad (10)$$

We further define

$$\rho(x) = \sum_{m=1}^M n_{\varepsilon}(x_m), \quad \text{for } x \in \mathbb{R}^M. \quad (11)$$

Using this notation the Hessian of energy (2) for a single example u_k can be written as

$$\nabla_u^2 E(u) = B^T W D'' W B + \exp(w_{\lambda}) I, \quad \text{where } D'' = \text{diag}(n''_{\varepsilon}((WBu)_1), \dots, n''_{\varepsilon}((WBu)_M)) \quad (12)$$

Finally, the energy gradients with respect to the parameterizations are given by

$$\begin{aligned} \frac{\partial^2 E(u)}{\partial u \partial w_{\lambda}} &= \exp(w_{\lambda})(u - g(w_g, I_k)) \\ \frac{\partial^2 E(u)}{\partial u \partial g} &= -\exp(w_{\lambda}) \\ \frac{\partial^2 E(u)}{\partial u \partial h} &= D' B + \text{diag}(Bu) D'' W B, \end{aligned} \quad (13)$$

where $D' = \text{diag}(n'_{\varepsilon}((WBu)_1), \dots, n'_{\varepsilon}((WBu)_M))$.

5 Additional Evaluations

NYU2 We evaluated our method additionally on the NYU-Depth V2 dataset [11]. The dataset contains 407,024 frames from a variety of indoor scenes captured with the Microsoft Kinect v1. We use the subset of 1,449 frames that have aligned RGB images as ground-truth. The set is split into 1,000 image pairs for training and the remainder was used for testing. In order to simulate the acquisition process of a depth sensor, we add multiplicative Gaussian noise with $\sigma \in \{0.2, 0.5, 0.7\}$ to the depth maps.

The results on this dataset are depicted in Table 1. We compare the same methods as in the denoising experiment in the paper. We can observe that our approach again performs

	Ours (NL)	Ours (L)	CNN	K-SVD	SAR-BM3D	BM3D	TGV-L2	TV-L1
$\sigma = 0.2$	2.852	2.903	4.152	5.132	3.976	3.262	3.336	4.133
$\sigma = 0.5$	4.616	4.744	7.181	7.417	10.027	5.601	5.682	7.024
$\sigma = 0.7$	5.481	5.546	9.213	9.117	13.209	6.983	6.729	8.671

Table 1: NYU2 denoising results: Quantitative comparison of our method with local regularization (L) and non-local regularization (NL), to the plain CNN and several other state-of-the-art methods over three noise levels on the NYU2 dataset. The error is measured as *RMSE* in *cm*.

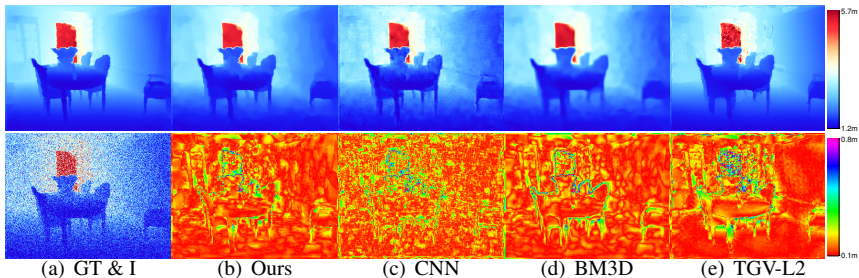


Figure 2: **Qualitative results on NYU2:** The first column presents the ground-truth and the noisy depth map input. The remaining columns depict the denoising results in the first row and the absolute error in the second row, respectively. The input depth maps have a multiplicative Gaussian noise with $\sigma = 0.5$. The results as RMSE in *cm* for our method with non-local regularization, the CNN output, BM3D and TGV-L2 are 6.106, 9.837, 15.758, 7.634, respectively.

best on this dataset and the non-local regularization improves the result when compared to the local regularization.

An exemplar qualitative result on this dataset is presented in Figure 2. We can observe that TGV-L2 produces smooth surfaces, but has difficulties in the background and near depth discontinuities. BM3D struggles with the multiplicative noise and also with depth discontinuities. Our method does not show these problems. However, the slanted surfaces are not as smooth as with a higher-order regularization term.

References

- [1] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *ECCV*, 2012.