Focal Track: Supporting Material

Qi Guo, Emma Alexander, and Todd Zickler

Harvard School of Engineering and Applied Science, TR-01-17

This manuscript contains experimental details and hardware information for the focal track sensor of Guo et al. [1].

Contents

1 Experimental details

1.1 Numerical stability

We add small constants to the denominators of some of the analytical expressions involving $\partial_x^j \partial_y^k V^i$ or $\partial_x^j \partial_y^k W^i$ to guarantee numerical stability. These additional terms are shown in Equations 24, 25, 26, and in our experiments, we use values $\epsilon_Z = 10^{-5}$, $\epsilon_C = 10^{-10}$, $\epsilon_L = 10^{-2}$. In Equations 24 and 25, the values are chosen to be less than the sensor's quantization noise so that they do not affect accuracy.

$$Z^{i,j,k} = \frac{\left(\partial_x^j \partial_y^k V^i\right)\left(\partial_x^j \partial_y^k W^i\right)}{\left(\partial_x^j \partial_y^k W^i\right)^2 + \epsilon_Z} + Z_0, \tag{24}$$

$$C^{i,j,k} = \frac{\left(\partial_x^j \partial_y^k W^i\right)^2}{\sqrt{\omega_0^{i,j,k}\left(\partial_x^j \partial_y^k W^i\right)^2 + \omega_1^{i,j,k}\left(\partial_x^j \partial_y^k V^i\right)^2 + \omega_2^{i,j,k}\left(\partial_x^j \partial_y^k V^i\right)\left(\partial_x^j \partial_y^k W^i\right) + \left(\partial_x^j \partial_y^k W^i\right)^4 + \epsilon_C}}, \tag{25}$$

$$L(Z - Z^*, C) = \left(\text{mean}\left(|Z - Z^*|^p + \epsilon_L\right)\right)^{1/p}. \tag{26}$$

1.2 Loss function

The area-under-sparsification-curve (AUSC) loss in Equation 23 of [1] is analogous to loss functions used for classification based on area under ROC curves. In classification, optimizing the (ROC) area loss requires a differentiable approximation to the indicator function [2, 3, 4] or non-gradient search [5, 6]. In contrast, the AUSC loss of [1] is based on real-valued errors so its derivatives are readily available:

$$\frac{dL}{dZ} = \lambda \frac{\partial \text{Sort}}{\partial |Z - Z^*|} \frac{d|Z - Z^*|}{dZ},$$
$$\frac{dL}{dC} = \lambda \frac{\partial \text{Sort}}{\partial |Z - Z^*|} \frac{d|Z - Z^*|}{dC}.$$

The Sort operation is differentiable as long as no two elements in C are equal (up to machine precision), which we have yet to encounter in practice.

1.3 Optimization Time

The following table shows the time to convergence for parameter optimization using the AUSC loss and three different p-norm losses on an NVIDIA Quadro K5000 GPU and Intel Xeon CPU E5620 x16 machine. The 1-norm loss converges in roughly half the time of the others.

Loss function	0.5-norm	1-norm	2-norm	AUSC
Training time (sec)	1212	651	1261	1435
Number of iterations	34	19	39	35

1.4 PID controller

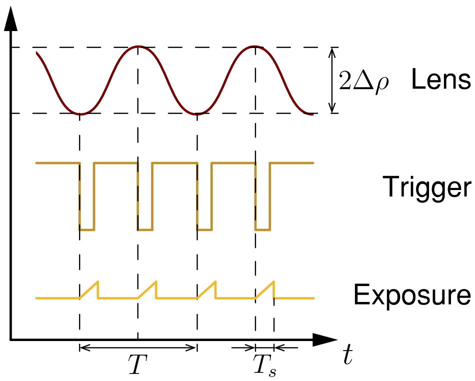
We control the focal distance $Z_f(t)$ using

$$Z_f(t+1) = Z_f(t) + K_p Z_e(t) + K_i \sum_0^t Z_e(\tau) + K_d \left(Z_e(t) - Z_e(t-1)\right),$$

where $Z_e(t) = \bar{Z} - Z_f(t)$, and $\bar{Z}$ is the median of high-confidence depth predictions ($C > 0.999$) at time t. In our experiments we use values $K_p = 0.2$, $K_i = 0.0001$ and $K_d = 0.01$. Other parameters can also be used to achieve specific response characteristics. The focal distance Z_f is converted to the control signal U via Equation 25 in [1].

1.5 Control signals

The control signals that are used to control the camera (Trigger) and the deformable lens are drawn below. The period of the lens signal is $T = 20$ms, and the amplitude is proportional to $2\Delta\rho = 0.8\text{m}^{-1}$. The camera's exposure time is $T_s = 5 \pm 1ms$. The finite exposure time induces a temporal averaging of the lens' dioptric power, so any change in the camera's exposure time will create a small change the effective averaged blur kernels. We find that our algorithm has some insensitivity to these changes, allowing the exposure time to be varied somewhat from that used during training (say, for exposure compensation).



1.6 Textures used for training and testing

We selected ten diverse textures from the Oxford version of the CuRET dataset dataset[1]. We used two for training and eight for validation, as shown in Figure 9. Note that each single texture provides a multitude of per-pixel and per-depth samples.

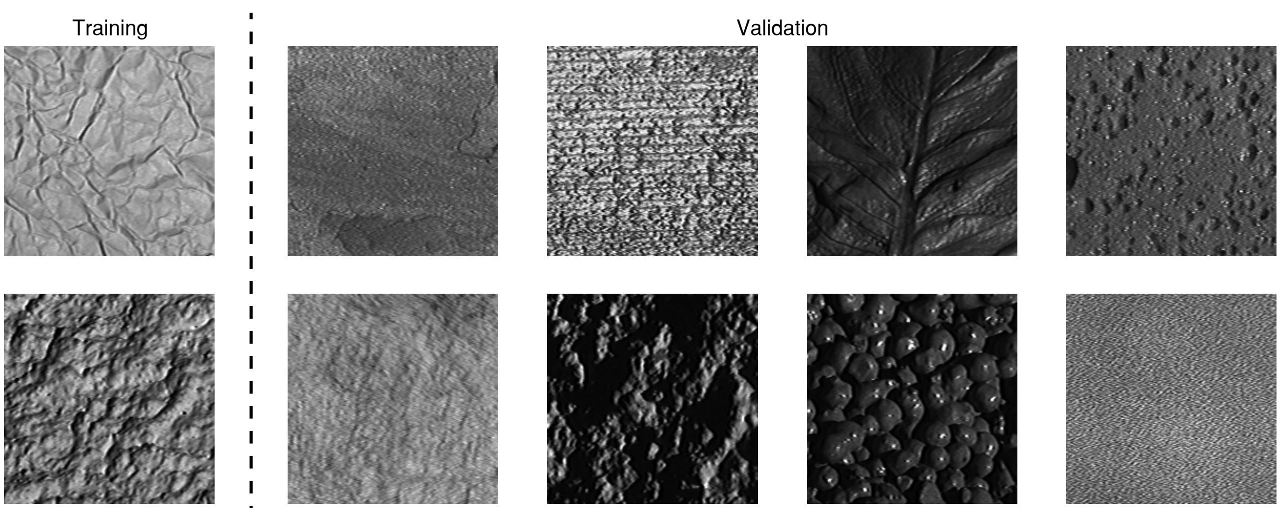


Figure 9: Textures used for training and validation.

[1] http://www.robots.ox.ac.uk/~vgg/research/texclass/

1.7 Performance of accommodation

Figure 10 compares the performance of the focal track sensor on the validation set with and without accommodation. When accommodation is active, the mean error drops to less than 5cm for every confidence level, and the working range increases to more than 75cm. Also, as shown in the bottom right of the figure, accommodation eliminates the dependency of working range on confidence level.

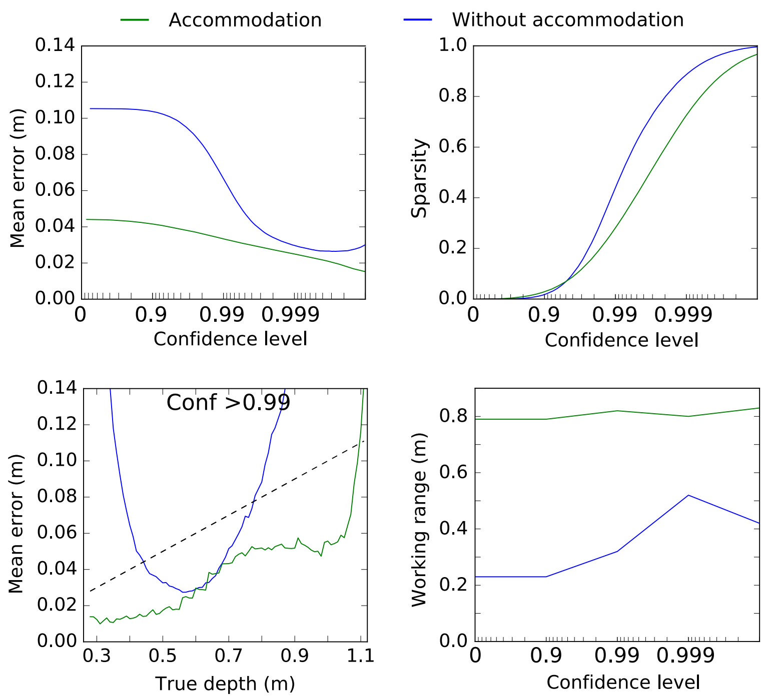


Figure 10: Sensor performance with accommodation (green) and without accommodation (blue). Similar to Figure 5 in [1], this figure shows the average error (top left), sparsity (top right), and working range (bottom right) at different confidence levels; as well as the mean error at different depths (bottom left) for one particular confidence level.

1.8 Additional results

Figure 11 shows depth and confidence maps for scenes that complement the ones in Figure 7 of [1]. Row **G** shows a slanted plane with printed vertical lines whose relative depths are known. Row **F** includes a planar mirror that is partially embossed with a diffuse logo pattern, and located behind a piece of translucent bubble wrap. The depths that are reported in the embossed region and the foreground bubble wrap region are accurate, whereas the depths reported in the mirroring regions correspond to the lengths of two-bounce light paths that connect each sensor pixel to some reflected point on the underside of the bubble wrap.

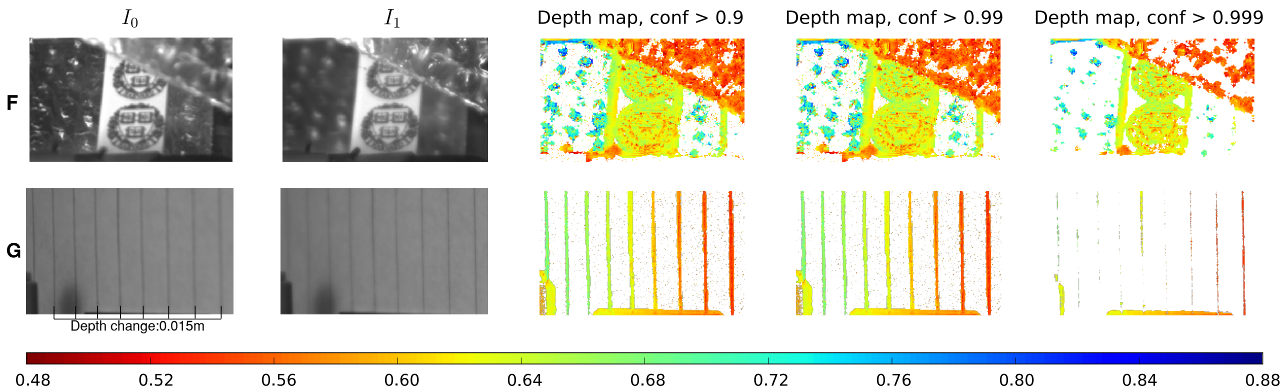
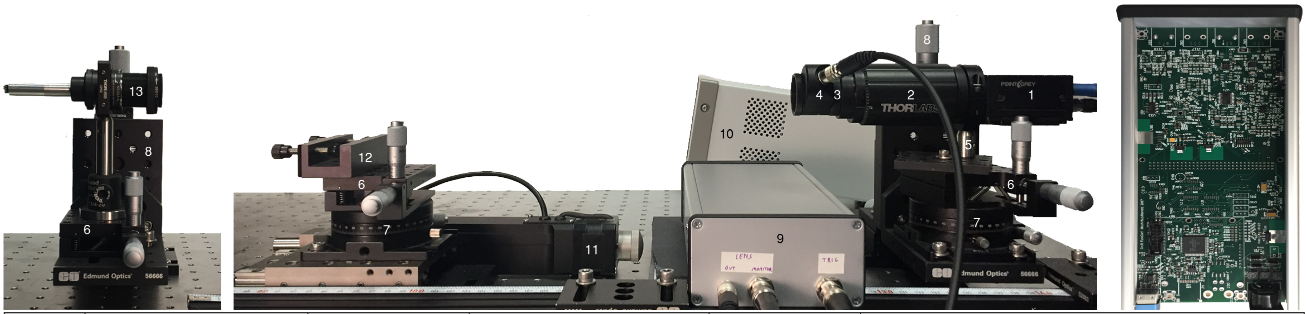


Figure 11: Depth and confidence results that complement those in Figure 7 in [1]. Reported depth in meters.

2 Hardware and optics

2.1 List of parts

The figure and table shows the hardware details of our setup. Object 1-4, 9 are for the focal track sensor, 6, 8, 13 are for alignment, 6, 7, 10-12 are for data collection and calibration. The right part of the figure shows the board of the signal generator (9). The circuit diagram is available upon request.



No.	Component	Source	Part Number	Quantity	Description
1	Camera	Point Grey	GS3-U3-23S6M-C	1	Monochrome, outside trigger powered by USB
2	Lens Tube Zoom Housing	Thorlabs	SM1NR1	1	SM1 thread, $\varnothing 1''$, Non-rotating $2''$ Travel
3	Lens	Thorlabs	LA1509-A	1	Planar-convex, $\varnothing 1''$, $10m^{-1}$, AR coated (350-700nm)
4	Deformable Lens	Optotune	EL-10-30-C-VIS-LD-MV	1	Tuning range $[-1.5m^{-1}, 3.5m^{-1}]$, $\varnothing 1''$, coated (400-700nm)
5	Lens Tube Mounts	Thorlabs	SM1TC+TR075	1	
6	Pitch & Yaw Platform	Thorlabs	PY003	3	
7	Rotation Platform	Thorlabs	PR01+PR01A	2	
8	X-Y Translation Stage	Thorlabs & EO	$2\times$PT1+PT101+PT102+EO56666	2	
9	Signal Generator	Custom		1	
10	Stepper Motor Controllers	Thorlabs	BSC201	1	Powered by 110V, connected with PC via USB
11	Translation Stage	Thorlabs	LNR50S	1	Controlled and powered by 10
12	Wide Plate Holder	Thorlabs	FP02	1	
13	Laser	Thorlabs	CPS532	1	Mounted with AD11F, SM1D12SZ, CP02, NE20A-A, SM1D12D

References

[1] Qi Guo, Emma Alexander, and Todd Zickler. Focal track: Depth and accommodation with oscillating lens deformation. In *International Conference on Computer Vision*. IEEE, 2017.

[2] Stephan Dreiseitl. Training multiclass classifiers by maximizing the volume under the roc surface. In *International Conference on Computer Aided Systems Theory*, pages 878–885. Springer, 2007.

[3] Alan Herschtal, Bhavani Raskutti, and Peter K Campbell. Area under roc optimisation using a ramp approximation. In *Proceedings of the 2006 SIAM International Conference on Data Mining*, pages 1–11. SIAM, 2006.

[4] Alain Rakotomamonjy. Optimizing area under roc curve with svms. In *ROCAI*, pages 71–80, 2004.

[5] Xiaofen Lu, Ke Tang, and Xin Yao. Evolving neural networks with maximum auc for imbalanced data classification. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 335–342. Springer, 2010.

[6] Yang Zhi, Guo-en Xia, and Wei-dong Jin. Optimizing area under the roc curve using genetic algorithm. In *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, volume 1, pages 672–675. IEEE, 2011.