InDepth: Real-time Depth Inpainting for Mobile Augmented Reality

Yunfan Zhang (Duke University), **Tim Scargill (Duke University)**, Ashutosh Vaishnav (Aalto University), Gopika Premsankar (University of Helsinki), Mario Di Francesco (Aalto University), Maria Gorlatova (Duke University)

Duke UNIVERSITY

Aalto University

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

Duke

Presentation Contents

- Background and Motivation
- Depth Sensing for Mobile AR
- DNN Design
- Evaluation
  - DNN
  - System
  - User Experience
- Conclusion and Future Work

Duke

Mobile Augmented Reality

- Augmented reality (AR): the overlaying of virtual content onto a view of the real world
- Mobile AR facilitates this through portable, handheld or wearable devices
- Wide variety of use cases, from e-commerce to education and medicine

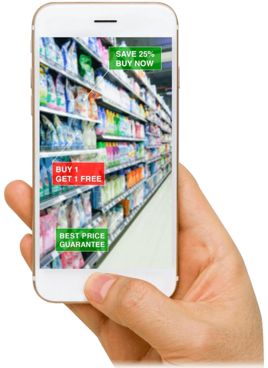
Duke

Mobile AR Devices

- Specialized headsets (Microsoft HoloLens 2, Magic Leap One)

- Smartphones and tablets, supported by ARCore (Android) and ARKit (iOS)

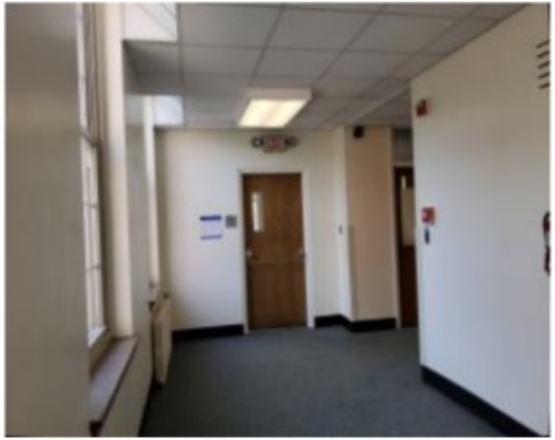
- Onboard sensors map environment and track device pose within it



AR Device Depth Sensors

- Headsets and high-end smartphones and tablets equipped with depth sensors
- Time-of-Flight (ToF) cameras provide real-time depth data; low power, high frame rate
- Raw depth maps incomplete due to range limitations and reflectance properties

RGB image



Raw depth map

Virtual Object Scale Errors

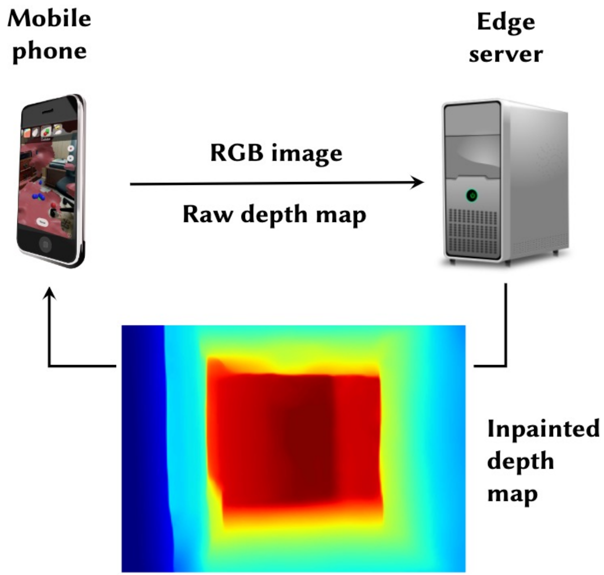
- Existing depth map completion methods (e.g., [1, 2]) result in inaccurate depth maps, causing errors in the size of rendered virtual objects

- Majority of respondents to our online survey on previous AR experiences indicated virtual object size errors are a somewhat frequent or a very frequent issue in mobile AR

[1] Jonathan T Barron and Ben Poole. 2016. The fast bilateral solver. In *ECCV*.
[2] Yinda Zhang and Thomas Funkhouser. 2018. Deep Depth Completion of a Single RGB-D Image. In *IEEE CVPR*.

InDepth Paper Contributions

- ToF18K dataset: 18.6K depth maps + RGB
- New DNN architecture which obtains accurate depth maps with latency as low as 8.7ms
- Mean absolute error of depth estimates of 20cm compared to 78cm in ARCore DepthLab
- In a user study 87% more participants rated virtual objects the correct size with InDepth than DepthLab

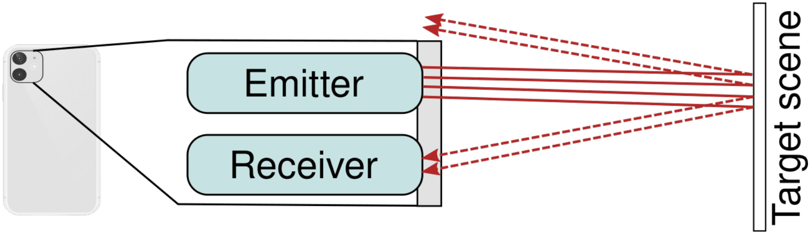


Time-of-Flight (ToF) Depth Cameras

- Consist of infrared emitter and receiver, measure properties of reflected light to estimate distance

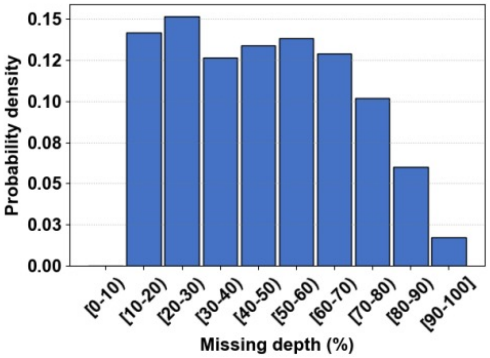
- We focus on indirect ToF: modulate light to detect phase shifts

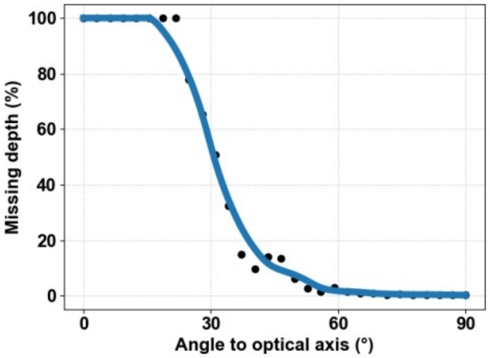
- Currently used on Samsung Galaxy Note 10+, Huawei P40 Pro, Microsoft HoloLens 2 and Magic Leap One

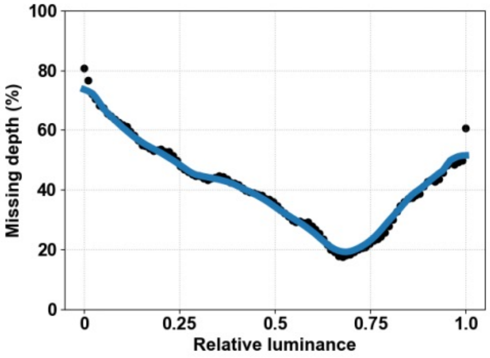


ToF18K Dataset: Limitations of ToF Cameras

- 18.6K depth maps (plus RGB images) collected on a Samsung Galaxy Note 10+, in variety of indoor scenes
  - 47.2% of total depth pixels were missing
  - 50% of captured maps had more than 40% missing pixels
  - Errors due to distance and surface brightness or orientation



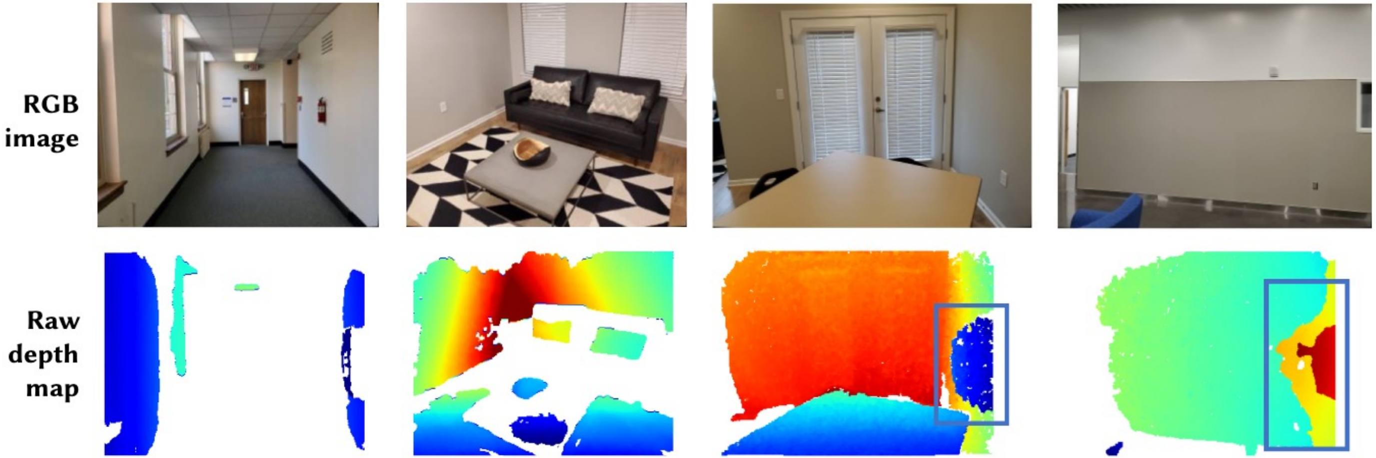




Duke

Sample Issues in Depth Maps

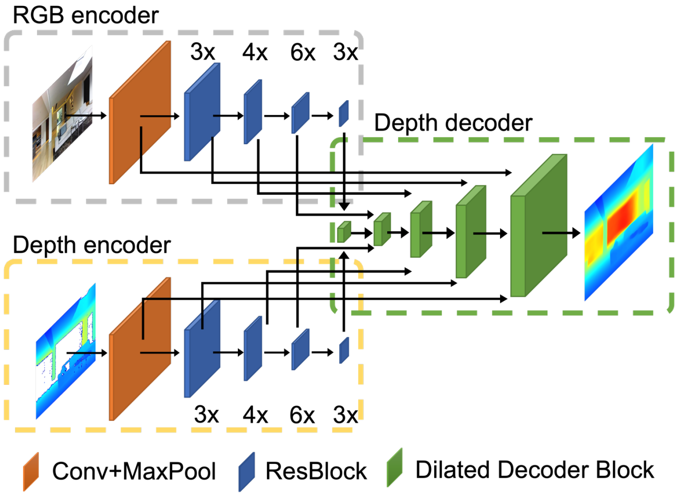
- Measurement errors and artifacts caused by:
  - Distant surfaces, surfaces parallel to camera axis
  - Very bright or dark materials
  - Undesired reflections

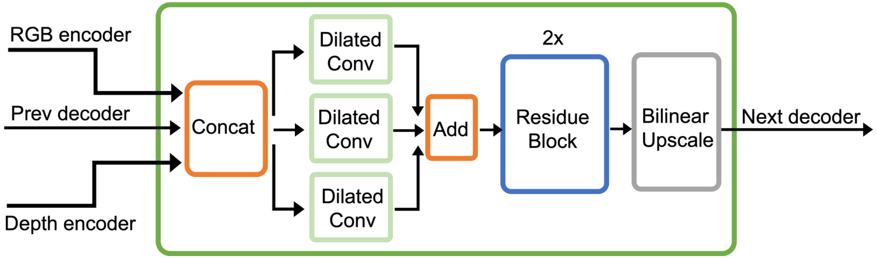


Duke

DNN Architecture

- Two-branch encoder to extract features from RGB and depth inputs
- Dilated decoder block for depth data (right)





Data Augmentation and Training

- Trained on Matterport 3D RGB-D dataset [3]
- Depth artifacts added based on our ToF18K dataset
- Custom hybrid loss function: weighted sum of Virtual Normal Loss [4], gradients of depth estimation error, and BerHu loss

[3] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3D: Learning from RGB-D Data in Indoor Environments. In *International Conference on 3D Vision (3DV)*.
[4] W. Yin, Y. Liu, C. Shen, and Y. Yan. 2019. Enforcing Geometric Constraints of Virtual Normal for Depth Prediction. In *IEEE ICCV*.

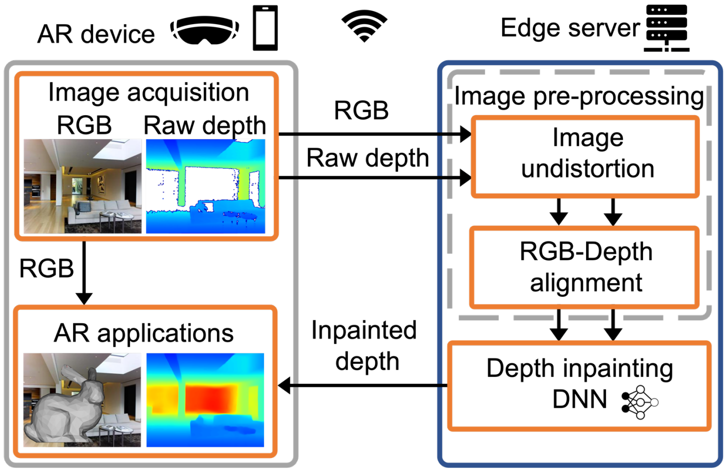
Data-Driven DNN Evaluation

Method	MAE (m)	RMSE (m)	1.25	Latency (ms)
Bilateral filtering	0.774	1.978	0.613	1457.1
Markov random fields	0.618	1.675	0.651	685.0
Anisotropic diffusion	0.610	1.653	0.663	896.0
[Zhang and Funkhouser, CVPR 2018]	0.461	1.316	0.781	4036.6
[Huang et al., ICCVW 2019]	0.342	1.092	0.850	70.2
InDepth DNN	**0.294**	**1.008**	**0.876**	**8.7**

Metrics: MAE: Mean Absolute Error; RMSE: Root Mean Square Error; 1.25: percentage of pixels within the 1.25 error range; inference latency.

Software Components and Testbeds

- Realized with OpenCV, PyTorch, and TensorRT
- Two edge testbeds (workstation-class and embedded-class)



Duke

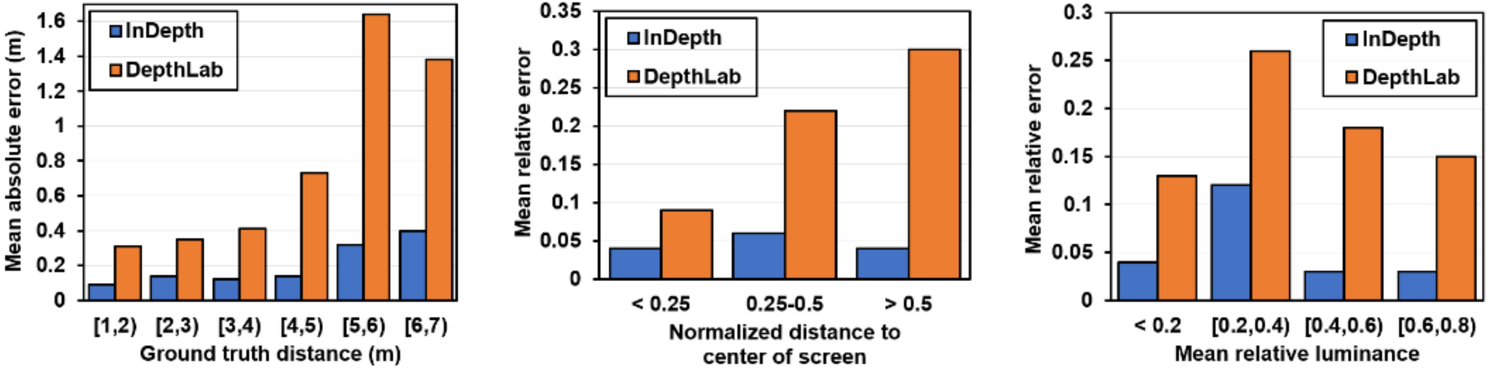
System Performance Evaluation

Edge testbed latency (ms)	w-class	e-class
End-to-end	**26.3**	36.5
Communication overhead	13.1	9.2
Image pre-processing	4.5	10.8
DNN inference	8.7	16.5

Accuracy	MAE (m)	RMSE (m)	1.25
Real-world experiments	**0.238**	**0.468**	**0.941**
Matterport 3D reference	0.294	1.008	0.876

Comparison with State of the Art

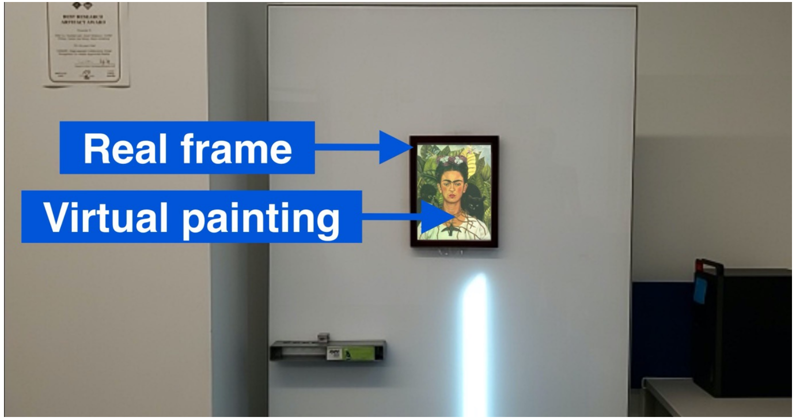
- ARCore DepthLab: open-source software to view and interact with depth maps, generated by the ARCore Depth API for Android
- Higher depth estimation error than InDepth

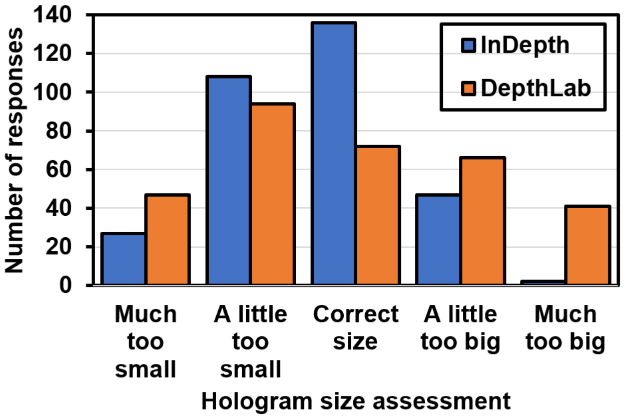


Duke

User Experience: InDepth vs State of the Art

- Participants rated images of real painting in virtual frame
- 87% more participants rated virtual objects rendered with InDepth of the correct size compared to DepthLab





Duke

Conclusions and Future Work

- InDepth DNN infers depth for missing regions in a depth map with a low latency; outperforms state of the art in both data-driven and user evaluations

- Future work includes incorporating technique into a full AR system with 6DoF tracking

- Fascinating opportunities for user studies which explore other types of virtual content errors

Acknowledgments and Resources

- NSF grants CSR-1903136, CNS-1908051, and CAREER-204607
- Academy of Finland grant numbers 326346, 319710, 332307 and 3388
- IBM Faculty Award

- Implementation code and ToF18K dataset available at: https://github.com/InDepthOpensource/Code

Duke