Multi range Real-time depth inference from a monocular stabilized footage using a Fully Convolutional Neural Network

P.Ashwini [1], V.Chiranjeevi [2], S.Nagamani [3]

[1]Assistant Professor, Swarna Bharathi Institute of Science & Technology, India, E-mail: ashwini.podila@gmail.com.
[2]Assistant Professor, Swarna Bharathi Institute of Science & Technology, India, E-mail: chiru508@gmail.com.
[3]Assistant Professor, Swarna Bharathi Institute of Science & Technology, India, E-mail: nagamanikunchipudi@gmail.com.

ABSTRACT:

Applying our proposed neural network architecture to UAV footage shot in static environments, we can extract depth maps from stabilized monocular footage. A new navigational synthetic dataset is used for training purposes; it simulates aerial imagery shot in stiff situations using a gimbal stabilized monocular camera. We suggest a multi-range architecture for unrestricted UAV flying based on this network, which uses flight data from sensors to create accurate depth maps for an outdoor setting free of obstructions. Using both simulated and real-world UAV f-light data, we test our approach. Results for synthetic scenes with a little orientation noise are provided quantitatively, demonstrating that our multi-range architecture enhances depth inference. (a) – (c) In Figure 1. One option for stabilizing a camera is a mechanical gimbal; another is to use dynamic cropping from a fish-eye lens; and a third is to use a handheld camera. An accompanying movie provides a more in-depth presentation of our findings.

1. INTRODUCTION:

An essential challenge for UAVs and autonomous vehicles in general is scene understanding from vision. Finding out how deep each pixel is in a series of camera-shot images is the main focus of this work. Since the majority of UAV flight systems incorporate a speed estimator, we may assume that we know the camera's velocity and, by extension, the displacement between the two frames, which resolves the scale invariance uncertainty in the depth map.

Environment scanning, using depth-based sense and avoid algorithms, and lightweight embedded systems with only a monocular camera are just a few of the difficulties that could benefit from this issue's resolution. Putting less emphasis on depth The UAV can be liberated from its constraints in terms of weight, cost, and functionality by including sensors like stereo vision, ToF cameras, LiDar, or Infra Red emitters and receivers. In particular, most RGB-D sensors have limited range and can be inefficient when we need long-range information, such in trajectory planning, and some of them can't work in sunlight (like IR and ToF) [7]. Depth from motion, in contrast to RGB-D sensors, is displacement-agnostic, meaning it can withstand fast speeds or large distances, as we can choose from a variety of displacements in prior frames.

Using a synthetic dataset and an input-only fully convolutional neural network trained on a pair of images captured at different times, we developed an end-to-end learning architecture for calculating these depth maps. The depth is supplied as an output without any preprocessing, such as optical flow computation or visual odometry.

Fig. 1. Camera stabilization can be done via a) mechanic gimbal or b) dynamic cropping from fish-eye camera, for drones or c) hand-held cameras

During training, we used a fixed displacement magnitude on a dataset of picture pairs that had random translation motions but no rotation.Two factors provide credence to the idea that videos without rotation can nevertheless be effective: Consumer drones equipped with inertial measurement unit (IMU) stabilised cameras or handheld steady-cams have made hardware rotation compensation a thing of the past (Fig 1). The vestibulo-ocular reflex (VOR) and human eyesight are somewhat related to this movement [2]. Even though turning our heads doesn't cause our eyeballs to point in a specific direction, our inner ear and other biological senses enable us to adjust for parasite rotation when we look in a certain direction.

An technique for actual condition depth inference from a stabilized UAV is proposed, making use of the trained network. The genuine depth map is computed using sensor displacement, which differs from the synthetic constant displacement images solely in terms of size. A posteriori optimization of the depth inference is also possible with our network's output. We can reduce the depth error for the next inference by modifying the frame shift to obtain a displacement that would give the network the same disparity distribution as during training. For instance, the ideal displacement between any two frames is greater at long distances, leading to a larger shift. In addition, we achieve a high level of accuracy for both nearby and distant objects, regardless of distance, provided that the UAV is sufficiently moved away from them, by utilizing multiple batch inference to calculate numerous depth maps focused on a certain range. These maps are then fused together.

2. RELATED WORK

Many other types of visual problems, including categorization [13] and hand-written digits recognition [14], have lately seen extensive usage of Deep Learning and Convolutional Neural Networks.A variety of training solutions have been developed to solve depth from vision, one of the neural network problems examined. A neural network may learn end-to-end depth or disparity on certain datasets [6], [19], [15], [22], [4]. Unsupervised training for depth from a single image or disparity between two frames of a stereo setup have both made use of projection error [20], [23] and [12], [5], respectively. Although intriguing, depth from a single image has a big flaw—overfitting. While decorrelating them is possible, the network is not given any motion during inference, and the depth that results is inferred from context.While this method may be enough for situations when the road is directly in front of the camera, it may not be suitable for use in UAV flights due to the potentially diverse sights that may be encountered. However, for aerial stabilized film to look realistic, depth from a stereo pair is needed because it only implies one lateral movement and doesn't include a forward component.

The majority of algorithms do not depend on deep learning, and the state-of-the-art approaches to depth from complicated movement captured by a monocular camera typically make advantage of motion, particularly structure from motion [1, 17, 11]. A sparse depth map is inferred using prior knowledge about the scene; the density of this map often increases with time. The sparse point-cloud based 3D maps produced by these techniques, which are also known as SLAM, necessitate extensive computation to maintain track of the scene structure and align newly detected 3D points to the existing ones. Unstructured movement, which includes translation and rotation of varying magnitudes, is the usual application for these methods.

We aim to create a dense depth map (i.e., one in which every point has a valid depth) by combining two separate but equally timed images taken by the same camera, with no further information about the scene or the subject's movement (apart from the fact that they are not rotated) and by applying a scale factor.

3. END-TO-END LEARNING OF DEPTH INFERENCE

We setup an end-to-end learning workflow by training a neural network to explicitly predict the depth of every pixel in a scene, from an image pair with constant displacement value. This was inspired by flow estimation and disparity, a problem to which there exist many very convincing methods [8], [10].

3.1 STILL BOX DATASET

Using Blender, a rendering program, we construct our own synthetic dataset and use it to create an infinite number of stiff scenes. These scenes are made up of fundamental 3D primitives, such as cubes, spheres, cones, and tores, and their textures are randomly selected from an image set that was scraped from Flickr (see Fig. 2).The scene is filled with arbitrarily sized items, and walls are created at enormous distances, making it look like the camera is within a box (thus the name). Every scene features the same constant speed camera movement in an equally distributed random direction. This movement can be in any direction, from forwards and backwards to laterally (thus resembling stereo vision).

3.2 DATASET AUGMENTATION

Our dataset is structured as a film with ten images, where each frame is associated with its ground truth depth. Because of this, we can decide on a posteriori distances distribution where the time difference between the two frames is changeable. A three-frame baseline shift allows us to, for example, presume a depth three times greater than for two consecutive frames (a one-frame shift). Negative shift is another option to think about; it will simply alter the direction of displacement and have no effect on the speed value. This way, we may avoid training a scene recognition algorithm that would underperform on a validation set due to over-fitting and obtain more uniformly distributed depth values to learn with a fixed dataset size. We can also de-correlate photos based on depth.

3.3 TRAINING FOR DEPTH INFERENCE

Our network, which we call DepthNet, takes its cues from the original FlowNetS [3], which was developed for flow inference. We present a synopsis of its architecture (Fig. 3) and performance here; a detailed description can be found in [18]. After each convolution (with the exception of depth modules), there is a layer for ReLU activation and Spatial Batch Normalization. Rectified Linear Unit (ReLU) is the common activation layer [21], and batch normalization promotes convergence and stability during training by transforming a convolution's output from a batch of numerous inputs into a single value with a mean and standard deviation of 1 [9]. The input is reduced to one feature map—the predicted depth map—at a given scale by the use of convolution modules, which are known as depth modules. Notably, FlowNetS originally employed LeakyReLU, which exhibits a non-null slope for negative values; nevertheless, our problem was better addressed by ReLU, as demonstrated by our testing.

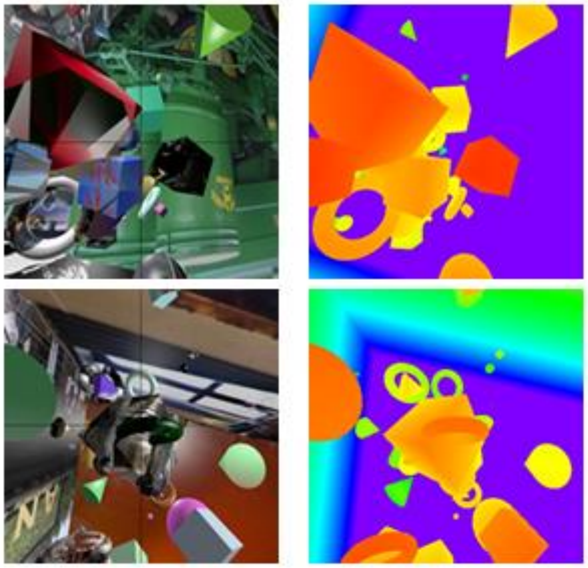
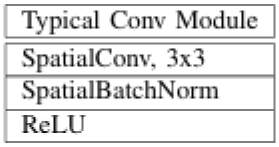
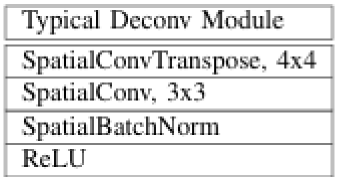


Fig. 2. Some examples of our renderings with associated depth maps (red is close, purple is far)

A key component of this network is the merging of upsampled feature maps with their matching previous convolution outputs; for instance, combining Conv2 and Deconv5 outputs. Information that is more firmly tied to pixels (due to fewer downsampling convolutions) is subsequently utilized for reconstruction and is correlated with higher semantic information.



Typical Conv Module
SpatialConv, 3x3
SpatialBatchNorm
ReLU



Typical Deconv Module
SpatialConvTranspose, 4x4
SpatialConv, 3x3
SpatialBatchNorm
ReLU

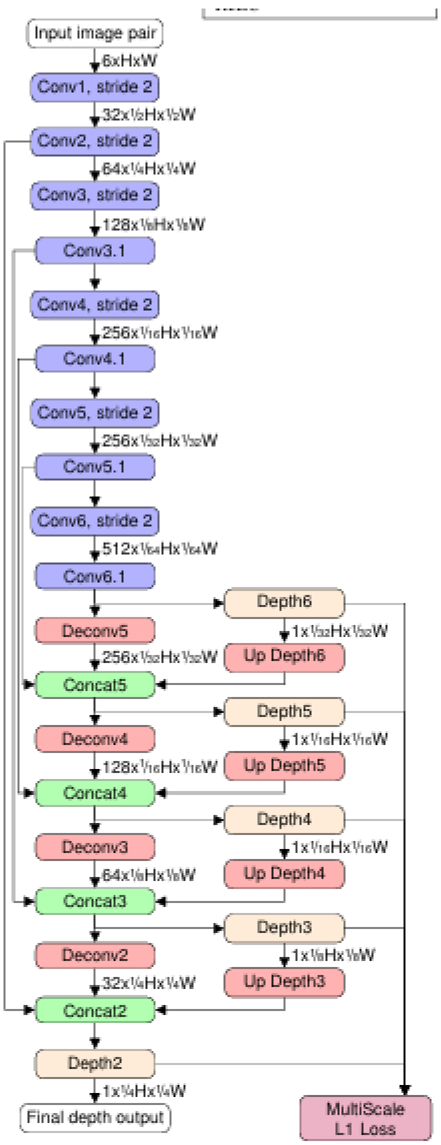


Fig. 3. DepthNet structure parameters, Conv and Deconv modules detailed above

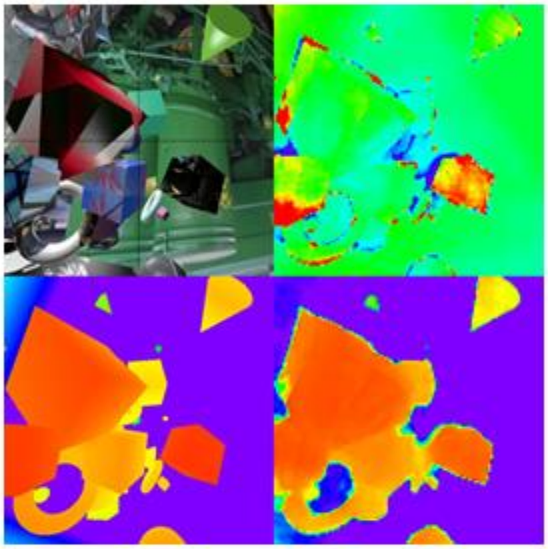


Fig. 4. Findings for 512 × 512 photos taken from DepthNet64 progresses from 128 to 256 to 512. Up top: input, down below: Ground Truth depth, lower-right: our network output Error (128x128), green indicates no error, while red indicates depth that is exaggerated, blue that is understated With its straightforward supervised learning procedure, this multi-scale design has shown to be highly effective for flow and disparity computing. The primary goal of this experiment is to demonstrate the efficiency of direct depth estimate with respect to unknown translation. We adopt a multi-scale criterion similar to FlowNetS, where each scale is assessed by its L1 reconstruction error:

$$Loss = \sum_{s \in scales} \gamma_s \frac{1}{H_s W_s} \sum_i \sum_j |\beta_s(i,j) - \zeta_s(i,j)| \quad (1)$$

Where,

- γs is the weight of the scale, arbitrarily chosen.
- (Hs,Ws) = (1/2sH,1/2sW) are the height and width of the output.
- ζs is the scaled depth groundtruth, using average pooling.
- βs is the ouput of the network at scale s

In addition to the more traditional techniques like flips and rotations, we augment the dataset with data utilizing various shifts, as mentioned before. Assuming its depth is 100m everywhere, we additionally give sample pairs without shift and clamp depth to a maximum of 100m. This means that the trained network can only deduce depths below 100 meters. From 64x64 to 512x512, we trained on images with a wide range of input sizes. Learned mean L1 reconstruction error is displayed in Fig. 4. The network's output is downsampled by a factor of 4 in relation to the input size, just like FlowNetS.

Network	L1Error		RMSE	
	train	test	train	test
FlowNetS$_{64}$	1.69	4.16	4.25	7.97
DepthNet$_{64}$	2.26	4.49	5.55	8.44
FlowNetS$_{64\rightarrow128\rightarrow256\rightarrow512}$	0.658	**2.44**	1.99	**4.77**
DepthNet$_{64\rightarrow128}$	1.20	3.07	3.43	6.30
DepthNet$_{64\rightarrow128\rightarrow256}$	0.876	**2.44**	2.69	4.99
DepthNet$_{64\rightarrow128\rightarrow256\rightarrow512}$	1.09	2.48	2.86	**4.90**
DepthNet$_{64\rightarrow512}$	1.02	2.57	2.81	5.13
DepthNet$_{512}$	1.74	4.59	4.91	8.62

TABLE I. Quantitative results for depth inference networks. FlowNetS is modified with 1 channel outputs (instead of 2 for flow), trained from scratch for depth with Still Box, subscript indicates fine tuning process.

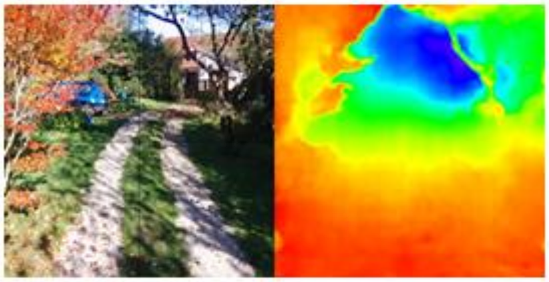


Fig. 5. Result on 512x512 real images input from a Bebop drone footage

4. CONCLUSION AND FUTURE WORK

Along with a methodology for dynamic range real flight computation, a very complete dataset for stabilized film analysis, and a new approach to calculating dense depth maps from motion, we also offered a method. All types of path planning, including long-range obstacle bypassing and collision avoidance, can be covered by this algorithm's very flexible application to depth-based sense and avoid algorithms.
Watching this video will give you a better idea of what the results are. The download link for the DeepNet Results video is http://perso. ensta-paristech.fr/˜manzaner/Download/ECMR2017.mp4. In the future, we aim to apply our path planning algorithm and build a real-world fine-tuning dataset utilizing film from unmanned aerial vehicles (UAVs) and an initial comprehensive 3D offline scan. Instead of relying on subjective metrics, we could now quantify the quality of our network for real footages. Unsupervised methods based on re-projection errors, as described in [23], are another option.

Our team is also confident in our network's ability to be enhanced with reinforcement learning features, which might lead to a fully functional end-to-end sense and avoid system for monocular cameras. However, the requirement that a scene be rigid is the main limitation of our method. This will never happen, and although while unmanned aerial vehicle footage is less likely to have objects in motion than autonomous driving film, this problem will still arise anytime a moving target needs to be followed. As in [20], it may be necessary to calculate an explicit camera and moving target movement equation in order to resolve this issue. Regardless, fully convolutional networks alone may not be enough to solve this problem, as we demonstrated in this study.

REFERENCES:

[1] Cesar Cadena, Luca Carlone, Henry Carrillo, Yasir Latif, Davide Scaramuzza, Jos´ e Neira, Ian Reid, and John J Leonard. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age. IEEE Transactions on Robotics, 32(6):1309 1332, 2016.

[2] Lorente De N´o, R. Vestibulo-ocular reflex arc. Archives of Neurology & Psychiatry, 30(2):245–291, 1933.

[3] A. Dosovitskiy, P. Fischer, E. Ilg, P. H¨ausser, C. Hazrba, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. Flownet: Learning optical f low with convolutional networks. In IEEE International Conference on Computer Vision (ICCV), 2015.

[4] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map pre diction from a single image using a multi-scale deep network. In Advances in neural information processing systems, pages 2366–2374, 2014.

[5] Ravi Garg, Vijay Kumar B. G, and Ian D. Reid. Unsupervised CNN for single view depth estimation: Geometry to the rescue. CoRR, abs/1603.04992, 2016.

[6] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 3354–3361. IEEE, 2012.

[7] Raia Hadsell, Pierre Sermanet, Jan Ben, Ayse Erkan, Marco Scoffier, Koray Kavukcuoglu, Urs Muller, and Yann LeCun. Learning long range vision for autonomous off-road driving. Journal of Field Robotics, 26(2):120–144, 2009.

[8] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical f low estimation with deep networks. arXiv preprint arXiv:1612.01925, 2016.

[9] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167, 2015.

[10] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry. End-to-End Learning of Geometry and Context for Deep Stereo Regression. ArXiv e-prints, March 2017.

[11] Georg Klein and David Murray. Parallel tracking and mapping for small ar workspaces. In Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on, pages 225 234. IEEE, 2007.

[12] Kishore Reddy Konda and Roland Memisevic. Unsupervised learning of depth and motion. CoRR, abs/1312.3429, 2013.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems, pages 1097–1105, 2012.

[14] Yann LeCun, L´ eon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.

[15] Wenjie Luo, Alexander G Schwing, and Raquel Urtasun. Efficient deep learning for stereo matching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 5695–5703, 2016.