Supervised Monocular Depth Estimation via Stacked Generalization

Chinchuthakun Worameth
Department of Transdisciplinary Science and Engineering
Tokyo Institute of Technology

I. Introduction

Monocular depth estimation has been extensively studied because it offers relatively low computational cost and energy consumption compared to stereo depth estimation. Despite its ill-posed nature, deep learning has rapidly advanced the progress in this task and continuously presented state-of-the-art results by exploring novel network architectures.

This paper studies a stacked generalization (SG) framework in monocular depth estimation. Several state-of-the-arts architectures [1]–[3] were adopted as base learners with some adjustments. An additional encoder-decoder network was employed as the meta-learner and trained to combine the predicted depth maps. This work demonstrates a successful application of SG in supervised monocular depth estimation on NYU Depth V2 dataset [4], achieving better performance on standard evaluation metrics. Ablation studies also serve as guidelines on how to effectively adopt this framework.

II. Related Works

Supervised Monocular Depth Estimation usually leverages deep learning. Previous works have introduced many architectures based on CNN and Visual Transformer. Other strategies have also been explored, including multi-tasking, domain adaptation, and lightweight network for fast inference.

Ensemble deep learning refers to approaches that infer a final prediction from multiple neural networks. This research focuses on stacked generalization, which linearly combines predictions from base learners. The optimal weight combination is learned with a neural network, producing a pixel-wise coefficient tensor as an output. While it has been adopted in many tasks, there is no application in monocular depth estimation according to a recent survey.

III. Methodology

A. Modifications of base learners

To cope with hardware's limitations and accelerate experimentation, models' size and latency were reduced by

1) Employing *GhostNet* as the encoder for all architectures.
2) Replacing convolution with *depthwise separable convolution* in [1] and [2] except for atrous convolution layers.
3) Interpolating *after* convolution instead of before in [1].

B. Design choices of meta-learner

Four design aspects were considered in ablation studies, viz.

1) **Training pipeline**: train base learners and controller (1) simultaneously or (2) sequentially.
2) **Transfer learning**: when training a meta-learner, (1) freeze base learners' parameters or (2) fine-tune them.
3) **Available input**: when training a meta-learner, specify its input as (1) raw RGB image; (2) predictions from base learners; or (3) both of them.
4) **Backbone architecture**: feature maps are extracted with (1) GhostNet, (2) MobileNetV2, or (3) DenseNet-161.

C. Loss functions

Inspired from [1], Pixel-wise depth loss was employed in [1]–[3]. Bichamfer loss was also utilized in [1] to encourage the distribution of bin centers to follow the ground truth.

IV. Experiment

Models were implemented in PyTorch and trained on 10 GTX 1080 Ti GPUs via distributed learning. Data augmentations, described in [1], were used to avoid overfitting. Evaluations on standard metrics were conducted with the official split of NYU Depth V2 dataset. Performance of base learners and ensembles are shown in Tables below, respectively.

	Variant	#Params	*higher is better*			*lower is better*				
			$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	REL	Sq REL	RSME	RSME log	log10
MOD	Adabins	17.2M	0.813	0.965	0.992	0.146	0.106	0.498	0.178	0.060
	BTS	8.9M	0.855	0.973	0.994	0.120	0.075	0.435	0.156	0.052
	LDRN	14.9M	0.831	0.967	0.993	0.130	0.085	0.455	0.167	0.056
SAME	BTS #1	8.9M	0.862	0.973	0.999	0.120	0.074	0.427	0.155	0.052
	BTS #2		0.855	0.972	0.993	0.121	0.077	0.434	0.157	0.053
	BTS #3		0.852	0.973	0.993	0.121	0.076	0.438	0.157	0.053
	BTS #4		0.856	0.973	0.994	0.119	0.073	0.431	0.155	0.052
	BTS #5		0.856	0.972	0.994	0.121	0.075	0.431	0.155	0.052
CV-BTS	BTS #1	8.9M	0.854	0.973	0.994	0.121	0.076	0.437	0.157	0.053
	BTS #2		0.852	0.971	0.994	0.122	0.076	0.439	0.158	0.053
	BTS #3		0.853	0.972	0.993	0.121	0.077	0.442	0.158	0.053
	BTS #4		0.856	0.973	0.993	0.122	0.078	0.433	0.156	0.052
	BTS #5		0.855	0.972	0.993	0.122	0.077	0.438	0.157	0.053
CV-LDRN	LDRN #1	14.9M	0.840	0.970	0.993	0.128	0.082	0.450	0.164	0.055
	LDRN #2		0.842	0.968	0.992	0.130	0.084	0.451	0.165	0.055
	LDRN #3		0.841	0.969	0.993	0.128	0.081	0.445	0.163	0.055
	LDRN #4		0.851	0.971	0.993	0.125	0.081	0.437	0.159	0.054
	LDRN #5		0.836	0.968	0.992	0.131	0.086	0.456	0.166	0.056

	Variant	#Params	*higher is better*			*lower is better*				
			$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$	REL	Sq REL	RSME	RSME log	log10
MOD	Baseline	-	0.8573	0.9758	0.9949	0.1215	0.0735	0.4245	0.1533	0.0517
	I-F-G	4.3M	0.8591	0.9753	0.9946	0.1191	0.0720	0.4261	0.1529	0.0513
	O-F-G	4.3M	0.8602	0.9749	0.9943	0.1186	0.0722	0.4240	0.1527	0.0511
	IO-F-G	4.3M	0.8607	0.9751	0.9945	0.1190	0.0718	0.4240	0.1525	0.0512
	IO-F-M2	3.7M	0.8589	0.9747	0.9942	0.1191	0.0727	0.4270	0.1532	0.0514
	IO-F-D161	30.3M	0.8607	0.9749	0.9944	0.1184	0.0717	0.4223	0.1523	0.0510
SAME	Baseline	-	0.8658	0.9759	0.9948	0.1161	0.069	0.4153	0.1491	0.0503
	IO-F-G	4.3M	0.8663	0.9761	0.9948	0.116	0.0689	0.414	0.1489	0.0503
	IO-F-D161	30.3M	0.8661	0.976	0.9948	0.1162	0.0693	0.4144	0.149	0.0503
CV-BTS	Baseline	-	0.8644	0.9761	0.9948	0.1171	0.0707	0.4204	0.1504	0.0508
	IO-F-G	4.3M	0.8649	0.976	0.9948	0.1171	0.0707	0.4194	0.1503	0.0508
	IO-F-D161	30.3M	0.8654	0.9762	0.9947	0.1169	0.0706	0.418	0.1501	0.0507
CV-LDRN	Baseline	-	0.8596	0.9749	0.9944	0.1206	0.0724	0.4202	0.1529	0.0516
	IO-F-G	4.3M	0.8594	0.9743	0.9941	0.1208	0.0744	0.4226	0.1538	0.0517
	IO-F-D161	30.3M	0.8570	0.9741	0.9940	0.1206	0.0741	0.4252	0.1546	0.0521

References

[1] S. F. Bhat, I. Alhashim, and P. Wonka, "Adabins: Depth estimation using adaptive bins," *CoRR*, vol. abs/2011.14141, 2020. [Online]. Available: https://arxiv.org/abs/2011.14141

[2] J. H. Lee, M. Han, D. W. Ko, and I. H. Suh, "From big to small: Multi-scale local planar guidance for monocular depth estimation," *CoRR*, vol. abs/1907.10326, 2019. [Online]. Available: http://arxiv.org/abs/1907.10326

[3] M. Song, S. Lim, and W. Kim, "Monocular depth estimation using laplacian pyramid-based depth residuals," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2021.

[4] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Computer Vision – ECCV 2012*, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 746–760.