On regression losses for deep depth estimation

Marcela Carvalho, Bertrand Le Saux, Pauline Trouvé-Peloux, Andrés Almansa, Fréderic Champagnat

ON REGRESSION LOSSES FOR DEEP DEPTH ESTIMATION

Marcela Carvalho†, Bertrand Le Saux†, Pauline Trouvé-Peloux†, Andrés Almansa, Frédéric Champagnat†*

† ONERA - French Aerospace Lab, Palaiseau, France
* Paris Descartes, Paris, France

ABSTRACT

Depth estimation from a single monocular image has reached great performances thanks to recent works based on deep networks. However, as various choices of losses, architectures and experimental conditions are proposed in the literature, it is difficult to establish their respective influence on the performances. In this paper we propose an in-depth study of various losses and experimental conditions for depth regression, on NYUv2 dataset. From this study we propose a new network for depth estimation combining an encoder-decoder architecture with an adversarial loss. This network reaches top ones state of the art on NUYv2 dataset while being simpler to train in a single phase.

***Index Terms*—** Depth estimation, deep learning, loss function.

1. INTRODUCTION

Depth estimation is a major problem in computer vision with several applications in human machine interaction, augmented reality and robotics. Standard approaches were based on stereoscopic vision, structured light, or Structure from Motion (SfM). However, these techniques often have limitations that depend on the environment (*e.g.* sun, texture) or that require several views of the scene. Thanks to easily generated Red Green Blue Depth (RGB-D) data, several approaches based on deep learning have been proposed in recent years, starting from [1]. They exploit geometrical aspects of a scene from a single point of view (a single image) to estimate the 3D structure with the use of convolutional neural networks (CNNs) [2, 3, 4, 5].

These networks usually optimize a regression on the reference depth map. The first main challenge faced by the aforementioned papers is defining an appropriate loss function for depth regression. $\mathcal{L}_2$ has often been a popular choice for this task, but a custom loss [1] and, more recently, an adversarial loss [6] have also been adopted with success. The second challenge concerns the network architecture, which usually followed the advances proposed every year in this flourishing field: VGG16 [2, 3], fully convolutional encoder-decoders [7], and recently Residual Networks (ResNet) [8]. Thus, the relationship between networks and objective functions is intricate, and their respective influences are difficult to distinguish. In this paper, we investigate how particular choices of loss functions and experimental conditions affect depth prediction performances.

Concretely, we lead an in-depth study of the various losses adopted until now, also analyzing standard regression losses. We highlight the main contributions of this paper as follows:

- We show that on **small training datasets**, the simple $\mathcal{L}_1$ loss usually performs better than previously proposed losses alongside with scale-invariant loss;
- We also show that with **large training data**, we can benefit from an adversarial loss to get even finer details in depth estimates, possibly because there is no mode collapse [9] in such cases.
- We show that our approach, which consists of an encoder-decoder network with dense blocks and skip connections and an adversarial loss, is among the top ones of the state of the art on NYUv2 [10] while being simpler to train than previous works.

The paper is organized as follows. Section 2 summarizes last contributions to deep depth prediction field. The proposed method is presented in details in section 3 and section 4 presents our experiments and analysis of results.

2. RELATED WORK

Several machine learning techniques have been proposed to solve the problem of depth estimation on a single image. One of the first solutions was the Make3D approach of Saxena *et al.* [12, 13], which formulates the problem as the capture of properties of the image (e.g., planarity, co-linearity) combined with regression on the true depth using a Markov Random Field (MRF). Recently, most new methods are based on deep convolutional neural networks (DCNN). This was made possible by the release of a large-scale RGB-D dataset: NYUv2 [10]. The first network architecture was proposed by Eigen *et al.* [1] who adopted a multi-scale DCNN. They carefully designed a scale-invariant loss (cf. $\mathcal{L}_{eigen}$ in Table 1) to encourage neighbor pixels to have similar depth values. Wang *et al.* [4] extended this work by exploring joint depth

Loss		Equation
Mean absolute	$\mathcal{L}_1$	$\frac{1}{N}\sum_i^N \lvert l_i \rvert$
Mean square	$\mathcal{L}_2$	$\frac{1}{N}\sum_i^N (l_i)^2$
Scale-invariant loss [1]	$\mathcal{L}_{eigen}$	$\frac{1}{N}\sum_i^N d_i^2 - \frac{\lambda}{N^2}(\sum_i^N d_i)^2$
Scale-invariant loss with gradients [2]	$\mathcal{L}_{eigengrad}$	$\frac{1}{N}\sum_i^N d_i^2 - \frac{\lambda}{2N^2}(\sum_i^N d_i)^2 + \frac{1}{N}\sum_i^N [(\nabla_x di)^2 + (\nabla_y di)^2]$
BerHu [8]	$\mathcal{L}_{berhu}$	$\begin{cases} \mathcal{L}_1(l_i) & \mathcal{L}_1(l_i) \leqslant c, \\ \frac{\mathcal{L}_2(l_i)+c^2}{2c} & else. \end{cases}$
Huber [8]	$\mathcal{L}_{huber}$	$\begin{cases} \mathcal{L}_1(l_i) & \mathcal{L}_1(l_i) \geqslant c, \\ \frac{\mathcal{L}_2(l_i)+c^2}{2c} & else. \end{cases}$
Least Squared Adversarial [9, 11]	$\mathcal{L}_{gan}$	$\frac{1}{2}\mathbb{E}_{x,y\sim p_{data}(x,y)}[(D(x,y) - 1)^2] + \frac{1}{2}\mathbb{E}_{x\sim p_{data}(x)}[(D(x,G(x)) - C)^2] + \lambda\mathcal{L}_{L1}(G(x)))$

Table 1: List of common losses for regression. Let y_i and $\hat{y}_i$ be the ground truth and the estimated distance in meters, $l_i = y_i - \hat{y}_i$, $d_i = log(y_i) - log(\hat{y}_i)$, G, the generator network, D, the discriminator network and x, the input image.

and semantic prediction with a hierarchical Conditional Random Field (HCRF) and, in [2], Eigen *et al.* included first order gradients in the loss ($L_{eigengrad}$) to enforce close local structure on depth prediction.

However, most subsequent works which base training on pixel-wise regression, simply used standard regression losses like mean absolute ($\mathcal{L}_1$) and mean square ($\mathcal{L}_2$) to train their networks [8, 14, 15]. Most contributions lie in the network architectures and the use of Condition Random Fields (CRF). Laina *et al.* [8] claim empirical improvements due to the loss design using the $\mathcal{L}_{berhu}$, instead of $\mathcal{L}_2$ alone, but their method also includes a new network and a new component, the up-projection blocks. Comparison between losses is performed only between $\mathcal{L}_{berhu}$ and $\mathcal{L}_2$. This work was extended in [15] with adoption of an $\mathcal{L}_1$ loss.

A recent method for regression is performed by Generative Adversarial Networks (GANs). They were first introduced by Goodfellow *et al.* [16] to produce realistic images from noise vectors and extended in [11] to condition the generated outputs to an input image (cGAN). They work by defining an adversarial loss which is modeled by a network that classifies the likeliness of the output, often regularized with a $\mathcal{L}_1$ term. Jung *et al.* [6] successfully used this idea of the adversarial loss to perform depth prediction with a two-phase training strategy: network is first trained with a $\mathcal{L}_1$ and posteriorly fine-tuned with the adversarial loss.

Finally, Kendall and Gal [17] proposed a Bayesian network based on [18, 19, 20] combined to a novel regression function that captures the uncertainty of the data (noisy observations) to improve learning.

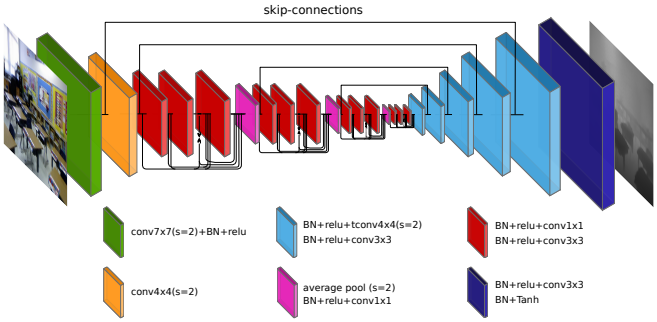


Fig. 1: D3-Net architecture. The encoder part corresponds to a modified version of DenseNet-121, where we replaced a max-pooling by a 4x4 convolution with stride=2 (yellow block).

All the aforementioned works made use of the latest state of the art networks to improve performance while adopting new losses. However, none of them performed a complete comparison between all the already proposed cost functions. In this work, we conduct a comparison of standard and custom losses including the long-discarded $\mathcal{L}_{eigen}$. Also, we bring a new insight to the use of the adversarial loss which requires a large amount of data to be effective. The network, which eventually get close to best performances on NYUv2, is also much simpler to train with respect to [14, 6] as it can be performed end-to-end.

3. DEEP DEPTH PREDICTION NETWORK

D3-Net architecture. To conduct the experiments, we propose an encoder-decoder architecture, referred to as D3-Net, illustrated in Figure 1, which is based on DenseNet-121 [19] for the contractive part, where we replaced a max-pooling by a 4x4 convolution with stride=2 (yellow block). Here, dense blocks, DBxs, $x \in [1, 2, 3, 4]$, contain 6, 12, 24 and 16 convolutions respectively. The decoder comprises blocks of 4x4 transposed convolutions with stride 2 and 3x3 convolutions with stride 1 to upsample feature maps to a higher resolution. The encoder and decoder parts are connected through skip connections like proposed by [21] to improve context-aware learning. In contrast to precedent architectures [14, 6], our network can be trained in a single phase and does not require any additional analytical model like CRFs [4, 14].

Patch-GAN. We modify the conditional patch GAN previously proposed in [11] to the task of depth estimation. The discriminator network is designed to measure and classify if an input depth map is true or false. True maps correspond to the ground truth depths and false maps correspond to generated depths. This network is trained to replace handcrafted loss functions as it tries to find a implicit definition of the loss function by learning a metric in the image space. However, to smooth GAN predictions and guide training, we add an $\mathcal{L}_1$ term to the output of D3-Net. The patch structure allows the discriminator to penalize the predictions per patches instead

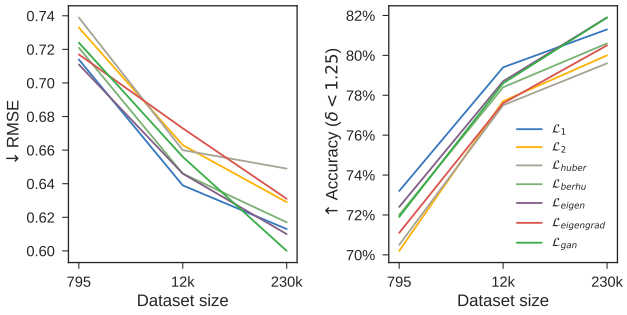


Fig. 2: Performance evolution for different dataset sizes and different losses using D3-Net architecture.

of penalizing the whole image, which leads to results with finer details. The output of the patch-discriminator is 78x62 for an input image of 320x256.

4. EXPERIMENTS

To compare the performances on depth estimation, we adopt standard error measurements proposed in [1, 22] and also a standard benchmark dataset for deep depth prediction: NYUv2. NYU-Depth V2 (NYUv2) dataset [10] has approximately 230k pairs of indoor images from 249 scenes for training and 215 scenes for testing. NYUv2 also contains a smallest dataset with 1449 pairs of aligned RGB and depth images, of which 795 pairs are used for training and 654 pairs for testing. Original frames from Microsoft Kinect output are 640x480. Pairs of images from the RGB and Depth sensors are posteriorly aligned, cropped and processed to fill-in invalid depth values. Final resolution is 561x427.

In the first experiment, we observe, for all regression losses in Table 1, the RMS error and accuracy variation according to different loads from the original dataset. We also study the convergence speed of the network to improve results. Note that to conduct direct comparisons, we carefully perform all training processes keeping network parameters without any change. Finally, to generalize our conclusions, we guide a second experiment where the front-end network of D3-Net, originally DenseNet-121, is replaced by ResNet-50, already adopted in [8, 14, 15]. We then study the variations of three error metrics for the different losses when changing the architecture.

Quantitative performance comparison. Figure 2 shows the evolution of the network performance with different losses when trained with different sizes of dataset. We adopt three different splits with the 795 pairs from the small NYUv2 dataset, 12k pairs from equally spaced samples of the complete dataset and 230k pairs of images from the whole dataset.

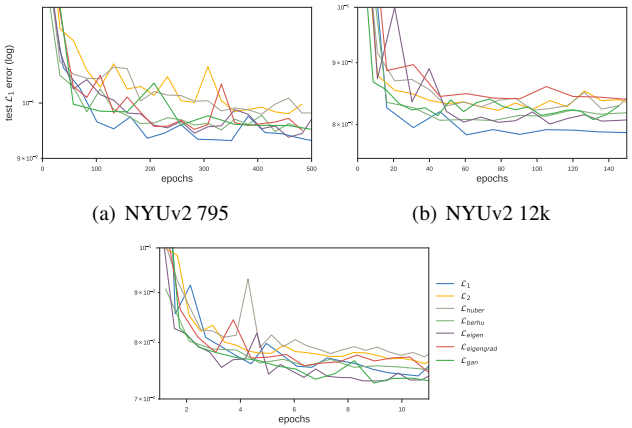
As one can expect, more data leads to better results in all cases. However, losses evolves differently from one split to another. In general terms, $\mathcal{L}_1$ and $\mathcal{L}_{eigen}$ present the best performances for different sizes of the dataset. On the other hand, $\mathcal{L}_{gan}$ becomes highly efficient when trained with a great amount of data. GANs have a well known instability (mode-collapse [9]) that, in our case, can be circumvented with more data.

From Figure 3, $\mathcal{L}_1$ and $\mathcal{L}_{eigen}$ also appear to converge more effectively than the other losses and then obtain better predictions faster. This remains true for the two smaller splits, but when training the model with 230k, we can notice the GAN model and $\mathcal{L}_{eigen}$ outperform other error functions.

Qualitative performance comparison. For better comparison between the models, we also provide visualization of predicted depth maps in Figure 4 for models trained on the complete NYUv2 dataset. In general, we notice that $\mathcal{L}_{huber}$ and $\mathcal{L}_2$ tend to smooth predictions. Even though $\mathcal{L}_{berhu}$ benefits from $\mathcal{L}_1$ for small errors, $\mathcal{L}_2$ factor seems to degrade estimations as well. It is important to notice that standard $\mathcal{L}_2$ encourages residuals where error is small, but $\mathcal{L}_1$ can encourage sparse solutions where error is zero. $\mathcal{L}_{berhu}$ proposes to take advantage of $\mathcal{L}_1$ for very small errors and use $\mathcal{L}_2$ otherwise. From the presented quantitative and qualitative experiments the squared term seems to favor smooth predictions when adopting $\mathcal{L}_{berhu}$ as well as $\mathcal{L}_{huber}$ and $\mathcal{L}_2$. On the other hand, $\mathcal{L}_{gan}$, $\mathcal{L}_{eigen}$ and $\mathcal{L}_1$ present nice visual predictions confirming previous quantitative results. The patch-GAN approach can lead the model to capture high-frequency details (*e.g.*, contours, small objects).

These characteristics can be clearly observed for example in the first row, where the contours of the different chairs in the back are well predicted when compared to $\mathcal{L}_{berhu}$ and $\mathcal{L}_{huber}$, for example, that almost ignore them. Other very fine details can be seen in the $\mathcal{L}_{gan}$ predictions of the second row for the shelves and the television.

Different front-end architectures. In order to general-



(a) NYUv2 795 (b) NYUv2 12k

(c) NYUv2 230k

Fig. 3: Comparison of the convergence speed between the losses in Table 1 on test data.

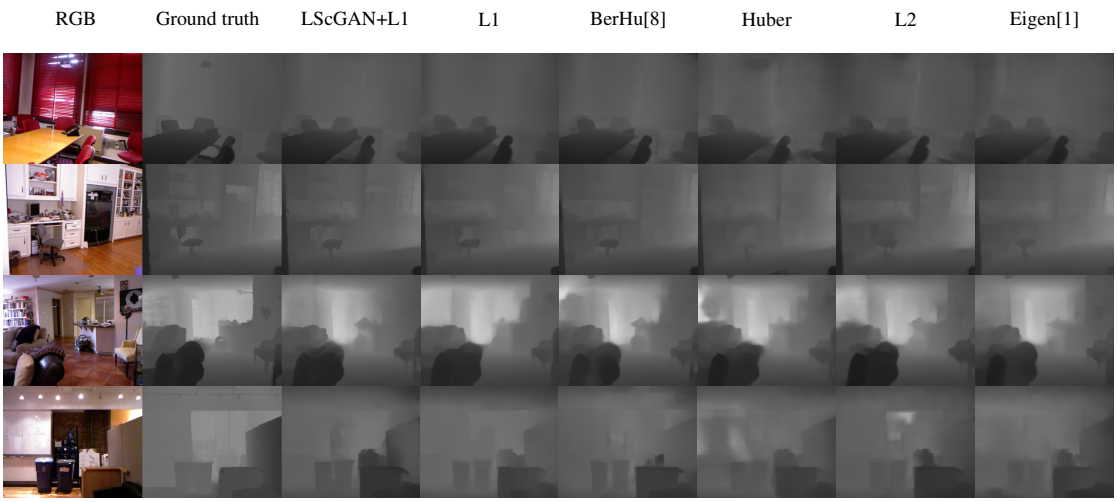


Fig. 4: Qualitative result of D3-Net trained to minimize different regression losses from the literature of depth from monocular images.

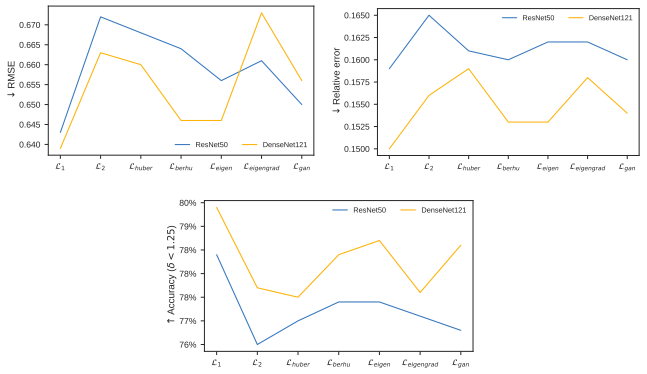


Fig. 5: Performance comparison of regression losses with different front-end architectures.

ize our study, we evaluate the performances of the presented losses with another front-end architecture: ResNet. The main difference with DenseNet is that ResNet learns by optimizing the residual information and DenseNet learns by feeding later layers with feature maps from precedent ones and more importantly, this allows gradients to flow directly to input signal diminishing cases of vanishing gradients. Figure 5 shows on the same graph performance of both front-end networks. We adopt the training split with 12k images to fasten training compared to the whole dataset. Our results show that $\mathcal{L}_1$ and $\mathcal{L}_{eigen}$ show better results for both architectures. Besides, DenseNet encoder presents globally better results than ResNet with the only exception of slightly poorer RMSE.

Comparison with state of the art methods Finally, we show in Table 2 that the proposed D3-Net architecture combined with $\mathcal{L}_{gan}$ and trained with NYUv2 230k reaches the top state of the art methods. Our method using adversarial loss can be trained end-to-end in a single phase, in contrast to [6]. Compared to [18], it does not require the use of a Monte Carlo method to capture the uncertainty of the model and improve performance, like [17].

Methods	Error↓				Accuracy↑		
	rel	log10	rms	rmslog	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Saxena [12]	0.349	-	1.214	-	44.7%	74.5%	89.7%
Eigen [2] (VGG16)	0.158	-	0.641	0.214	76.9%	95.0%	98.8%
Laina [8]	0.127	0.055	0.573	**0.195**	81.1%	95.3%	98.8%
Xu [14]	0.121	0.052	0.586	-	81.1%	95.4%	98.7%
Cao [23]	0.141	0.060	0.540	-	81.9%	96.5%	99.2%
D3-Net	0.135	0.059	0.600	*0.199*	*81.9%*	95.7%	98.7%
Jung[6]	*0.134*	-	*0.527*	-	**82.2%**	*97.1%*	*99.3%*
Kendall and Gal [17]	**0.110**	**0.045**	**0.506**	-	81.7%	95.9%	98.9%

Table 2: Performance metrics obtained by state of the art methods of deep depth estimation with NYUv2 dataset. Results extracted from original papers. Our best result consists on the D3-Net trained with $\mathcal{L}_{gan}$ with 230k pairs of images.

5. CONCLUSION

In this paper, we have presented a study of the influence of regression losses and experimental conditions on depth estimation using deep learning. Several losses from the literature as well as standard losses have been considered. Performance tests have been conducted on NYUV2 datasets with various sizes, and two different encoder-decoder architectures. We have shown that on small datasets, $\mathcal{L}_1$ and $\mathcal{L}_{eigen}$ losses produce the best performances and when the size of the dataset increases, the performance benefits from the use of adversarial loss. Finally, based on this study we have proposed a network combining a simple encoder-decoder architecture with dense blocks and skip connections and an adversarial loss. This network reaches the top ones results on the NYUv2 dataset while being simpler to train than previous works such as [14, 17].

6. REFERENCES

[1] David Eigen, Christian Puhrsch, and Rob Fergus, "Depth map prediction from a single image using a multi-scale deep network," *NIPS*, 2014.

[2] David Eigen and Rob Fergus, "Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-Scale Convolutional Architecture," *ICCV*, 2015.

[3] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian D. Reid, "Learning Depth from Single Monocular Images Using Deep Convolutional Neural Fields," *TPAMI*, 2015.

[4] Peng Wang, Xiaohui Shen, Zhe Lin, Scott Cohen, Brian Price, and Alan L Yuille, "Towards unified depth and semantic prediction from a single image," in *CVPR*, 2015.

[5] Ayan Chakrabarti, Jingyu Shao, and Greg Shakhnarovich, "Depth from a single image by harmonizing overcomplete local network predictions," *NIPS*, 2016.

[6] Hyungjoo Jung, Youngjung Kim1, Dongbo Min, Changjae Oh, and Kwanghoon Sohn, "Depth prediction from a single image with conditional adversarial networks," in *ICIP*, 2017.

[7] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox, "Demon: Depth and motion network for learning monocular stereo," *arXiv preprint arXiv:1612.02401*, 2016.

[8] Iro Laina, Christian Rupprecht, Vasileios Belagiannis, Federico Tombari, and Nassir Navab, "Deeper depth prediction with fully convolutional residual networks," in *3D Vision (3DV), 2016 Fourth International Conference on*. IEEE, 2016, pp. 239–248.

[9] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley, "Least squares generative adversarial networks," *arXiv preprint ArXiv:1611.04076*, 2016.

[10] Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus, "Indoor segmentation and support inference from rgbd images," in *ECCV*, 2012.

[11] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint arXiv:1611.07004*, 2016.

[12] Ashutosh Saxena, Sung H Chung, and Andrew Y Ng, "Learning Depth from Single Monocular Images," *NIPS*, 2006.

[13] Ashutosh Saxena, Min Sun, and Andrew Y Ng, "Make3d: Learning 3d scene structure from a single still image," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 5, pp. 824–840, 2009.

[14] Dan Xu, Elisa Ricci, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe, "Multi-scale continuous crfs as sequential deep networks for monocular depth estimation," *arXiv preprint arXiv:1704.02157*, 2017.

[15] Fangchang Ma and Sertac Karaman, "Sparse-to-dense: Depth prediction from sparse depth samples and a single image," *ICRA*, 2018.

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, "Generative adversarial nets," *NIPS*, 2014.

[17] Alex Kendall and Yarin Gal, "What uncertainties do we need in bayesian deep learning for computer vision?," *arXiv preprint arXiv:1703.04977*, 2017.

[18] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," *arXiv preprint arXiv:1511.02680*, 2015.

[19] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q Weinberger, "Densely connected convolutional networks," in *CVPR*, 2017.

[20] Simon Jégou, Michal Drozdzal, David Vazquez, Adriana Romero, and Yoshua Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE, 2017, pp. 1175–1183.

[21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," *MICCAI*, 2015.

[22] Fayao Liu, Chunhua Shen, and Guosheng Lin, "Deep Convolutional Neural Fields for Depth Estimation from a Single Image," *CVPR*, 2015.

[23] Yuanzhouhan Cao, Zifeng Wu, and Chunhua Shen, "Estimating depth from monocular images as classification using deep fully convolutional residual networks," *IEEE Transactions on Circuits and Systems for Video Technology*, 2017.