Mini V-Net: Depth Estimation from Single Indoor-Outdoor Images using Strided-CNN

Ahmed J. Afifi[1][a], Olaf Hellwich[1] and Toufique Ahmed Soomro[2][b]

[1]*Computer Vision and Remote Sensing, Technische Universität Berlin, Berlin, Germany*

[2]*Electronic Engineering Department Quaid-e-Awam University of Engineering and Technology, Larkana Campus, Pakistan*

Keywords: Convolutional Neural Networks, CNN, Depth Estimation, Single View.

Abstract: Depth estimation plays a vital role in many computer vision tasks including scene understanding and reconstruction. However, it is an ill-posed problem when it comes to estimating the depth from a single view due to the ambiguity and the lack of cues and prior knowledge. Proposed solutions so far estimate blurry depth images with low resolutions. Recently, Convolutional Neural Network (CNN) has been applied successfully to solve different computer vision tasks such as classification, detection, and segmentation. In this paper, we present a simple fully-convolutional encoder-decoder CNN for estimating depth images from a single RGB image with the same image resolution. For robustness, we leverage a non-convex loss function which is robust to the outliers to optimize the network. Our results show that a light simple model trained using a robust loss function outperforms or achieves comparable results with other methods quantitatively and qualitatively and produces better depth information of the scenes with sharper objects' boundaries. Our model predicts the depth information in one shot with the same input resolution and without any further post-processing steps.

1 INTRODUCTION

Depth estimation from a single image, i.e., estimating the distance of each pixel in the image to the camera, is an ill-posed problem with the absence of the environmental assumptions. Depth information, besides the RGB images, is an important component for understanding the 3D geometry of a scene and a richer representation for the objects. It has an influence on many applications from semantic segmentation (Ladicky et al., 2014) and labeling, scenes modeling (Hoiem et al., 2005), augmented reality (AR), robotics (Hadsell et al., 2009), to autonomous driving. Normally, the RGB-D data are collected using depth sensors either from outdoor scenes or indoor scenes. These data are used to investigate and solve the depth estimation problem either from single or multiple views. For multi-view systems, local correspondence is found and utilized to estimate the depth information. Structure-from-Motion (SfM) (Roberts et al., 2011) is a promising method that uses multiple images to estimate the camera poses, the local correspondences, and the depth. For single view systems, estimating the depth information from a single image is inherently ambiguous as the image scene could correspond to different scenes and it is difficult to map the color information from the RGB image into depth values. Prior information is needed to estimate the depth information, and solving this problem with plausible accuracy helps in improving the outcomes of many computer vision tasks, such as recognition (Ren et al., 2012) and reconstruction (Silberman et al., 2012).

Comparing to multi-view depth estimation, few researchers have focused on the problem of single-view depth estimation compared to the stereo images' systems. For stereo images, the correspondence between the images can be extracted accurately and then the depth information can be recovered from the correspondences (Roberts et al., 2011). As humans, it is interstingly that we can solve such an ill-posed problem by exploiting our knowledge. However, automatic estimating the depth from a single view needs prior knowledge and cues of the scene which can be restricted by the scene environment such as parallel lines for the indoor scenes, the sky and the ground for the outdoor scenes or assuming a box model for room scenes. Also, object position and size play an important role in depth estimation from a single

[a] https://orcid.org/0000-0001-6782-6753
[b] https://orcid.org/0000-0002-8560-0026

Afifi, A., Hellwich, O. and Soomro, T.
Mini V-Net: Depth Estimation from Single Indoor-Outdoor Images using Strided-CNN.
DOI: 10.5220/0009356102050214
In *Proceedings of the 15th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2020) - Volume 5: VISAPP*, pages 205-214
ISBN: 978-989-758-402-2; ISSN: 2184-4321

view. These assumptions and cues restrict the applications and cannot be generalized for further data or even new tasks. Other methods depend on retrieving similar models that try to align them with the input scene to infer depth information. In recent years, researchers have incorporated different sources of information such as user annotations and labeling to perform depth estimation. Still, the mentioned methods depend on hand-crafted features to solve the problem of depth estimation from a single image.

Recently, Convolution Neural Networks (CNNs) have shown a breakthrough performance in solving computer vision tasks. This success led many researchers to apply deep learning to solve different computer vision tasks. Starting from AlexNet (Krizhevsky et al., 2012) as a base network for object classification, many deeper networks have been proposed to solve different computer vision tasks such as VGGNet (Simonyan and Zisserman, 2014), GoogLeNet (Szegedy et al., 2015), and deep ResNet (He et al., 2016). Moreover, CNNs are employed to learn implicit relations between RGB images and depth, such as object detection and localization, scene segmentation, depth estimation, and medical image segmentation (Soomro et al., 2019a) (Soomro et al., 2019b) . In general, deep learning outperforms the traditional hand-crafted features methods (e.g. SIFT (Lowe, 2004), HOG (Dalal and Triggs, 2005), and FV (Perronnin et al., 2010)) in solving these problems as the CNNs learned useful features directly from the images.

In this paper, we propose a CNN model to solve the problem of depth estimation from a single RGB image. Our model is a fully convolutional encoder-decoder model, a light version of V-Net (Milletari et al., 2016), with skip connections between the encoder part and the decoder part to generate the depth image. In the encoder, the pooling layers were replaced with strided convolutional layers (Springenberg et al., 2014). The decoder part is mirrored from the encoder with additional layers; the upsampling layers (deconvolutional layers) and the concatenation layers. The encoder and the decoder are connected via the skip connections. The concatenation layers, which perform as fusion layers, fuse features from the encoder part with the features from the upsampling layers. Here, the output from each block in the encoder part is fused with the corresponding upsampled output in the decoder part. In this way, the fine-grained features are fused with the decoder features, where the decoder features lose some information due to the upsampling operation. As a consequence, this step improves the quality of the predicted depth image. Also, the generated output has the same resolution as the input image so, there is no loss in the output resolution. Lastly, the proposed model was trained by optimizing using Tukey's biweight loss function which is a non-convex loss function that is robust in regression tasks.

We test the proposed model on a dataset and evaluate the performance quantitatively and qualitatively. The results show that our model performs better than other proposed methods.

2 RELATED WORK

The first work on depth estimation was originally based on stereo vision, where pairs of images of the same scene were used for 3D shape reconstruction. Most approaches for single-view depth estimation depend on different shooting conditions; such as Shape-from-Shading (SfS) (Zhang et al., 1999), and Shape-from-Defocus (SfD) (Suwajanakorn et al., 2015). While depth estimation from a single view is a challenging task, many researchers proposed different methods to solve it. Following, we revise the related work of single-view depth estimation using both classical and deep learning methods.

Early work in depth estimation was using hand-crafted features to predict the depth from a single image. Saxena et al. (Saxena et al., 2009) extracted local and global features from the image to infer the depth information using a Markov Random Field (MRF). Superpixels were introduced to enforce the consistency of the neighboring regions. Liu et al. (Liu et al., 2010) predicted the depth information from semantic segmented labels to simplify the problem and achieve improved results using a MRF model. Other non-parametric methods, include SIFT (Lowe, 2004), HOG (Dalal and Triggs, 2005), performed features matching between the input image and images in a dataset to find the most similar image. The depth information of the matched image is then retrieved to infer the final depth information of the input image. Liu et al. (Liu et al., 2014) imposed that similar regions in images have similar depth cues, so they defined the optimization problem as a Conditional Random Field (CRF) to infer the depth of different superpixels.

In the deep learning era (starting from the success of applying CNNs in classification tasks (Krizhevsky et al., 2012)), researchers have applied deep learning in solving the depth estimation problem. Eigen et al. (Eigen et al., 2014) proposed a CNN to predict the depth directly from a single image. The model was multi-stage where the coarse depth is predicted from the first stage of the network and was combined with the output of the first convolutional layer in the second stage to infer the final depth map. The authors extended their model to estimate depth, normals, and semantic labels (Eigen and Fergus, 2015). Afifi and Hellwich (Afifi and Hellwich, 2016) proposed a fully CNN model to estimate the depth of objects from a single image. The model was optimized using a non-convex loss function and the L2 norm. The disadvantage of the above-mentioned models is that the output resolution is smaller than the input image. This resulted in a blurry output and the generated images lose many details describing the objects. In (Liu et al., 2015), the authors proposed a CNN to infer the depth map from a single image and used CRFs to model the relations between the neighboring superpixels. Cao et al. (Cao et al., 2018) formulated the depth estimation as a pixel-wise classification task using ResNet (He et al., 2016). The continuous depth values were discretized into multiple categories depends on the depth range. The network was trained to classify each pixel into a depth range. To improve the output depth map, fully connected CRF was applied as a post-processing step to enforce local smoothness interactions.

Our work utilizes a different approach than the previous ones. While the previous works focused on the final output, we exploit from the intermediate features and fuse them with other features during the network layers and improve the final output. Depth information is used in many applications such as object alignment, object detection, and 3D scene and object reconstruction. Our model is an encoder-decoder model, where we utilize the features from the encoder part to improve the final generated depth. It is a single-stage model that estimates depth information as opposed to the multi-stage models that use multiple networks for estimating the depth (Eigen and Fergus, 2015). Importantly, the output depth image has a similar resolution of the input image. This gives more details regarding the objects and as a consequence, the depth images are not blurry as in the previously mentioned work. Finally, in our approach, a post-processing step to enhance the results is not required like in (Liu et al., 2010).

3 PROPOSED ARCHITECTURE AND LOSS FUNCTION

In this section, we present in detail the proposed model of depth estimation from a single RGB image, then we present and discuss the non-convex loss function that has been used for task optimization. In general, our method treats the depth estimation problem as a regression task where we estimate the depth value for each pixel in the image.

3.1 Proposed Architecture

When designing the CNN to solve a problem, the nature of the problem, either a classification or a regression problem, plays an important role in selecting the layers and the loss function for the optimization. For example, AlexNet (Krizhevsky et al., 2012) consists of consecutive convolutional layers, each followed by Rectified Linear Unit (ReLU), and pooling layers to decrease the features resolution and the computation cost. In the end, fully connected layers (FC) are used for classification. The output layer depends on the nature of the task to be solved. This arrangement of layers performs both linear and non-linear operations to extract features that are subsequently used to solve the problem.

Our model is inspired by V-Net, a 3D-CNN proposed for medical image segmentation (Milletari et al., 2016). The proposed CNN model is an encoder-decoder model for 2D single-view depth estimation. The model is trained end-to-end from scratch and is a fully convolutional model in the encoder and decoder as shown in Fig. 1.

The encoder comprises three consecutive fully convolutional blocks with feature sizes of 16, 32, and 64, respectively. For the first two blocks, we use two convolutional layers, and in the third block, we use three convolutional layers. The convolutional layers have a kernel of size 3×3 and a stride of 1 and are followed by leaky-ReLU (Xu et al., 2015) as an activation function. We use strided convolutional layers instead of the pooling layers as the later ones are mostly used in CNN to decrease the features' size. In (Springenberg et al., 2014), the authors proved that max-pooling layers can simply be replaced by convolutional layers with increased stride. The advantage of using strided convolutional layers is that they can be easily reversed, trained, and tuned rather than fixing them to max or average operations. In our model, the strided convolutional layers are of 2×2 kernel size with a stride of 2. This decreases the size of the feature maps to a half.

The decoder has the same structure of the encoder, but with some additional layers that reconstruct the image again and generates the depth image with the same size as that of the input image. The sizes of the features for each convolutional block in the decoder are 64, 32, and 16, respectively. In particular, we add the upsampling (upconvolution) layers of 3×3 kernel size in the decoder part to reconstruct the features and generate the final depth image with the same resolution as that of the input image. LReLU is used as an activation function in each block. To generate a better depth image with more details, we concatenate the

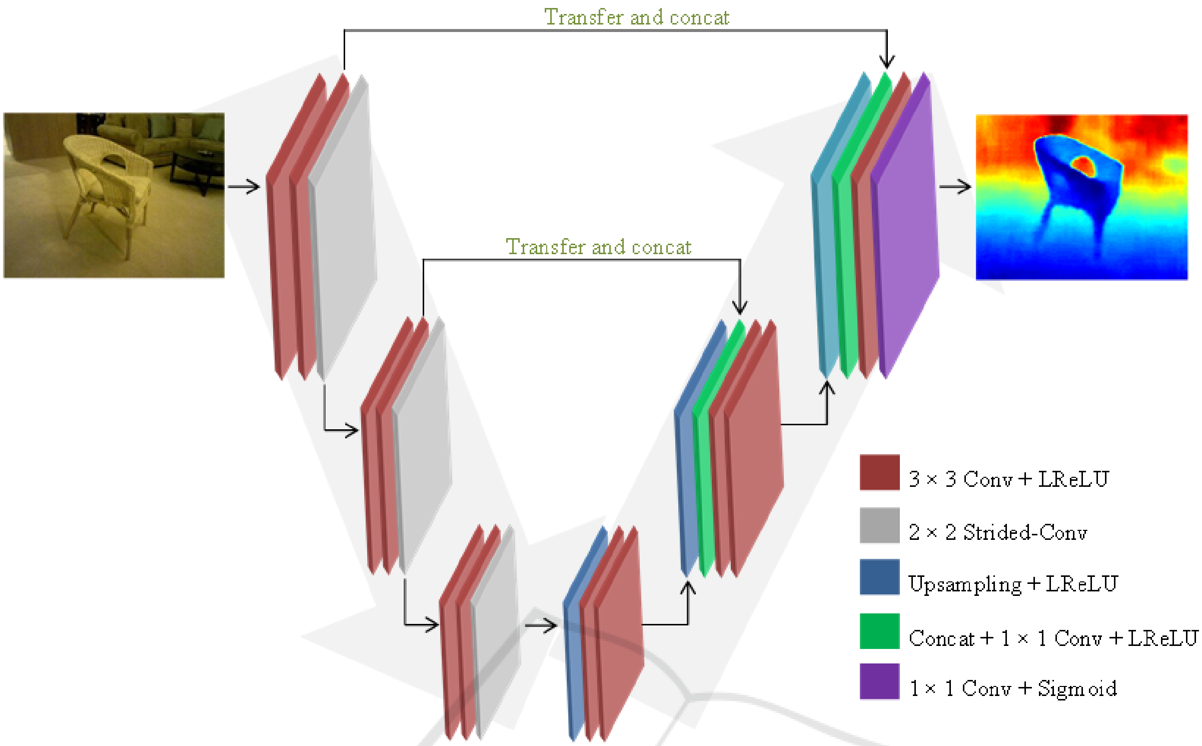

Figure 1: The proposed mini strided V-Net architecture. Pooling layers are replaced by strided-convolutional layers to generate fine-grained output of same resolution as that of the input. Depth images are visualized with log scale.

output of some layers from the encoder part with the corresponding output of the upsampling layers from the decoder part as shown in Fig. 1 (the concatenation links between the encoder part and the decoder part). After the concatenation, we apply a convolutional operation of 1×1 kernel size to fuse the concatenated feature maps. We noticed that the concatenation layers (or as they are called the skip connections) add more details, and as a consequence, the objects in the output images have sharper edges and hence are less blurry. The depth image is generated using a sigmoid layer as the output range is between the interval of (0,1).

Generally speaking, the encoder-decoder models have been introduced into many computer vision tasks such as semantic segmentation, image reconstruction, and optical flow estimation. They have significantly outperformed other models in solving the same tasks. The encoder-decoder models have shown impressive success in solving single-view depth estimation for scenes in supervised and unsupervised training modes. In the results section, we will compare the proposed model with other models and show that the proposed encoder-decoder model outperforms other models and can generate better depth images with more details.

3.2 Loss Function

Selecting a suitable loss function plays a critical step in training a CNN. In our case, the loss function measures the error between the generated depth image and the ground-truth depth image to optimize and update the model weights. It should fulfill some constraints regarding the task and the nature of the training dataset. Our depth estimation problem is considered as a regression task and a straightforward loss function like L2 norm can be used to compute the error between the estimated values $\hat{y}$ and the ground-truth y (Liu et al., 2017).

For depth estimation, L2 norm is not robust to outliers (the large error calculated between the predicted depth and the ground-truth) (Liu et al., 2017). Optimizing the model using L2 norm biases the training process towards the outliers because small errors (differences between the ground-truth and the predicted depth values) have little influence on the CNN weight modifications, while the large errors (outliers) incur a large penalty.

To overcome this issue, we propose to use a non-convex loss function that is robust in regression tasks namely, Tukey's biweight loss function (Eq. 2) (Black and Rangarajan, 1996). The advantage of using this loss function is that the small residual values (the dif-

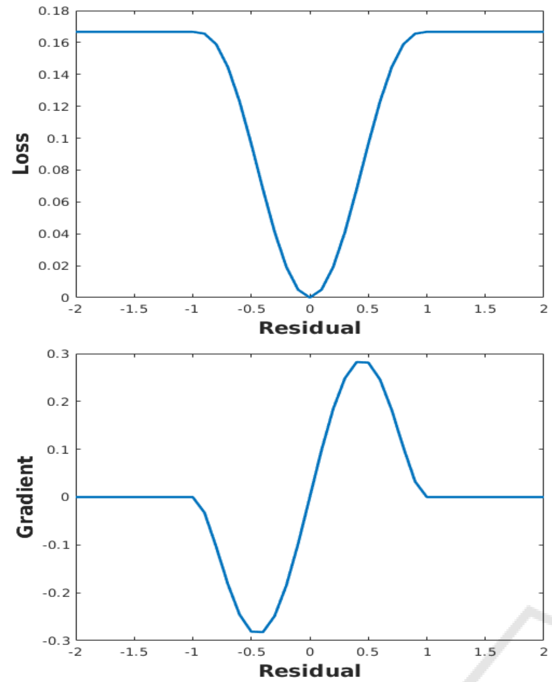

Figure 2: Tukey's biweight loss function (top) and its first-order derivative (bottom) where c = 1.

ference between the predicted depth and the ground-truth depth) influence the training process and robust to the outliers. During the training process, the loss function suppresses the influence of the outliers and sets the magnitude of the outlier gradients close to zero.

Formally, the difference between the ground-truth depth y and the estimated depth value ŷ (i.e. the residual r) is calculated as:

$$r = \hat{y} - y \tag{1}$$

Given the residual r (Eq. 1), Tukey's biweight loss function is defined by:

$$\rho(r) = \begin{cases} \frac{c^2}{6}\left[1-\left(1-\frac{r^2}{c^2}\right)^3\right] & \text{if } |r| \leq c \\ \frac{c^2}{6} & \text{if } |r| > c \end{cases} \tag{2}$$

The first-order derivative of Tukey's biweight loss with respect to r is defined as:

$$\acute{\rho}(r) = \begin{cases} r\left(1-\frac{r^2}{c^2}\right)^2 & \text{if } |r| \leq c \\ 0 & \text{if } |r| > c \end{cases} \tag{3}$$

To correctly apply this loss function, the residual (r) should be scaled (r should be drawn from distribution with unit variance). Median absolute deviation (MAD) is selected to measure the variability in the training data to scale the residuals. MAD is defined as:

$$MAD_i = median(|r_i|) \tag{4}$$

MAD_i scales the residuals to obtain the unit variance. The scaled residual (r_i^{MAD}) is calculated as:

$$r_i^{MAD} = \frac{y_i - \hat{y}_i}{1.4826 \times MAD_i} \tag{5}$$

The scaled r_i^{MAD} in Eq. 5 is used by the loss function Eq. 2 with $c = 4.6851$. An advantage of using Tukey's biweight function as a loss function is that it is differentiable and the training process converges better when the depth values are represented in a log scale. Fig. 2 shows Tukey's biweight loss and its first derivative when $c = 1$.

3.3 Dataset & Implementation Details

We use MatConvNet (Vedaldi and Lenc, 2015), a MATLAB toolbox implementing CNNs for computer vision applications, to train and evaluate our proposed model. The weights of the layers are initialized using Xavier initialization method (Glorot and Bengio, 2010). The model was trained from scratch using backpropagation. Stochastic Gradient Descent (SGD) was used to optimize the network with the following settings: the momentum was set to 0.9 and the weight decay was set to 10^{-5}. The learning rate was initialized to 10^{-3} and was divided by 10 when the validation error didn't change. The training process was repeated until the validation accuracy stopped increasing.

The proposed model was trained on real images. The dataset includes RGB images and their corresponding depth image of real objects. A Large Dataset of Object Scans (Choi et al., 2016) is a publicly available dataset that contains more than tens of thousands of 3D scans of different real objects captured at a resolution of 640×480. We collected different scenes from the chair class. The collected set was split into a training set and a testing set with a respective distribution of 80% and 20%, respectively. The main model was trained using Tukey's biweight loss function. We selected the chair object because the dataset has a massive number of images with diversity in shapes. The network was trained on almost 10 different shapes of chairs. Each chair shape was between 1k and 2k images of different viewpoints and distances in both indoor and outdoor scenes. We applied a pre-processing step on the depth images to fill the missing depth values in the images as shown in Fig. 3.

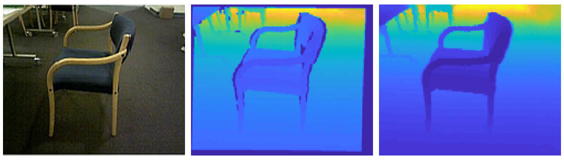

Figure 3: A sample of a training RGB image with a depth image. From left to right: RGB image, original depth image, and preprocessed depth image.

We applied data augmentation on the training set to reduce the overfitting during training and for better generalization performance. Horizontal flipping (mirroring) of images is applied at a probability of 0.5. Vertical flipping on indoor scene images will not help during training. Also, we applied photo-metric transformation, i.e. swapping the color channels of the RGB images, to increase the performance.

4 EXPERIMENTAL RESULTS AND DISCUSSIONS

In this section, we report thorough analyses and results of the proposed model on single-view depth estimation in indoor and outdoor scenes. Moreover, we perform ablation studies to analyze the impact of the loss functions on the results. Finally, we compare and discuss the results and the performance of the proposed model to other state-of-the-art models that used the same dataset for training and testing. The results (qualitatively and quantitatively) show that the proposed model with non-convex loss function performs better than the other models and other loss functions that are used in the regression tasks. For the quantitative evaluation and comparison, the same metrics used in (Afifi and Hellwich, 2016) are computed on our experimental results. The error metrics are defined as:

- Average Absolute Relative Error (rel):

$$\frac{1}{n}\sum_{p}^{n}\frac{|y_p-\hat{y}_p|}{y} \tag{6}$$

- Root Mean Square Error (rms):

$$\sqrt{\frac{1}{n}\sum_{p}^{n}(y_p-\hat{y}_p)^2} \tag{7}$$

- Average log_{10} error (log_{10}):

$$\frac{1}{n}\sum_{p}^{n}|log_{10}(y_p)-log_{10}(\hat{y}_p)| \tag{8}$$

- Threshold accuracy (δ_i): % of $\hat{y}_p$ s.t.

$$max(\frac{y_p}{\hat{y}_p},\frac{\hat{y}_p}{y_p})<\delta_i \tag{9}$$

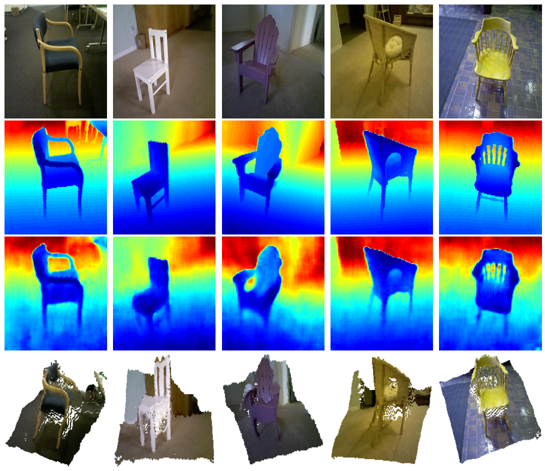

Figure 4: Qualitative results from Large Dataset of Object Scans (chosen from the testing dataset). From top to bottom: input RGB images, the ground-truth images, the predicted depth image, and the reconstructed images using the predicted depth values.

where $\delta_i = 1.25^i$, $i = 1,2,3$

y_p is a pixel in ground-truth depth image y, $\hat{y}_p$ is a pixel in the predicted depth image $\hat{y}$, and n is the total number of pixels for each depth image.

4.1 Mini V-Net Evaluation on Large Dataset of Object Scans

Fig. 4 shows the depth estimation results that are predicted using the proposed model. The predicted depth images have the object details and can be easily distinguished from the background. The fine details of the objects such as the holes in the back of the chairs are predicted accurately and the network succeeds to estimate the objects' parts. Interestingly, the proposed model predicts depth information directly from a single input image without any further post-processing steps. The model is trained end-to-end to estimate the depth of an image in a single shot. On the other hand, other previously described methods improved the predicted depth images through many steps. One method (Eigen and Fergus, 2015) uses a multi-stage model and combines the coarse depth image generated in one stage and the original RGB input image to generate the final depth image. This may introduce noise and reduce global scale depth information. Of note, the output resolution usually is smaller than the input resolution and many details may be missed. Moreover, CRF is used as a post-processing step to generate a more detailed depth image (Liu et al., 2015). As a result, the predicted depth image cannot be estimated directly from the CNN. However, our model differs from these models in be-

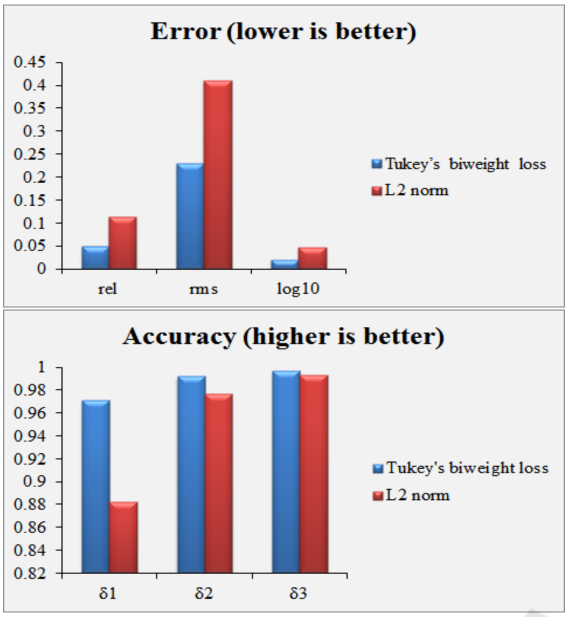

Figure 5: Error (top) and Accuracy (bottom) results of the proposed model using different loss functions.

ing a single stage model whereby no post-processing steps are required to generate the output.

Moreover, we reconstruct the images in 3D using the predicted depth values as shown in Fig. 4 (the last row). It clearly is shown that the depth values were predicted very well and there are different levels of depth values that can distinguish between object parts, the floor, and the background.

4.2 Analysis of Different Loss Functions

We trained the proposed model using two different loss functions; L2 norm and Tukey's biweight loss. In our task, the small difference between the depth values is important because these values highlight the basic features of the object and differentiate it from other objects in the scene. We compared the error and the accuracy of the estimated depth from Tukey's biweight loss with the one from L2 norm loss quantitatively. Fig. 5 (top) shows that the error computed from Tukey's biweight loss model is smaller than the model trained on L2 norm with a large margin. Also, the accuracy (bottom) when using the non-convex loss function is better for training the model in the regression problems.

Fig. 6 elucidates that the model trained using Tukey's biweight loss outperforms the model trained using L2 norm. In detail, the pixels with smaller distances are sensitive to smaller errors. This influences the relative error to be higher and results in larger gradients of Tukey's biweight loss over L2 norm. Consequently, the non-convex loss function is more robust to the outliers and takes care of the small errors between distances such that the output is estimated with finer details compared to L2 norm model. Fig. 6 also shows that the results predicted by the model trained using L2 norm are relatively blurry depth images and the object details are almost missed (the forth row in Fig. 6). Some object parts are fused with the background and the object details are not visible. On the other hand, the depth images generated by Tukey's biweight loss have captured finer details and the object inside the images can be recognized easily from the background. In addition, the network learns to preserve some details related to object shapes such as the holes in the chair's hands and the empty space between the back of the chair and seat.

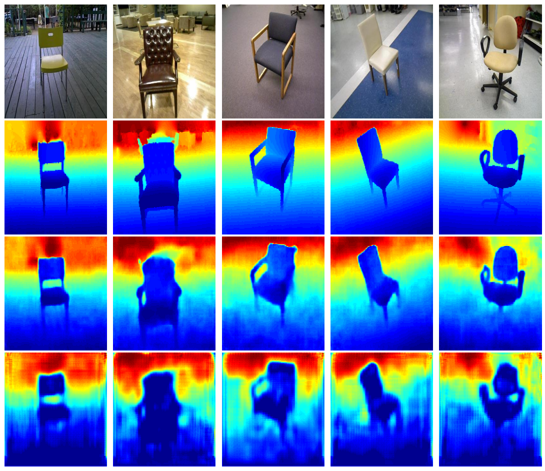

Figure 6: Qualitative comparison results on Large Dataset of Object Scans using different loss functions. From top to bottom: input RGB image, ground-truth, model trained using Tukey's biweight loss, model trained using L2 norm. Depths are shown in log scale and in color (blue is close, red is far).

4.3 Comparison with Other Models

Depth estimation from a single image is an ambiguous task. We compare the proposed model with (Afifi and Hellwich, 2016) which used the same dataset for training but their model is different. Their CNN is a fully convolutional model that simply used convolutional layers and pooling layers and the output was generated from a sigmoid function. Moreover, they trained the fully-CNN with Tukey's biweight loss similar to our approach. Table 1 shows the quantitative comparison with respect to errors and accuracy. Fig. 7 shows the qualitative results predicted by our model along with a comparison to those generated by the model in (Afifi and Hellwich, 2016).

The model in (Afifi and Hellwich, 2016) was

Table 1: Performance Comparison of different methods trained using different loss functions on Large Dataset of Object Scans ($\downarrow$ lower is better, $\uparrow$ higher is better).

Architecture	rel$\downarrow$	rms$\downarrow$	log10$\downarrow$	$\delta_1\uparrow$	$\delta_2\uparrow$	$\delta_3\uparrow$
Models trained using Tukey's biweight loss						
F-CNN (full) (Afifi and Hellwich, 2016)	0.2940	0.9516	0.1264	0.4895	0.7958	0.9205
F-CNN (half) (Afifi and Hellwich, 2016)	0.2341	0.7644	0.0970	0.5971	0.8940	0.9720
Ours	**0.0507**	**0.2314**	**0.0218**	**0.9713**	**0.9927**	**0.9972**
Models trained using L2 norm						
F-CNN (full) (Afifi and Hellwich, 2016)	0.3047	1.2146	0.1661	0.3771	0.6662	0.8344
F-CNN (half) (Afifi and Hellwich, 2016)	0.2571	0.9976	0.1317	0.4453	0.7794	0.9202
Ours	**0.1150**	**0.4104**	**0.0479**	**0.8825**	**0.9772**	**0.9935**

trained on different image resolutions. For each resolution, the network generated the depth image that had a size of $\frac{1}{4}$ of the input image size. This is because the model uses two pooling layers which decreases the image resolution twice. Pooling layers are used to decrease the computational costs and reduce the feature dimensions. However, some features are lost from these layers. Our proposed model is an encoder-decoder model where deconvolutional layers (upconvolutional layers) are used to reconstruct the image again to its original resolution. Using these layers allows us to solve the issues related to the output resolution. Furthermore, we use skip connections between the encoder and the decoder. The purpose of the skip connections is to transfer useful information that has been extracted from the encoder part and utilize them when predicting the depth information in the decoder part. These connections improve the quality of the generated images and make the objects' parts sharper. The object parts and the holes in the chairs can be easily recognized from the depth images generated by our encoder-decoder model. To decrease the feature dimensions, we use strided convolutional layers that have features to be learned and can extract useful features, unlike the pooling layers. Also, they preserve the spatial location of the information and feed them to the next layer during the training.

As shown in Fig. 7, the predicted depth images using the proposed method is better than the ones predicted using (Afifi and Hellwich, 2016) that they contain more details. Moreover, our method predicts the depth at a higher quality where the edges and the holes almost match the ground-truth images with fewer artifacts.

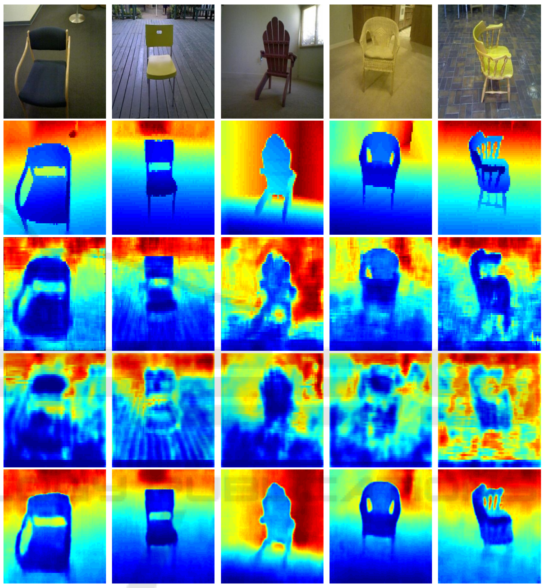

Figure 7: Qualitative comparison results on Large Dataset of Object Scans. From top to bottom: input image, ground-truth, F-CNN (Afifi and Hellwich, 2016) trained on Tukey's biweight loss, F-CNN (Afifi and Hellwich, 2016) trained on L2 norm, and ours. Depths are shown in log scale and in color (blue is close, red is far).

5 CONCLUSION

Single-view depth estimation is an extremely challenging problem. In this paper, we proposed a light and simple fully-convolutional encoder-decoder model for depth estimation from a single RGB image. Unlike the traditional models that require multi-stage or post-processing steps to predict the depth, our model is a simple single-stage model that predicts the depth images directly without any further post-processing steps. By contrast to other methods, that struggle to generate high-resolution images, the generated depth images using the proposed model have the same resolution as that of the input images. We demonstrate that the loss function influences the final output, and for our specific problem the non-convex loss functions are more suitable for regression tasks because they are robust to the outliers. We show that our simple and well-designed model outperforms other models on the same datasets and using the same loss functions during training. Our work generates high-quality depth images that capture the boundaries and reveal finer parts such as the holes in the back.

We believe that the encoder-decoder model for depth estimation can be applied within areas such as scene depth estimation of monocular SLAM and the depth information can be utilized for further applications such as semantic segmentation and scene reconstruction.

REFERENCES

Afifi, A. J. and Hellwich, O. (2016). Object depth estimation from a single image using fully convolutional neural network. In *2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pages 1–7. IEEE.

Black, M. J. and Rangarajan, A. (1996). On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *International Journal of Computer Vision*, 19(1):57–91.

Cao, Y., Wu, Z., and Shen, C. (2018). Estimating depth from monocular images as classification using deep fully convolutional residual networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(11):3174–3182.

Choi, S., Zhou, Q.-Y., Miller, S., and Koltun, V. (2016). A large dataset of object scans. *arXiv preprint arXiv:1602.02481*.

Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *international Conference on computer vision & Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE Computer Society.

Eigen, D. and Fergus, R. (2015). Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, pages 2650–2658.

Eigen, D., Puhrsch, C., and Fergus, R. (2014). Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374.

Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256.

Hadsell, R., Sermanet, P., Ben, J., Erkan, A., Scoffier, M., Kavukcuoglu, K., Muller, U., and LeCun, Y. (2009). Learning long-range vision for autonomous off-road driving. *Journal of Field Robotics*, 26(2):120–144.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Hoiem, D., Efros, A. A., and Hebert, M. (2005). Automatic photo pop-up. *ACM transactions on graphics (TOG)*, 24(3):577–584.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105.

Ladicky, L., Shi, J., and Pollefeys, M. (2014). Pulling things out of perspective. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 89–96.

Liu, B., Gould, S., and Koller, D. (2010). Single image depth estimation from predicted semantic labels. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1253–1260. IEEE.

Liu, F., Lin, G., and Shen, C. (2017). Discriminative training of deep fully connected continuous crfs with task-specific loss. *IEEE Transactions on Image Processing*, 26(5):2127–2136.

Liu, F., Shen, C., and Lin, G. (2015). Deep convolutional neural fields for depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5162–5170.

Liu, M., Salzmann, M., and He, X. (2014). Discrete-continuous depth estimation from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 716–723.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110.

Milletari, F., Navab, N., and Ahmadi, S.-A. (2016). V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE.

Perronnin, F., Sánchez, J., and Mensink, T. (2010). Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer.

Ren, X., Bo, L., and Fox, D. (2012). Rgb-(d) scene labeling: Features and algorithms. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2759–2766. IEEE.

Roberts, R., Sinha, S. N., Szeliski, R., and Steedly, D. (2011). Structure from motion for scenes with large duplicate structures. In *CVPR 2011*, pages 3137–3144. IEEE.

Saxena, A., Sun, M., and Ng, A. Y. (2009). Make3d: Learning 3d scene structure from a single still image. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):824–840.

Silberman, N., Hoiem, D., Kohli, P., and Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Soomro, T. A., Afifi, A. J., Gao, J., Hellwich, O., Zheng, L., and Paul, M. (2019a). Strided fully convolutional neural network for boosting the sensitivity of retinal blood vessels segmentation. *Expert Systems with Applications*, 134:36–52.

Soomro, T. A., Afifi, A. J., Zheng, L., Soomro, S., Gao, J., Hellwich, O., and Paul, M. (2019b). Deep learning models for retinal blood vessels segmentation: A review. *IEEE Access*, 7:71696–71717.

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Suwajanakorn, S., Hernandez, C., and Seitz, S. M. (2015). Depth from focus with your mobile phone. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3497–3506.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Vedaldi, A. and Lenc, K. (2015). Matconvnet: Convolutional neural networks for matlab. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 689–692. ACM.

Xu, B., Wang, N., Chen, T., and Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*.

Zhang, R., Tsai, P., Cryer, J. E., and Shah, M. (1999). Shape-fromshading: a survey. pattern analysis and machine intelligence. *IEEE Transactions on*, 21(8):690–706.