Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue

Ravi Garg(✉), Vijay Kumar B.G., Gustavo Carneiro, and Ian Reid

The University of Adelaide, Adelaide, SA 5005, Australia
{ravi.garg,vijay.kumar,gustavo.carneiro,ian.reid}@adelaide.edu.au

Abstract. A significant weakness of most current deep Convolutional Neural Networks is the need to train them using vast amounts of manually labelled data. In this work we propose a unsupervised framework to learn a deep convolutional neural network for single view depth prediction, without requiring a pre-training stage or annotated ground-truth depths. We achieve this by training the network in a manner analogous to an autoencoder. At training time we consider a pair of images, source and target, with small, known camera motion between the two such as a stereo pair. We train the convolutional encoder for the task of predicting the depth map for the source image. To do so, we explicitly generate an inverse warp of the target image using the predicted depth and known inter-view displacement, to reconstruct the source image; the photometric error in the reconstruction is the reconstruction loss for the encoder. The acquisition of this training data is considerably simpler than for equivalent systems, requiring no manual annotation, nor calibration of depth sensor to camera. We show that our network trained on less than half of the KITTI dataset gives comparable performance to that of the state-of-the-art supervised methods for single view depth estimation.

1 Introduction

The availability of very large human annotated datasets like Imagenet [6] has led to a surge of deep learning approaches successfully addressing various vision problems. Trained initially on tasks such as image classification, and fine-tuned to fit other tasks, supervised CNNs are now state-of-the-art for object detection [14], per-pixel image classification [28], depth and normal prediction from single image [22], human pose estimation [9] and many other applications. A significant and abiding weakness, however, is the need to accrue labeled data for the supervised learning. Providing per-pixel segmentation masks on large datasets like CoCo [23], or classification labels for Imagenet requires significant human effort and is prone to error. Supervised training for single view depth estimation for outdoor scenes requires expensive hardware and careful acquisition [8,21,24,29].

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46484-8_45) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part VIII, LNCS 9912, pp. 740–756, 2016.
DOI: 10.1007/978-3-319-46484-8_45

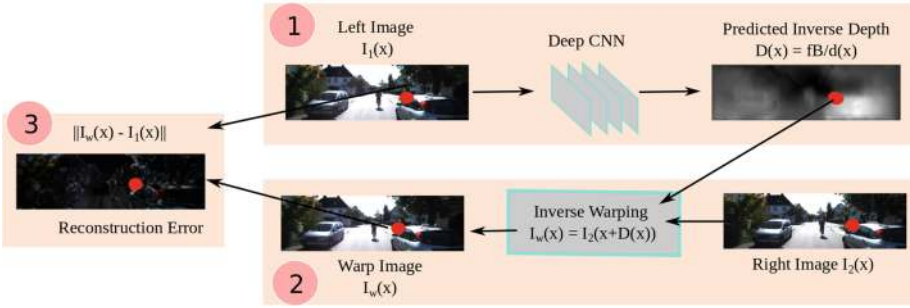

Fig. 1. We propose a stereopsis based auto-encoder setup: the encoder (Part 1) is a traditional convolutional neural network with stacked convolutions and pooling layers (See Fig. 2) and maps the left image (I_1) of the rectified stereo pair into its depth map. Our decoder (Part 2) explicitly forces the encoder output to be disparities (scaled inverse depth) by synthesizing a backward warp image (I_w) by moving pixels from right image I_2 along the scan-line. We use the reconstructed output I_w to be matched with the encoder input (Part 3) via a simple loss. For end-to-end training, we minimize the reconstruction loss with a simple smoothness prior on disparities which deals with the aperture problem, while at test time our CNN performs single-view disparity (inverse depth) prediction, up to the scene scale given in form of fB at the time of training.

For example, despite using state-of-the-art 3D sensors, multiple calibrated cameras and inertial sensors, a dataset like KITTI [13] provides sparse depthmaps with less than 5 % density on the captured image resolutions and with only a limited reliable depth range. A significant challenge now is to develop unsupervised training regimes that can train networks that perform either as well as, or better than those trained used using these supervised methods. This will be a major step towards realizing in-situ learning, in which we can retrain or tune a network for specific circumstances, and towards life-long learning, in which continuous acquisition of data leads to improved performance over time.

In this paper we are particularly concerned with the task of single-view depth estimation, in which the goal is to learn a non linear prediction function which maps an image to its depth map. CNNs have achieved the state-of-the-art performance on this task due to their ability to capture the complex and implicit relationships between scene depth and the corresponding image textures, scene semantics, and local and global context in the image. State-of-the-art supervised learning methods for this task train a CNN to minimize a loss based on either the scale invariant RMS [8], or the log RMS [24] of the depth predictions from ground-truth. These networks have been trained using datasets that provide both RGB images and corresponding depthmaps such as NYUv2 and KITTI.

However as noted in [24], the networks learned by these systems do not generalize well outside their immediate domain of application. For example, [24] trained two separate networks, one for indoors (using NYUv2) and one for street scenes (using KITTI), because the weights learned in one do not work well in the other. To transfer the idea of single-view depth estimation into yet another domain would require indulging in the expensive task of acquiring a new RGB-

D dataset with well aligned image and depth values, and re-train the network. An alternative to this would be to generate a large synthetic or semi-synthetic dataset using graphical rendering, an approach that has met with some success in [15]. However it is difficult to capture the full variability of real-world images in such datasets.

Another possible approach would be to capture a large dataset of stereo images, and use standard geometric methods to compute the disparity map for each pair, yielding a large set of image-plus-disparity-map pairs. We could then train a network to predict a disparity map from a single view. However such system will likely learn the systematic errors in estimated depths, "baking in" the failure modes of the stereo algorithm. Factors such as sensor flare, motion blur, lighting changes, shadows, etc. are present in real images and rarely dealt with adequately by standard stereo algorithms.

We adopt a different approach that moves towards a system capable of in-situ training or even lifelong learning, using real un-annotated imagery. We take inspiration from the idea of autoencoders, and leverage well-understood ideas in visual geometry. The result is a convolutional neural network for single-view depth estimation, the first of its kind that can be trained end-to-end from scratch, in a fully unsupervised fashion, simply using data captured using a stereo rig.

2 Approach

In this section we give more detail of our approach. Figure 1 explains our idea graphically. To train our network, we make use of pairs of images with a known camera motion between the two, such as stereo pairs. Such data are considerably more easily acquired than calibrated depthmaps and aligned images. In our case we use large numbers of stereo pairs, but the method applies equally to data acquired from a moving SLAM system in an otherwise static scene.

We learn a CNN to model the complex non-linear transformation which converts the image to a depth-map. The loss we use for learning this CNN is the photometric difference between the input – or source – image, and the inverse-warped target image (the other image in the stereo pair). This loss is both differentiable (to facilitate back-propagation) and is highly correlated with the prediction error - i.e. can be used to accurately rank two different depth-maps without using ground-truth labels.

This approach can be interpreted in the context of convolutional autoencoders. The task of the standard autoencoder is to encode the input with a series of non-linear operations to a compressed code that captures sufficient core information so that a decoder can reconstruct the input with minimal reconstruction error. In our case we replace the decoder with a standard geometric image warp, based on the predicted depth map and the relative camera positions. This has two advantages: first, the decoder in our case does not need to be learned, since it is already a well-understood geometric operation; second, our reconstruction loss naturally encourages the code to be the correct depth image.

2.1 Autoencoder Loss

Every training instance $i \in \{1 \cdots N\}$ in our setup is a rectified stereo pair $\{I_1^i, I_2^i\}$ captured by a single pre-calibrated stereo rig with two cameras having focal length f each which are separated horizontally by a distance B.[1] Assuming that the predicted depth of a pixel x for the left image of the rig via CNN is $d^i(x)$, the motion of the pixel along the scan-line $D^i(x)$ is then $fB/d^i(x)$. Thus, using the right image I_2^i, a warp I_w^i can be synthesized as $I_2^i(x + fB/d^i(x))$.

With this explicit parameterization of the warp, we propose to minimize standard color constancy (photometric) error between the reconstructed image I_w^i and the left image I_1^i:

$$E^i_{recons} = \int_\Omega \|I^i_w(x) - I^i_1(x)\|^2 dx = \int_\Omega \|I^i_2(x + \underbrace{D^i(x)}_{fB/d^i(x)}) - I^i_1(x)\|^2 dx \tag{1}$$

It is well known that this photometric loss function is non-informative in homogeneous regions of the scene. Thus multiple disparities can generate equally good warps I_w's and a prior on the disparities is needed to get a unique depthmap. We use very simple $L2$ regularization on the disparity discontinuities as our prior to deal with the aperture problem:

$$E^i_{smooth} = \|\nabla D^i(x)\|^2 \tag{2}$$

This regularizer is known to over-smooth the estimated motion, however a vast literature of more sophisticated edge preserving regularizers with robust penalty functions like [2,33] for which gradients can be computed are at our disposal and can be easily used with our setup to get sharper depthmaps. As the main purpose of our work is to prove that end-to-end training of the proposed autoencoder is feasible and helpful for depth prediction, we choose to minimize the simplest suitable loss summed over all training instances:

$$E = \sum_{i=1}^{N} E^i_{recons} + \gamma E^i_{smooth} \tag{3}$$

where γ is the strength of the regularization forcing the estimated depthmaps to be smooth.

Our loss function as described in (3) is similar to the standard Horn and Schunck optic flow cost [17] for every frame. However, the major difference is that our disparity maps D^i's are parametrized to be a non-linear function of the input image and unknown weights of the CNN which are shared for estimating the motion between every stereo pair. This parameter sharing enforces consistency in the estimated depths over 1000's of correlated training images of a large dataset like KITTI. Our autoencoder's reconstruction loss can be seen as a generalization

[1] All training images are assumed to be taken with a fixed rectified stereo setup as is the case in KITTI for simplicity but our method is generalizable to work with instances taken by different calibrated stereos.

of the multi-frame optic flow methods like [11,12]. The difference is, instead of modeling the correlations in the estimated motions for a shorter video sequence with a predefined linear subspace [11], our autoencoder learns (and models) valid flows which are consistent throughout the dataset non-linearly.

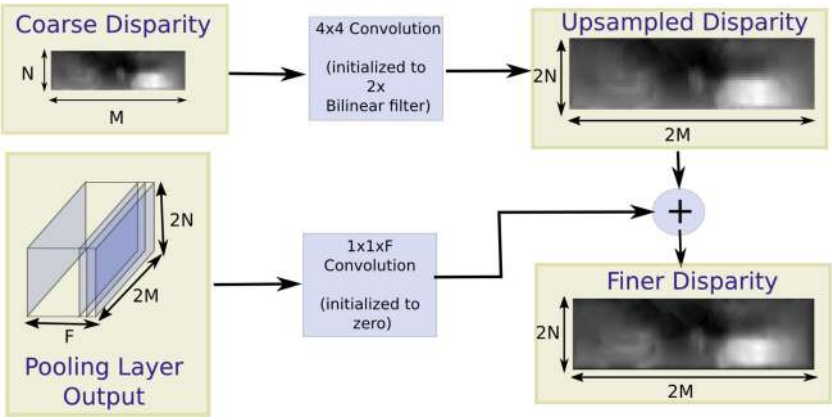

Fig. 2. Coarse-to-fine stereo with CNN with results on a sample validation instance: We adapt the convolution based upsampling architecture proposed in [26] to mimic the coarse-to-fine stereo estimations. Our upsampling filter is initialized with simple bilinear interpolation kernel and we initialize the corresponding pooling layer contribution by setting both bias and 1×1 convolution filter to be zero. The figure shows how features coming from previous layers of the CNN (L3) combined with finer resolution loss function generate better depthmaps at 44×172 from our bilinear upsampled initial estimate of coarser prediction at 22×76.

3 Coarse-to-Fine Training with Skip Architecture

To compute the gradient for standard back-propagation on our cost (1), we need to linearize the warp image at the current estimate of the disparities using Taylor expansion:

$$I_2(x + D^n(x)) = I_2(x + D^{n-1}(x)) + (D^n(x) - D^{n-1}(x))I_{2h}(x + D^{n-1}(x)) \quad (4)$$

where I_{2h} represents the horizontal gradient of the warp image computed at the current disparity D^{n-1} at iteration n.[2] This linearization is valid only for small values of $D^n(x) - D^{n-1}(x)$ limiting the magnitude of estimated disparities in the image. To estimate larger motions (smaller depths) accurately, a coarse-to-fine strategy with iterative warping is well established in the stereo and optic flow literature which facilitates gradient descent-based continuous optimization. We

[2] We have dropped the training instance index i for simplicity.

refer the readers to [30] for more detailed discussion of the requirements of this linearization, its limitations and existing alternatives.

However, our disparities are a non-linear function of the CNN parameters and the input image. To move from coarse-to-fine level, we not only need a good disparity initialization at the finer resolutions to linearize the warps but also the corresponding CNN parameters which predict these initial disparities for each training instance. Fortunately, the recent fully-convolutional architecture with upsampling, proposed in [26], is a suitable choice to enable coarse-to-fine warping for our system. As depicted in Fig. 2, given a network which predicts an $M \times N$ disparities, we can use a simple bilinear upsampling filter to initialize upscaled disparities (to get $2M \times 2N$ depthmaps) keeping the other network parameters fixed. It has been shown that the finer details of the images are captured in the previous layers of CNN, and fusing back such information is helpful for refining a coarse CNN prediction. We use 1×1 convolution with the filter and bias both initialized to zero and the convolved output is then fused with the upscaled depths with an element-wise sum layer for refinement.

4 Network Architecture

The network architecture for our deep convolutional encoder is shown in Fig. 3 which is similar to the Alexnet architecture [19] up to the C5 layer. We replace the fully connected layer of Alexnet by a fully convolutional layer with 2048 convolution filters of size 5×5 each.[3] This reduces the number of parameters in the network and allows for the network to accept variable size inputs at test time. More importantly, it preserves the spatial information present in the image and allows us to upsample the predictions in a stage-wise manner in the layers that follow the L7 output of the figure, which is a requirement for our stereopsis based autoencoder. Inspired by the observations from [26], that the finer details in the images are lost in the last few layers of the deep convolutional network we employ the "skip architecture" that combines the coarser depth prediction with the local image information to get finer predictions. The effect of this is illustrated using an example from the validation set in Fig. 2. The layers following the L9 output (22×76 depthmap) in our network are simple 4×4 convolutions each converting a coarser low resolution depth map to a higher resolution output.

5 Experiments

We evaluate our method on the publicly available KITTI dataset [13] that comprises several outdoor scenes captured using a stereo camera mounted on a moving vehicle. We employ the same train/test split used in [8]: from the 56 scenes belonging to the categories "city", "residential" and "road", we choose 28 for training and the remaining 28 for testing. We downsample the left images by a

[3] A 5×18 convolution can be used instead to increase network capacity and replicate the effect of a fully connected layer of [19].

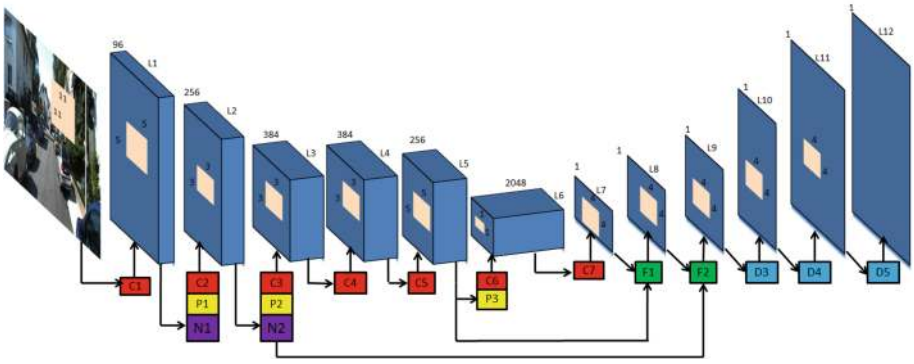

Fig. 3. Network architecture: The blocks C (red), P (yellow), L (dark blue), F (green), D (blue) correspond to convolution, pooling, local response normalization, FCN and upsampling layers respectively. The FCN blocks F1 and F2 upsample the predictions from layers (L7, L8) and combine it with the input of the pooling layers P3 and P2 respectively. (Color figure online)

factor of 2 to bring them to 188×620, and at this resolution they are used as input to the network. Each corresponding right image in a stereo pair is used at the resolution of the predicted depthmap at each stage of our coarse-to-fine training to generate the warp and match it with a resized left image.

The training set consists of 23488 stereo pairs out of which we use 22600 for training and the remaining for validation. Neither the right to left stereo nor any data augmentation are used for the coarse-to-fine training in multiple stages. For testing, we use the 697 images provided by [8]. We do not use any ground-truth depths for training the network. To evaluate all the results produced by our network we use simple upscaling of the low resolution disparity predictions to the resolution at which the stereo images were captured. Using the stereo baseline of 0.54 m, we convert the upsampled disparities to generate depthmaps at KITTI resolution using $d = fB/D$.

For fair comparison with state-of-the-art single view depth prediction, we evaluate our results on the same cropped region of interest as [8]. Since the supervised methods are trained using the ground-truth depth that ranges between 1 and 50 m whereas we can predict larger depths, we clamp the predicted depth values for our method between 1 and 50 for evaluation. i.e. setting the depths bigger than 50 m to 50. We evaluate our method using the error measures reported in [8,24]:

$$\text{RMS}: \sqrt{\frac{1}{T}\sum_{i \in T} \|d_i - d_i^{gt}\|^2} \qquad \text{log RMS}: \sqrt{\frac{1}{T}\sum_{i \in T} \|log(d_i) - log(d_i^{gt})\|^2}$$

$$\text{abs. relative}: \frac{1}{T}\sum_{i \in T} \frac{|d_i - d_i^{gt}|}{d_i^{gt}} \qquad \text{sq. relative}: \frac{1}{T}\sum_{i \in T} \frac{\|d_i - d_i^{gt}\|^2}{d_i^{gt}}$$

$$\text{Accuracies: \% of } d_i \text{ s.t. } max\left(\frac{d_i}{d_i^{gt}}, \frac{d_i^{gt}}{d_i}\right) = \delta < thr$$

Table 1. Performance of the proposed framework at various stages of training.

Methods	Resolution	RMS	*log* RMS	Absolute relative	Square relative	Accuracies $\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Ours L9	22×76	5.740	0.310	0.205	1.353	0.660	0.872	0.948
Ours L10 + skip[a]	46×154	5.850	0.338	0.246	1.673	0.607	0.842	0.937
Ours L10	44×152	5.434	0.292	0.189	1.214	0.705	0.889	0.955
Ours L11	88×304	5.326	0.285	0.179	1.177	0.721	0.892	0.958
Ours L12	176×608	5.285	0.282	0.177	1.169	0.727	0.896	0.958
Ours L12, Aug. 8x		5.104	0.273	0.169	1.08	0.740	0.904	0.962

[a] Layer 10 result while using 3rd skip-connection.

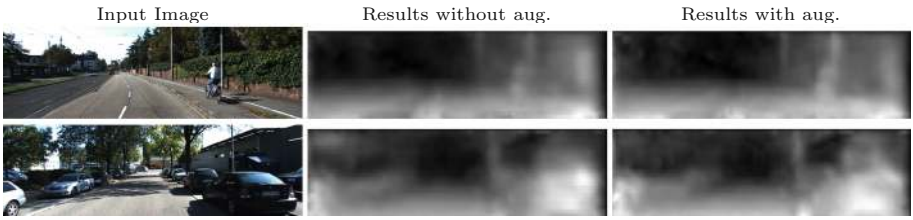

Fig. 4. Data augmentation improves the predicted disparities for smaller objects. Look at the biker in the first and the bottom right car in the second example.

5.1 Implementation Details

We train our network using the CNN toolbox MatConvnet [31]. We use SGD for optimization with momentum 0.9 and weight decay of 0.0005. Our network weights are initialized randomly for the first 5 layers of the Alexnet and we append the 5×5 fully convolutional layer initialized with zero weights to get zero disparity estimates. We subtract every pixel's color by 128 and divide it by 255 to have both left and right images $\in [-0.5, 0.5]$. The smoothness prior strength γ was set to 0.01.

Due to the linearization of the loss function as explained in Sect. 3, we learn the network proposed in Fig. 3 in multiple stages, starting from the coarsest level (L7 in Fig. 3), and iteratively adding upsampling layers one at a time. The learning rate for the network which predicts depths at the coarsest resolution is initialized to 0.01 and gradually decreased after each epoch using the factor $1/(1 + \alpha * n)^{(n-1)}$ where n is the index of current epoch and $\alpha = 0.0005$. The smoothness prior strength γ was set to 0.01. We train this coarse depth prediction network (L1-L7) for 100 epochs.

5.2 Effect of Upsampling

Having the coarser depth estimates for the training-set, we iteratively add upsampling layers which increases the resolution of the predictions by a fac-

tor of ≈ 2.[4] Since the number of pixels in the images are increased by a factor of 4, the cost approximately increases by the same factor when moving from coarser to finer level training. Hence we decrease the initial learning rate by a factor of 4 for training the finer networks. Starting from the coarsest predictions (L7) we progressively add upsampling layers L8 to L12 to get depths at resolutions $10 \times 37, 22 \times 76, 44 \times 152, 88 \times 304$ and 176×608 respectively. We train each of the finer networks for 100 epochs with the decaying learning rate as described in previous section. While adding the upsampling layers, we crop and pad the layers such that the resolution of predictions in L8 and L9 matches the resolution of the input to the pooling layers P3 and P2 respectively. For the upsampling layers without skip-connection padding of 1 pixel is used.

Table 1 analyses the disparity estimation accuracy for our network on the KITTI test-set at various stages of the training. Row 1 and 2 of our table correspond to our L9 and L10 output with 2 and 3 FCN blocks respectively. Consistent with [26] we also observe that after 2 upsampling layers, the skipped architecture starts to give diminishing returns. As evident from the third row in Table 1 layer L10 without skip-connection outperforms the counterpart. We believe that this is due to the fact that the features learned in the first few layers of the CNN are more relevant to ordinary photometric images than to the depth images. Thus, a simple weighted sum of these features with that of the depth map does not work well. However, higher resolution images still have richer information for image correspondences which can be back-propagated via our loss function for better predictions. The gradual improvement in disparity estimations using high resolution images is evident in Table 1.

5.3 Fine Tuning with Augmentation

Once we have our base network trained in the stage-wise manner described above, we further fine-tune this network (without coarse-to-fine training) for another 100 epochs with following augmentations:

- Color (2×): Color channels are multiplied by a factor $c \in [0.9, 1.1]$ randomly.
- Scale (2×): We scale the input image by a factor of $s \in [1, 1.6]$ and randomly crop the images to match the network input size.
- Left-Right flips (2×): We flip left and right images horizontally and swap them to get new training pair with positive disparities to keep consistency.

Consistent with other CNNs, fine tuning our network with this new augmented dataset leads to noticeable improvements in depth prediction. Figure 4 illustrates how 8× data for the fine tuning improves the reconstructions. Notice in particular the improved localization of object edges. This is particularly encouraging for our stereopsis loss based unsupervised training procedure as its fine tuning only requires a cheap stereo-rig to collect new data in the wild. For example, we can resort to much larger road scene understanding dataset like cityscapes [5]

[4] Alexnet uses uneven padding for some convolutions leading to change in the aspect ratio and the image size.

Table 2. Comparison with state-of-the-art methods on KITTI dataset.

Methods	Resolution	RMS	*log*RMS	Absolute relative	Square relative	$\delta < 1.25$	Accuracies $\delta < 1.25^2$	$\delta < 1.25^3$
Ours L12	176 × 608	5.285	0.282	0.177	1.169	0.727	0.896	0.958
Ours L12, Aug 8x		**5.104**	0.273	**0.169**	**1.080**	**0.740**	**0.904**	0.962
Mean	-	9.635	0.444	0.412	5.712	0.556	0.752	0.870
Make3D [29]	Dense	8.734	0.361	0.280	3.012	0.601	0.820	0.926
Eigen *et al.* (c^a) [8]	28 × 144	7.216	0.273	0.194	1.531	0.679	0.897	**0.967**
Eigen *et al.* (f) [8]	27 × 142	7.156	**0.270**	0.190	1.515	0.692	0.899	**0.967**
Fayao *et al.* (pt) [24]	superpix	7.421	-	-	-	0.613	0.858	0.949
Fayao *et al.* (ft) [24]	superpix	7.046	-	-	-	0.656	0.881	0.958

[a] *c* and *f* indicates the coarse and fine networks of [8]. Also pt and ft indicates the pre-trained and fine-tuned networks of [24]

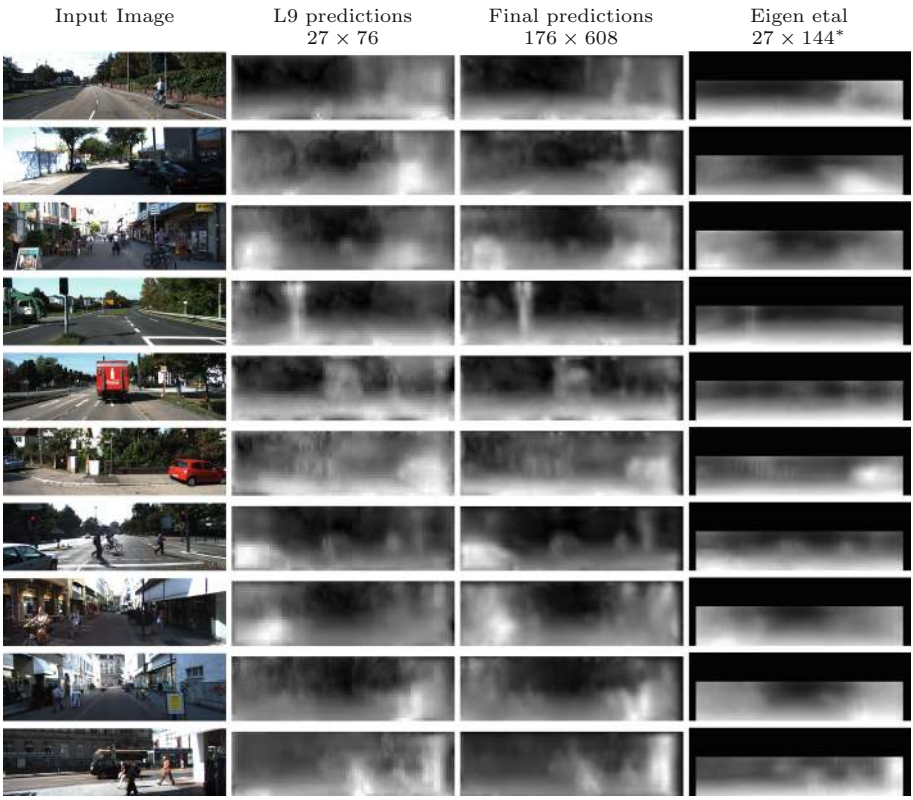

Fig. 5. Inverse Depths visualizations. Brighter color means closer pixel.

(captured without laser sensor) or a vast collection of 3D movies much like recently published work Deep3D [32] to repeat this fine-tuning experiment for single view depth prediction in the wild.

5.4 Comparison with State-of-the-Art Methods on KITTI Dataset

In Table 2, we compare the performance our network with state-of-the-art single view depth prediction methods [8,24,29]. Errors for other methods are taken from [8,24]. Our method achieves the lowest RMS and Square relative error on the dataset and significantly outperforms other methods for these measures. It performs on par with the state-of-the-art methods on all other evaluation measures. Eigen *et al.* [8] obtains slightly lower error in terms of *log* RMS compared to ours. However, as [8,24] are trained by minimizing *log* RMS error with respect to the true depths, we expect the best performance of these methods under same metric.

The most noteworthy point is that our is a completely unsupervised network trained with randomly initialized weights, whereas [8,24] initialize the networks using Alexnet and VGG-16 respectively, and are supervised.

Figure 5 compares the output inverse depthmaps (scaled to [0 1]) for the L9 (2^{nd} column) and L12 (3^{rd} column) layers of the proposed method and [8]. We appropriately pad the predictions provided by the authors of [8] to generate the visualizations at the correct scale. It is evident from the figure that both L9 and L12 are able to capture objects that are closer to the camera with significantly more details. For example, notice the traffic light in Row 4, truck in Row 5 and pedestrians in Row 6 and Row 10; these important scene elements are "washed out" in the predictions generated by [8]. Edges are localized more accurately in L12 results compared to L9. This depicts that even with the simple linear interpolation of the coarse depth estimation, the finer image alignment errors are correctly back-propagated leading to the performance boost. Blurred object boundaries in the finer reconstructions point to well-known limitations of upsampling based approaches which to a certain extent can be addressed with the atrous algorithm [3], a fully connected CRF [18,35] or polynomial interpolations replacing simple linear interpolation layers.

In summary, our simple, skinnier network than [8] gives on par results without any supervision, and which look visually more appealing. Our results could be further refined using better loss functions and replacing linear interpolation filter with a learned CRF. As our method is completely unsupervised, it can be trained on theoretically limitless data with deeper networks to capture variation and give depthmaps at full image resolutions.

5.5 Comparisons with Baseline Supervised Networks and Stereo

As discussed in Sect. 1, an alternative to our proposal of directly minimizing the loss (3), would be to train with a standard "depth loss" using the output of an off-the-shelf stereo algorithm to generate proxy ground-truth depth for

Table 3. Comparison of proposed auto-encoder framework with a supervised CNN trained on stereo data, and stereo baselines on the KITTI dataset.

Methods	Coverage	RMS	*log* RMS	Absolute relative	Square relative	$\delta < 1.25$	Accuracies $\delta < 1.25^2$	$\delta < 1.25^3$
Ours L12	100 %	**5.285**	**0.282**	**0.177**	**1.169**	**0.727**	**0.896**	**0.958**
HS→ CNN,$\gamma = .01$	100 %	6.691	0.385	0.309	2.657	0.476	0.750	0.891
HS→ CNN	100 %	6.292	0.338	0.238	1.639	0.573	0.841	0.941
SGM→ CNN	100 %	5.680	0.300	0.185	1.370	0.703	0.886	0.955
HS-Stereo, $\gamma = .01$	**100 %**	6.077	0.381	0.299	3.264	0.677	0.822	0.90
HS-Stereo	**100 %**	6.760	0.366	0.254	4.040	0.754	0.872	0.928
SGM-Stereo	87 %	3.030	0.150	0.064	0.506	0.955	0.979	0.989

training. In this section, we substantiate that the proposed autoencoder framework is superior to this alternative approach (which we denote as Stereo → CNN). For this purpose, we train the network described in Fig. 3, end-to-end, with least square loss on the disparity difference between CNN prediction and stereo prediction.[5]

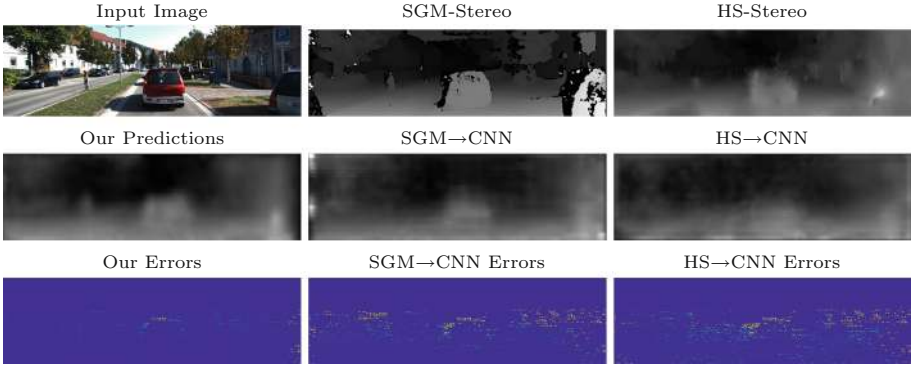

Fig. 6. Comparing depth predictions baseline stereo methods (top row), with the proposed unsupervised CNN (left column-middle row) and Stereo→CNN approaches (center/right column-middle row). Bottom row shows the depth estimation errors as heatmaps for the corresponding methods in middle row.

To generate the stereo prediction, we use a variational Horn-Schunck algorithm. While this is clearly not a state-of-the-art stereo algorithm, it is a fair baseline since this is the same loss on which we train our photometric loss network. We use the OpenCV implementation, with 6 coarse-to-fine pyramid levels with scale factor 0.5. To make sure the algorithm converges properly, we

[5] Much like the log depth, inverse depth parametrization is less prone to the higher depth errors at very distant points and is used successfully in many stereo [13] and SLAM frameworks [27].

increased the number of warp iterations to 1000. We additionally tried HS → CNN with the disparity regularization strength $\gamma = 0.01$ as well, but the results were less accurate.

As shown in Table 3, depth prediction accuracy of this HS → CNN baseline falls significantly short of the proposed framework on all accuracy measures. We also incorporate the test-set depth estimation accuracies for the baseline HS-stereo method (which uses both left and right image) for the reference. A very surprising observation is that our single view depth prediction network works on par with even HS-stereo thanks to the common structure present in the road scenes that our network successfully learns. Having access to two images, HS-stereo was able to estimate disparity of the closer points with much more precision but over-reliance on the depth regularization and unawareness of the scene context results in wrong depths near edges – where the single view depth estimation even outperforms the HS Stereo.

In addition to HS-Stereo, we also used Semi Global Matching (SGM) algorithm [16] to supervise the CNN. Semi Global matching is known to produce more accurate depths and is an integral part of many of the state-of-the-art stereo algorithms on KITTI stereo dataset [13]. This stereo method gave very accurate results on the test-set for 87 % of the pixels but left holes in the reconstructions. We train SGM → CNN by minimizing the sum of least square error for predicted disparities on the training data, ignoring the points where SGM gave no disparity. We observed SGM → CNN performed on par with the state-of-the-art fully supervised single view depth estimation algorithm but the results were not as accurate as the proposed approach. We believe that the reason for this was the systematic holes which were left in the SGM-Stereo reconstructions.

To validate this, in Fig. 6 we analyze if regions with lower depth accuracy of SGM→CNN coincide with the holes left by SGM-Stereo. The correlation in errors SGM-Stereo depthmap with that of SGM →CNN suggests that the supervised training with proxy ground-truth indeed is prone to learn systematic errors in the proxy ground truth and advocates need for a more principled integration of a state-of-the-art stereo method with deep learning. The proposed autoencoder setup is the reasonable first step towards this goal.

6 Related Work

In this work we have proposed a geometry-inspired unsupervised setup for visual learning, in particular addressing the problem of single view depth estimation. Our main objective was to address the downsides of training deep networks with large amount of labeled data. Another body of work which attempts to address this issue is the set of methods like [7,15,20] which rely mainly on generating synthetic/semi-synthetic training data with the aim to mimic the real world and use it to train deep network in a *supervised* fashion. For example, in [7], CNN is used to discriminate a set of surrogate classes where the data for each class is generated automatically from unlabeled images. The network thus learned is shown to perform well on the task image classification. Handa *et al.* [15] learn

a network for semantic segmentation using synthetic data of indoor scenes and show that the network can generalize well on the real-world scenes. Similarly, [20] employs a CNN to learn local image descriptors where the correspondences between the patches are obtained using a multi-view stereo algorithm.

Recently, many methods have used CNN to learn good visual features for matching patches which are sampled from stereo datasets like KITTI [4,34], and match these features while doing classical stereo to achieve state-of-the-art depth estimation. These methods are reliant on local matching and lose global information about the scene; furthermore they use ground-truth. But their success is already an indicator that a joint visual learning and depth estimation approach like ours could be extended at the test time to use a pair of images.

There have been few works recently that approach the problem of novel view synthesis with CNN [10,32]. Deep stereo [10] uses a large set of posed images to learn a CNN that can interpolate between the set of input views that are separated by a wide baseline. A concurrent work with ours, [32] addresses the problem of generating 3D stereo pairs from 2D images. It employs a CNN to infer a soft disparity map from a single view image which in turn is used to render the second view. Although, these methods generate depth-like maps as an intermediate step in the pipeline, their goal however is to generate new views and hence do not evaluate the computed depth maps

Using camera motion as the information for visual learning is also explored in the works like [1,25] which directly regress over the 6DOF camera poses to learn a deep network which performs well on various visual tasks. In contrast to that work, we train our CNN for a more generic task of synthesizing image and get the state-of-the-art single view depth estimation. It will be of immense interest to evaluate the quality of the features learned with our framework on other semantic scene understand tasks.

7 Conclusions

In spite of the enormous growth and success of deep neural networks for a variety of visual tasks, an abiding weakness is the need for vast amounts of annotated training data. We are motivated by the desire to build systems that can be trained relatively cheaply without the need for costly manual labeling or even trained on the fly. To this end we have presented the first convolutional neural network for single-view depth estimation that can be trained end-to-end from scratch, in a fully unsupervised fashion, simply using data captured using a stereo rig. We have shown that our network trained on less than half of the KITTI dataset gives comparable performance to the current state-of-the-art supervised methods for single view depth estimation.

Various natural extensions to our work present themselves. Instead of training on KITTI data (which is nevertheless convenient because it provides a clear baseline) we aim to train on a continuous feed from a stereo rig "in the wild", and to explore the effect on accuracy by augmenting the KITTI data with new stereo pairs. Furthermore, as intimated in the Introduction, our method is not restricted

to stereo pairs, and a natural extension is to use a monocular SLAM system to compute camera motion, and use this known motion within our autoencoder framework; here the warp function is slightly more complex than for rectified stereo, but still well understood. The resulting single-view depth estimation system could be used for bootstrapping structure, or generating useful priors on the scene structure that capture much richer information than typical continuity or smoothness assumptions. It also seems likely that the low-level features learned by our system will prove effective for other tasks such as classification, in a manner analogous to [1,7], but this hypothesis remains to be proven experimentally.

Acknowledgments. This research was supported by the Australian Research Council through the Centre of Excellence in Robotic Vision, CE140100016, and through Laureate Fellowship FL130100102 to IDR.

References

1. Agrawal, P., Carreira, J., Malik, J.: Learning to see by moving. In: IEEE International Conference on Computer Vision (ICCV) (2015)
2. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Pajdla, T., Matas, J.G. (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)
3. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: International Conference on Learning Representations (ICLR) (2015)
4. Chen, Z., Sun, X., Wang, L., Yu, Y., Huang, C.: A deep visual correspondence embedding model for stereo matching costs. In: IEEE International Conference on Computer Vision (ICCV) (2015)
5. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2009)
7. Dosovitskiy, A., Springenberg, J.T., Riedmiller, M., Brox, T.: Discriminative unsupervised feature learning with convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS) (2014)
8. Eigen, D., Puhrsch, C., Fergus, R.: Depth map prediction from a single image using a multi-scale deep network. In: Advances in Neural Information Processing Systems (NIPS) (2014)
9. Fan, X., Zheng, K., Lin, Y., Wang, S.: Combining local appearance and holistic view: dual-source deep neural networks for human pose estimation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
10. Flynn, J., Neulander, I., Philbin, J., Snavely, N.: Deepstereo: learning to predict new views from the world's imagery (2016)
11. Garg, R., Pizarro, L., Rueckert, D., Agapito, L.: Dense multi-frame optic flow for non-rigid objects using subspace constraints. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part IV. LNCS, vol. 6495, pp. 460–473. Springer, Heidelberg (2011)

12. Garg, R., Roussos, A., Agapito, L.: Dense variational reconstruction of non-rigid surfaces from monocular video. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2013)
13. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: the KITTI dataset. Int. J. Robot. Res. (IJRR) **32**, 1229–1235 (2013)
14. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
15. Handa, A., Patraucean, V., Badrinarayanan, V., Stent, S., Cipolla, R.: Scenenet: understanding real world indoor scenes with synthetic data. arXiv preprint (2015). arXiv:1511.07041
16. Hirschmuller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2005)
17. Horn, B.K., Schunck, B.G.: Determining optical flow. In: 1981 technical symposium east, pp. 319–331. International Society for Optics and Photonics (1981)
18. Koltun, V.: Efficient inference in fully connected CRFs with Gaussian edge potentials. In: Neural Information Processing Systems (NIPS) (2011)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS) (2012)
20. Vijay Kumar, B.G., Carneiro, G., Reid, I.: Learning local image descriptors with deep siamese and triplet convolutional networks by minimising global loss functions. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
21. Ladicky, L., Shi, J., Pollefeys, M.: Pulling things out of perspective. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2014)
22. Li, B., Shen, C., Dai, Y., van den Hengel, A., He, M.: Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
23. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part V. LNCS, vol. 8693, pp. 740–755. Springer, Heidelberg (2014)
24. Liu, F., Shen, C., Lin, G., Reid, I.: Learning depth from single monocular images using deep convolutional neural fields. In: IEEE Transactions on Pattern Analysis and Machine Intelligence (2016)
25. Long, G., Kneip, L., Alvarez, J.M., Li, H.: Learning image matching by simply watching video. CoRR abs/1603.06041 (2016). http://arxiv.org/abs/1603.06041
26. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
27. Newcombe, R.A., Lovegrove, S.J., Davison, A.J.: Dtam: dense tracking and mapping in real-time. In: IEEE International Conference on Computer Vision (ICCV) (2011)
28. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: IEEE International on Computer Vision (ICCV) (2015)
29. Saxena, A., Sun, M., Ng, A.: Make3d: learning 3d scene structure from a single still image. IEEE Trans. Pattern Anal. Mach. Intell. (PAMI) **31**, 824–840 (2009)

30. Steinbrücker, F., Pock, T., Cremers, D.: Large displacement optical flow computation withoutwarping. In: IEEE International Conference on Computer Vision (ICCV) (2009)
31. Vedaldi, A., Lenc, K.: Matconvnet - convolutional neural networks for matlab (2015)
32. Xie, J., Girshick, R., Farhadi, A.: Deep. 3d: fully automatic 2d-to-3d video conversion with deep convolutional neural networks. arXiv preprint (2016). arXiv:1604.03650
33. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime TV-L^1 optical flow. In: Hamprecht, F.A., Schnörr, C., Jähne, B. (eds.) Pattern Recognition, vol. 4713, pp. 214–223. Springer, Heidelberg (2007)
34. Zbontar, J., LeCun, Y.: Computing the stereo matching cost with a convolutional neural network. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
35. Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D., Huang, C., Torr, P.H.S.: Conditional random fields as recurrent neural networks. In: IEEE International Conference on Computer Vision (ICCV) (2015)