International Journal of Innovative Computing, Information and Control
Volume **16**, Number **1**, February **2020**

ICIC International ©2020 ISSN 1349-4198
pp. **91–106**

UNSUPERVISED MONOCULAR DEPTH ESTIMATION OF DRIVING SCENES USING SIAMESE CONVOLUTIONAL LSTM NETWORKS

John Paul Tan Yusiong[1,2] and Prospero Clara Naval, Jr.[1]

[1]Computer Vision and Machine Intelligence Group
Department of Computer Science
College of Engineering
University of the Philippines
Diliman, Quezon City 1101, Philippines
jtyusiong@up.edu.ph; pcnaval@dcs.upd.edu.ph

[2]Division of Natural Sciences and Mathematics
University of the Philippines Visayas Tacloban College
Tacloban City, Leyte 6500, Philippines

Received May 2019; revised September 2019

Abstract. *Estimating depth from a single RGB image is an active research topic in computer vision because of its broad applications in scene understanding, autonomous driving, and traffic surveillance systems. This task involves estimating a pixel-wise depth map from a single image. Significant progress has been made on monocular depth estimation using deep learning-based techniques. Current approaches employ geometry-based image reconstruction methods instead of ground truth depth labels to perform depth estimation in an unsupervised manner. In this paper, we present a deep learning model to simultaneously learn and refine depth maps from a single RGB image and in an end-to-end manner by casting the monocular depth estimation as an image reconstruction problem. We propose an unsupervised framework for monocular depth estimation that trains a Siamese convolutional long short-term memory (Siamese convLSTM) network to jointly perform estimation and refinement of depth maps using rectified stereo image pairs and produce a depth map from a single RGB image at test time. Experimental results show that simultaneously performing these two tasks leads to improving depth estimation accuracy. In particular, using the KITTI 2015 driving dataset for evaluation, our proposed Siamese convLSTM network achieves excellent performance on monocular depth estimation, both quantitatively and qualitatively.*
Keywords: Monocular depth estimation, Disparity refinement, Siamese convolutional LSTM networks, Stereo vision, Unsupervised learning

1. **Introduction.** Many computer vision problems benefit from the incorporation of depth information [1, 2, 3]. Depth estimation from a single RGB image or monocular depth estimation is an active research topic in computer vision because of its broad applications in scene understanding, autonomous driving, and traffic surveillance systems [4]. This problem deals with estimating a pixel-wise depth map from a single RGB image and is formulated as a regression task that performs pixel-wise depth reconstruction in which an image of $H \times W \times 3$ resolution is represented as an $H \times W \times 1$ depth map [5]. Inspired by the recent developments of deep learning methods in solving many computer vision problems [6, 7, 8], researchers also utilized deep learning methods to develop better-suited models for image-based depth prediction problems such as the monocular depth estimation problem. As a result, significant progress has been made on monocular depth

DOI: 10.24507/ijicic.16.01.91

estimation using deep learning-based techniques. These methods rely on training deep network architectures to learn a set of kernels to extract and combine local and global features from a single RGB image to infer depth, and these methods can be grouped into two categories, supervised and unsupervised methods.

For supervised methods, the networks are directly trained on a large number of images with pixel-level ground truth depth labels [9]. However, the main factor inhibiting the use of supervised methods is the availability and acquisition of ground truth depths since collecting vast quantities of images with the corresponding ground truth depths is a non-trivial, costly, and labor-intensive process. To overcome the limitations of the aforementioned supervised methods, researchers employ unsupervised methods by casting the problem as an image reconstruction task and utilizing secondary information based on the underlying theory of epipolar constraints instead of requiring ground truth depths during training. In other words, current approaches employ geometry-based image reconstruction methods instead of ground truth depth labels to perform depth estimation in an unsupervised manner. As a result, the training data consist of either monocular video sequences [10] or rectified stereo image pairs only [11, 12] and do not include the difficulty to acquire ground truth depths. Although deep learning-based methods for monocular depth estimation learn to infer meaningful depth representations, these methods can suffer from undesirable artifacts due to scene ambiguity. As a result, different techniques have been used to improve depth representations. Supervised methods use conditional random fields (CRF) to refine depths [13]. For unsupervised methods, depth refinement can be done either by applying a non-trainable post-processing step such as in [12] when trained using rectified stereo image pairs or by incorporating another term in the loss function to handle occlusions if trained using monocular video sequences [10].

We follow the trend by using deep learning for monocular depth estimation with rectified stereo image pairs as training data and explore autoencoder networks for this task to enable the model to process local and global context information and multi-level representations more effectively [11, 12, 14]. However, autoencoder networks are limited in their ability to capture long-range context. In this paper, we propose to integrate convolutional long short-term memory layers [15] to a typical autoencoder network to address this issue since we observed that the training data for driving scenes are correlated. Moreover, unlike the previous approaches [12, 13] which consider training the model and refining the depth map as two separate tasks that have to be carried out sequentially, we show that it is possible to incorporate post-processing heuristics that refine depth maps as a trainable component of the framework. This approach enables the model to perform learning and refining of depth maps simultaneously. We train our model to synthesize depth from one image of the stereo pair, then reconstruct the other view by the synthesized depth. Specifically, we propose a technique that concurrently estimates the depth and refines the predicted depth map of a single RGB image in an unsupervised framework by training a Siamese convolutional long short-term memory network using the Adam optimization algorithm [16].

The Adam optimization algorithm [16] is a variant of the stochastic gradient descent algorithm designed for training deep neural networks and involves minimizing a loss function. It is a well-known deep learning optimization algorithm that achieves excellent results because it employs an adaptive learning rate method that computes the individual learning rates for each parameter. Adam estimates the first and second moments of the gradient to adapt the learning rate for each parameter of the deep neural network instead of merely maintaining a single learning rate for all parameters that is never altered during training.

Furthermore, since driving scenes such as the KITTI driving dataset [17] contain images with a very similar geometric layout, using only the spatial features extracted from these images may not be sufficient. We propose a model that incorporates the convolutional LSTM units [15, 18] at various stages in the network to enable it to not only extract spatial features but extract temporal features as well. ConvLSTM units are suitable for image processing applications and are very useful in modeling the spatio-temporal relationship of images. These units can memorize past information and reuse them in the current time step, that is, adding convLSTM units at various stages in the network enables it to reuse information from the previous images instead of solely relying on the information of current batch of images during training. ConvLSTM units are very useful in identifying spatio-temporal variations in the image data over time because of their capability to learn features at different spatial and temporal scales. Hence, capturing both the spatial and temporal features instead of solely relying on spatial features can significantly improve the model's performance.

Our contributions can be summarized as follows.

1) We propose a Siamese convolutional long short-term memory network for depth estimation, which employs convolutional and convolutional long short-term (convLSTM) layers to extract and combine local and global contextual information from rectified stereo image pairs and capture the long-range context of the scenes.
2) We integrate the Siamese convLSTM network into the unsupervised framework for monocular depth estimation using rectified stereo image pairs as training data and train the network to simultaneously learn and refine depth maps in an end-to-end manner by incorporating post-processing heuristics as a trainable component.
3) We introduce a modified image reconstruction loss function that optimizes these two tasks concurrently.
4) We present extensive experimental results on the KITTI 2015 driving dataset, and the results show that our proposed Siamese convLSTM network achieves excellent performance on monocular depth estimation, both quantitatively and qualitatively.

The remainder of the paper is arranged as follows. Section 2 discusses the related works. Section 3 describes in detail our proposed unsupervised framework for monocular depth estimation. Section 4 reports the experimental results on the standard benchmark dataset. Finally, Section 5 concludes the paper.

2. **Related Work.**

The pioneering work of Eigen et al. [9] demonstrated the superiority of deep learning-based models to learn depth from single RGB images, and these generated a lot of research interest to develop better-suited deep learning-based models for monocular depth estimation. In their work, Eigen et al. [9] directly trained two deep networks on a large number of images with pixel-level ground truth depth labels where the first network generates coarse global depth predictions, and these outputs are refined locally by the second network. Hua and Tian [13] also showed that the outputs of deep networks could be refined using conditional random fields (CRF) to obtain the final depth predictions. However, researchers explored unsupervised learning methods by framing monocular depth estimation as an image reconstruction task to forgo the need for ground truth depth labels since obtaining them is non-trivial, costly, and labor-intensive. Given the extensive works related to monocular depth estimation using deep learning methods, we limit the review to previous works that dealt with monocular depth estimation in an unsupervised manner. Specifically, we briefly review the state-of-the-art unsupervised methods based on the type of images used for training; 1) stereo sequences, or 2) monocular video sequences.

2.1. **Unsupervised learning of depth from stereo sequences.** For unsupervised learning of depth from stereo sequences, the training data consists of the left and right images and the known camera pose which is acquired by a calibrated stereo camera rig. With stereo sequences as training data, the disparity matching problem is simplified to a one-dimensional search. Given rectified stereo image pairs as inputs during the training phase, the model learns to generate depth estimates by tackling the monocular depth estimation problem as an image reconstruction task. The supervisory signal is generated by warping one view of a stereo pair into the other view using the predicted disparity maps. Garg et al. [11] presented a seminal work that used this methodology wherein they trained an encoder-decoder network or a stereopsis-based autoencoder for monocular depth estimation using rectified stereo image pairs to minimize the image reconstruction loss. Specifically, they introduced an unsupervised learning framework that employs a large stereo dataset without ground truth depth labels to train a network to predict depth from a single RGB image. Their framework is trainable in an end-to-end manner because they used a Taylor approximation to make their loss linear and fully-differentiable, but it resulted in a training objective that is more challenging to optimize.

On the other hand, Godard et al. [12] trained a convolutional encoder-decoder architecture to generate depth maps from both images in a stereo pair using a more robust image reconstruction loss function that incorporates a left-right consistency term. Primarily, their model used bilinear sampling to generate images from the predicted disparities and enforced a left-right consistency check between the predicted disparities for the left image and the right image. They also introduced a simple post-processing step to refine depth predictions but is a non-trainable component of their framework and is decoupled from the training process, making it optional. Inspired by the work of Godard et al. [12], Yusiong and Naval, Jr. [14] trained a multi-scale stacked encoder-decoder architecture that performs upsampling and autoencoding after each upsampling step to obtain better depth estimates at each scale. They also modified the image reconstruction loss function by applying the Charbonnier function instead of L_1 or L_2 to the terms in the training loss function.

2.2. **Unsupervised learning of depth from monocular video sequences.** Unsupervised learning of depth from monocular video sequences involves the use of video frames taken from an unconstrained single moving camera. Researchers who use this type of data have to deal with two critical issues. First, since the camera pose is unknown, the model has to perform pose estimation and depth estimation jointly. Second, the model has to deal with motion or moving objects. Zhou et al. [10] were the first to propose a framework that can predict depth and ego-motion from monocular video sequences. They trained a model consisting of two encoder-decoder networks to simultaneously predict depth from monocular video sequences by calculating a reconstruction loss between consecutive frames and the relative pose between these frames. This approach removes the need to use rectified stereo image pairs during training because of the pose estimation network. Also, they incorporated an explainability mask to deal with objects in motion and occluded regions. Wang et al. [19] proposed a framework that can infer depth from monocular videos using a differentiable direct visual odometry (DVO) instead of a pose estimation network to decrease the number of network parameters. They also introduced a simple depth normalization technique to handle the scale sensitivity problem wherein the technique normalizes the depth maps generated by the model before computing for the training loss. Mahjourian et al. [20] also improved the work of Zhou et al. [10] by combining the 2D photometric loss with their proposed 3D ICP-based loss to simultaneously

infer depth and camera motion from monocular videos. They also incorporated principled masks that exclude areas where no useful information exists to minimize training loss degradation.

Other researchers [21, 22, 23, 24] exploited additional geometry cues between structure and motion which they used in their models to improve the depth estimation accuracy, and these additional cues include edge, surface normal, and optical flow. Yang et al. [21] proposed a framework for joint depth, surface normal and edge learning from monocular videos using a "3D as-smooth-as-possible (3D-ASAP)" prior and enforcing consistency between these outputs. They also presented a similar approach in [22] wherein their model performs depth and surface normal estimation using an edge-aware depth-normal consistency term that enforces geometric consistency between the different scene projections, that is, the predicted depths must be compatible with the predicted normals. Moreover, Yin and Shi [23] and Zou et al. [24] concurrently proposed an unsupervised learning framework for jointly training monocular depth, ego-motion, and optical flow from monocular video sequences but with significant differences. Yin and Shi [23] introduced GeoNet, which consists of a rigid structure reconstructor for static scenes and a non-rigid motion localizer for dynamic scenes. They also proposed an adaptive geometry consistency loss to resolve occlusions and non-Lambertian surfaces effectively, thereby increasing the robustness of the model. In contrast, Zou et al. [24] presented DF-Net, a framework consisting of three major components, depth net, pose net, and flow net that performs depth estimation, relative camera pose estimation, and optical flow estimation, respectively. They also proposed a novel cross-consistency loss to enforce consistency within the valid regions between the rigid flow (derived from the depth and pose estimates) and the estimated optical flow.

Hence, prior works reveal that the supervised approaches to training monocular depth estimation models require the use of ground truth depth labels but collecting datasets with ground truth depth labels is a challenging task in itself. An alternative approach is to employ unsupervised learning methods to train monocular depth estimation models using either monocular video sequences or rectified stereo image pairs as training data. Among these two unsupervised approaches, using rectified stereo images pairs is more attractive than using monocular video sequences because models trained with this type of training data perform much better than models trained with monocular video sequences and it simplifies the monocular depth estimation problem from a two-dimensional stereo correspondence problem to a one-dimensional search problem. Also, training monocular depth estimation models with rectified stereo image pairs involves certain design choices, such as pre-training the network, formulating the image reconstruction loss function, designing the network architecture, and adding a post-processing step to refine depth maps [11, 12, 14]. These design choices affect the quality of the predicted depth maps. For instance, training a model with a simple image reconstruction loss function can already generate low-quality depth maps [11] while pre-training the network and introducing a more sophisticated image reconstruction loss function can lead to better results [12]. Other ways to produce superior results include adding a post-processing step at test time [12] to deal with visual artifacts and blurred boundaries and designing a very deep network architecture [14].

In this work, we address past design issues to obtain significant improvements in monocular depth estimation involving rectified stereo image pairs as training data. First, instead of solely relying on convolutional layers to extract spatial features, we introduce convolutional LSTM layers to our convolutional encoder-decoder architecture as a way for the model to capture spatial and temporal features. This design choice is crucial because driving scenes contain images with a very similar geometric layout and using only the

spatial features extracted from these images may not be sufficient, and by introducing convolutional LSTM layers to our network, we can obtain better results than a very deep network architecture [14]. Second, we incorporate post-processing heuristics as a trainable component of our model to perform depth refinement during training rather than doing it at test time like in [12]. Lastly, we resolve the issue of decoupling depth estimation from depth refinement by employing a Siamese network architecture to handle these two tasks simultaneously. In essence, our contributions rely on altering the essential building blocks of the convolutional encoder-decoder network architecture and incorporating a post-processing step as a trainable component of our model to produce better depth estimates.

3. **Proposed Approach.** This section describes in detail our proposed framework. Essentially, our method jointly learns to predict and refine disparity maps in an unsupervised manner using rectified stereo image pairs (I_L, I_R) as training data. Figure 1 provides an overview of our framework and its components. At the core of our approach is a Siamese network architecture, which has two convLSTM encoder-decoder networks that share weight parameters. More precisely, one network deals with the image reconstruction task while the other network refines the predicted disparity maps.

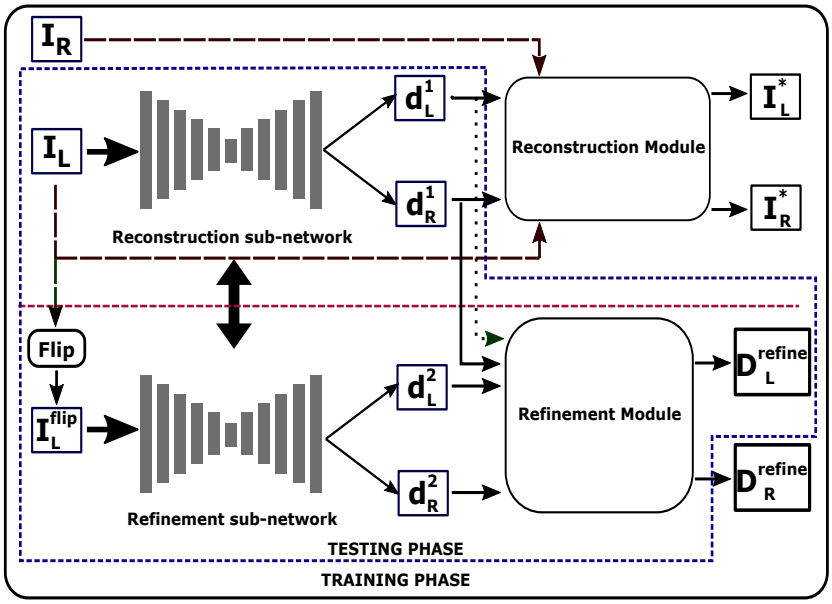

FIGURE 1. Our proposed unsupervised learning framework for joint learning and refining of depth maps using rectified stereo image pairs (I_L, I_R) as training data. The reconstruction sub-network receives the original left images I_L as inputs and generates a pair of disparity maps (d_L^1, d_R^1). The pair of disparity maps and the original right images I_R are used by the reconstruction module to reconstruct the images I_R^* and I_L^*. The refinement sub-network receives the horizontally flipped version of the left input images I_L^{flip} and predicts a pair of disparity maps (d_L^2, d_R^2). The refinement module produces a pair of refined disparity maps, $\left(D_L^{refine}, D_R^{refine}\right)$ using all the predicted disparity maps.

3.1. **Network architecture.** Our proposed framework adopts a Siamese network with the U-net model [25] as the autoencoder, but we modified its structure by adding convolutional long short-term memory layers [15]. Specifically, the skip connections in the

original U-Net architecture are replaced by the convLSTM 'skip' connections, and we added two convolutional LSTM layers before the decoder section of the network. Table 1 details the various layers in the network while Figure 2 depicts the convLSTM network architecture.

TABLE 1. The layers of our convLSTM encoder-decoder architecture. **conv** and **iconv** refer to a convolutional layer with an ELU activation function, **pool** refers to the maxpooling layer, **deconv** refers to a deconvolutional layer with an ELU activation function, **convlstm** refers to a convolutional LSTM layer with a leaky RELU activation function, and **disp** is the predicted disparity map generated using a convolutional layer with a sigmoid activation function. **k** is the kernel size, **s** is the stride, **ch** is the number of output channels for each layer, **dim** is the downscaling factor for each layer relative to the input image dimension, **input** corresponds to the inputs of each layer, where + means concatenation and $*$ means a $2\times$ nearest-neighbor upsampling of the layer.

Encoder					
Layer	**k**	**s**	**ch**	**dim**	**input**
conv1a	3	1	32	1	left
conv1b	3	1	32	1	conv1a
convlstm_skip1	3	1	32	1	conv1b
pool1	2	2	–	2	conv1b
conv2a	3	1	64	2	pool1
conv2b	3	1	64	2	conv2a
convlstm_skip2	3	1	64	2	conv2b
pool2	2	2	–	4	conv2b
conv3a	3	1	128	4	pool2
conv3b	3	1	128	4	conv3a
convlstm_skip3	3	1	128	4	conv3b
pool3	2	2	–	8	conv3b
conv4a	3	1	256	8	pool3
conv4b	3	1	256	8	conv4a
convlstm_skip4	3	1	256	8	conv4b
pool4	2	2	–	16	conv4b
conv5a	3	1	512	16	pool4
conv5b	3	1	512	16	conv5a
convlstm_skip5	3	1	512	16	conv5b
pool5	2	2	–	32	conv5b
conv6a	3	1	1024	32	pool5
conv6b	3	1	1024	32	conv6a
convlstm7a	3	1	512	32	conv6b
convlstm7b	3	1	512	32	convlstm7a

Decoder					
Layer	**k**	**s**	**ch**	**dim**	**input**
deconv5	3	2	512	16	convlstm7b
iconv5b	3	1	512	16	deconv5+convlstm_skip5
iconv5a	3	1	512	16	iconv5b
deconv4	3	2	256	8	iconv5a
iconv4b	3	1	256	8	deconv4+convlstm_skip4
iconv4a	3	1	256	8	iconv4b
disp4	3	1	2	8	iconv4a
deconv3	3	2	128	4	iconv4a
iconv3b	3	1	128	4	deconv3+convlstm_skip3+disp4*
iconv3a	3	1	128	4	iconv3b
disp3	3	1	2	4	iconv3a
deconv2	3	2	64	2	iconv3a
iconv2b	3	1	64	2	deconv2+convlstm_skip2+disp3*
iconv2a	3	1	64	2	iconv2b
disp2	3	1	2	2	iconv2a
deconv1	3	2	32	1	iconv2a
iconv1b	3	1	32	1	deconv1+convlstm_skip1+disp2*
iconv1a	3	1	32	1	iconv1b
disp1	3	1	2	1	iconv1a

The convLSTM network consists of convolutional layers, maxpooling layers, deconvolutional layers, and convolutional LSTM layers. The maxpooling layers have a kernel size of 2×2 and a stride of 2 while all the other layers use the same kernel of size 3×3 and a stride of 1. An exponential linear unit (ELU) is applied to the output of the convolutional and deconvolutional layers except for the last convolutional layers (that is, the disp layers) where the sigmoid activation function is used to generate the disparity maps. On the other hand, the convolutional LSTM layers use leaky RELU.

The Siamese architecture consists of two identical convLSTM networks that share weight parameters. The first network, referred to as the reconstruction sub-network, receives the original left images I_L as inputs while the second network, referred to as the refinement sub-network, receives the horizontally flipped version of the left input images I_L^{flip} and each sub-network generates a pair of disparity maps, (d_L^1, d_R^1) and (d_L^2, d_R^2), respectively.

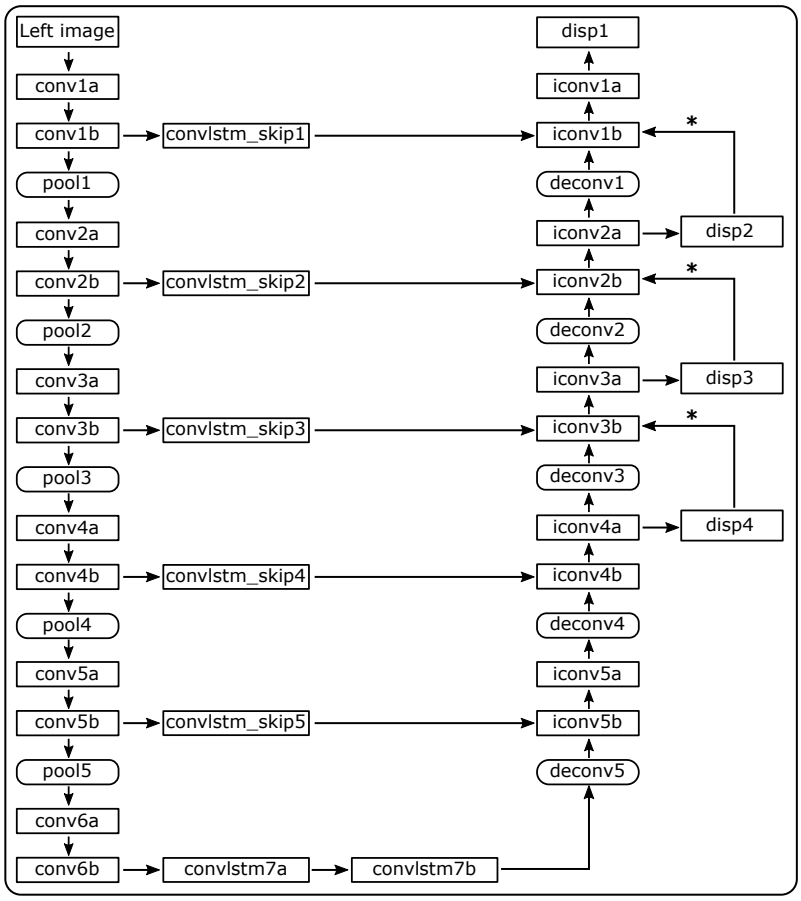

FIGURE 2. The convLSTM network architecture

The module in the reconstruction sub-network accepts two pairs of inputs (I_L, d_R^1) and (I_R, d_L^1) to reconstruct I_R^* and I_L^*, respectively using the sampler from the spatial transformer network (STN) [26] that performs bilinear interpolation. On the other hand, the module in the refinement sub-network is the post-processing heuristic described in [12], which has been incorporated as a trainable component of our framework. It processes the disparity maps (d_L^1, d_R^1) and (d_L^2, d_R^2), and outputs a pair of refined disparity maps $\left(D_L^{refine}, D_R^{refine}\right)$.

The refinement module employs a disparity flip operator to perform the horizontal flip operation on the disparity map (d_L^2, d_R^2) to produce $\left(d_L^{flip}, d_R^{flip}\right)$. To obtain the refined left disparity maps D_L^{refine}, the module applies a pixel-wise mean operation on $\left(d_L^1 \text{ and } d_L^{flip}\right)$ and then removes the disparity ramps on the boundary pixels. The same steps are taken to fuse $\left(d_R^1, d_R^{flip}\right)$ to produce the refined right disparity map D_R^{refine}. This module significantly helps in improving performance since the model can deal with visual artifacts and blurred boundaries more effectively.

Our network produces a pair of refined disparity maps, $\left(D_L^{refine}, D_R^{refine}\right)$ from the left image I_L at four different scales; however, only the refined left disparity map D_L^{refine} with scale equals 1 is useful at test time.

3.2. **Loss function.** Current training loss functions for monocular depth estimation with rectified stereo image pairs as training data were designed for depth estimation only

and did not consider disparity refinement as a trainable component of their framework. Our model design enables us to modify existing training loss functions for joint depth estimation and refinement using a Siamese convolutional long short-term memory network. The total training loss L is the sum of the four different scales, $L = \sum_{s=1}^{4} L_s$. The loss at each scale L_s, as defined in (1), is the weighted sum of four terms: appearance dissimilarity, edge-aware disparity smoothness, left-right consistency, and maximum depth heuristic.

$$L_s = \alpha_{app} L_{app} + \alpha_{smooth} L_{smooth} + \alpha_{lr} L_{lr} + \alpha_{mdh} L_{mdh} \tag{1}$$
$$L_{app} = L_{app}^{left} + L_{app}^{right} \tag{2}$$
$$L_{smooth} = L_{smooth}^{left} + L_{smooth}^{right} \tag{3}$$
$$L_{lr} = L_{lr}^{left} + L_{lr}^{right} \tag{4}$$
$$L_{mdh} = L_{mdh}^{left} + L_{mdh}^{right} \tag{5}$$

where L_{app} is the appearance dissimilarity term, L_{smooth} is the edge-aware disparity smoothness term, L_{lr} is the left-right consistency term, and L_{mdh} is the maximum depth heuristic term. Since our training data consists of rectified stereo image pairs, the training loss function takes into account the left and right images where each component is in terms of the left image $\left(L_{app}^{left}, L_{smooth}^{left}, L_{lr}^{left}, L_{mdh}^{left}\right)$ and the right image $\left(L_{app}^{right}, L_{smooth}^{right}, L_{lr}^{right}, L_{mdh}^{right}\right)$.

The appearance dissimilarity term, as defined in (6), is a linear combination of the single-scale structural SIMilarity (SSIM) [27] term and the L_1 photometric term. This term measures the quality of the synthesized target image by minimizing the pixel-level dissimilarity between the target image I and the synthesized target image I^*. This term is also widely used in previous studies [12, 14, 19, 20, 23].

$$\begin{aligned} L_{app}^{left} &= \frac{1}{N} \sum_{x,y} \omega \frac{1 - SSIM\left(I_L(x,y), I_L^*(x,y)\right)}{2} + (1-\omega) \left\| I_L(x,y) - I_L^*(x,y) \right\| \\ L_{app}^{right} &= \frac{1}{N} \sum_{x,y} \omega \frac{1 - SSIM\left(I_R(x,y), I_R^*(x,y)\right)}{2} + (1-\omega) \left\| I_R(x,y) - I_R^*(x,y) \right\| \end{aligned} \tag{6}$$

with a 3×3 box filter for the SSIM term and ω is set to 0.85 similar to [12, 14].

The edge-aware disparity smoothness term regularizes the predicted disparities in spatially similar regions to enforce the assumption that the predicted disparities must be locally smooth but allows for sharpness at the edges. As shown in (7) and described in [12, 20], this term penalizes large disparities between neighboring pixels except if there are strong intensity gradients in the input image. We modified this term to include the refined disparity maps, D_L^{refine} and D_R^{refine}, in the training loss computation. This simple modification is necessary since our Siamese network generates two pairs of predicted disparity maps and are refined in the refinement module to obtain the final output of the network, a pair of refined disparity maps, $\left(D_L^{refine}, D_R^{refine}\right)$.

$$\begin{aligned} L_{smooth}^{left} = \frac{1}{N} \sum_{x,y} \Big(&\Big(\left|\partial_x d_L^2(x,y)\right| e^{-\left|\partial_x I_L^{flip}(x,y)\right|} + \left|\partial_y d_L^2(x,y)\right| e^{-\left|\partial_y I_L^{flip}(x,y)\right|} \Big) \\ &+ \Big(\left|\partial_x D_L^{refine}(x,y)\right| e^{-\left|\partial_x I_L(x,y)\right|} + \left|\partial_y D_L^{refine}(x,y)\right| e^{-\left|\partial_y I_L(x,y)\right|} \Big) \Big) \\ L_{smooth}^{right} = \frac{1}{N} \sum_{x,y} \Big(&\Big(\left|\partial_x d_R^2(x,y)\right| e^{-\left|\partial_x I_R^{flip}(x,y)\right|} + \left|\partial_y d_R^2(x,y)\right| e^{-\left|\partial_y I_R^{flip}(x,y)\right|} \Big) \\ &+ \Big(\left|\partial_x D_R^{refine}(x,y)\right| e^{-\left|\partial_x I_R(x,y)\right|} + \left|\partial_y D_R^{refine}(x,y)\right| e^{-\left|\partial_y I_R(x,y)\right|} \Big) \Big) \end{aligned} \tag{7}$$

As described in [12, 14], the left-right consistency term enforces consistency between the left and right disparities as defined in (8). Like in (7), we modified this term to take into account the refined disparity maps, D_L^{refine} and D_R^{refine}, in computing the training loss since our Siamese network predicts two pairs of predicted disparity maps and they are refined in the refinement module to obtain the final output of the network, a pair of refined disparity maps, $\left(D_L^{refine}, D_R^{refine}\right)$.

$$\begin{aligned} L_{lr}^{left} &= \frac{1}{N}\sum_{x,y}\left|D_L^{refine}(x,y) - \left(d_R^1\left(x - d_L^1(x,y), y\right)\right)\right| \\ L_{lr}^{right} &= \frac{1}{N}\sum_{x,y}\left|D_R^{refine}(x,y) - \left(d_L^1\left(x + d_R^1(x,y), y\right)\right)\right| \end{aligned} \tag{8}$$

To improve the quality of the refined disparity maps and enable the model to deal with texture-less regions and minimize the ghosting effect, we also adopted the maximum depth heuristic term similar to [28] and it is defined in (9).

$$\begin{aligned} L_{mdh}^{left} &= \frac{1}{N}\sum_{x,y}\left|D_L^{refine}(x,y)\right| \\ L_{mdh}^{right} &= \frac{1}{N}\sum_{x,y}\left|D_R^{refine}(x,y)\right| \end{aligned} \tag{9}$$

4. **Experiments.** In this section, we briefly describe the datasets and evaluation metrics used in our experiments and the implementation details of our network architecture and training protocols. Afterward, we present an extensive evaluation of our proposed approach for unsupervised monocular depth estimation on the publicly available KITTI driving dataset and report both the quantitative and qualitative results to highlight the effectiveness of our method. Also, we conducted an ablation study to validate the effect on the performance when incorporating convolutional long short-term memory layers in our proposed framework.

4.1. **Datasets and evaluation metrics.** For all the experiment setups, we used the KITTI 2015 driving dataset [17] that consists of rectified stereo image pairs from 61 urban scenes to train and evaluate our proposed approach since this dataset is the standard benchmark for depth estimation. Like in the previous works, we use the Eigen split [9] for training and testing, and this split contains 22,600 rectified stereo image pairs for training and 697 rectified stereo image pairs for testing. The train set consists of rectified stereo images pairs extracted from 32 out of the 61 scenes, while the test set was acquired from the remaining 29 scenes where the ground truth depth maps can be obtained by using the Velodyne data to project 3D points into the left images of the rectified stereo image pairs. On the other hand, for the experiments involving pre-training the models, we used the Cityscapes [29] dataset, which is a more recent urban driving dataset consisting of high-resolution images with many dynamic scenes. The training set consists of 22,973 rectified stereo image pairs.

To quantitatively evaluate our proposed approach, we apply the same standard evaluation metrics used in the previous works [9, 12, 14]. The standard evaluation metrics for depth estimation measure the average errors, where lower values are better and accuracy scores, where higher values are preferred.

4.2. **Implementation details.** We implemented our proposed Siamese convLSTM network for monocular depth estimation using Tensorflow [30]. We utilized a single Nvidia

GTX 1080 Ti GPU with 11 GB of memory to train the network from scratch by randomly initializing the weight parameters. We used Adam [16] optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$ to minimize the training loss where the training loss is the mean of the losses from different scales. The network training is for 50 epochs with a mini-batch size of 4, and an initial learning rate of $\lambda = 10^{-4}$. The initial learning rate is used for the first 30 epochs, and afterward, it is reduced by half every 10 epochs. We set the weights in the loss function to $\alpha_{app} = 1.0$, $\alpha_{smooth} = 0.1/2^s$, $\alpha_{lr} = 1.0$, and $\alpha_{mdh} = 0.01$, where s is the output scale ranging from 0-3 since the network generates depth maps at four different scales. We resized the rectified stereo image pairs to a resolution of 512×256 for training, but at test time the network can generate depth maps from single RGB images with different dimensions. To prevent overfitting, we applied L_2 regularization on all the weight parameters by adding a small constant, 0.00001, and performed online data augmentation, which includes horizontal flip and gamma, brightness, and color changes. The experimental settings are adopted from prior works since these settings have been found to produce excellent results for monocular depth estimation [12, 14].

4.3. **Results and discussion.** We evaluate the monocular depth estimation performance of our proposed approach on the 697 images from the test split of Eigen et al. [9] using the Velodyne ground truth data. We compare our results with the different methods that used the Eigen train/test split by directly using the reported results in their published works. Similar to the previous studies, we also conducted experiments where we pre-train our network on the larger Cityscapes dataset and then fine-tune on the KITTI dataset to obtain better results.

Table 2 shows the effectiveness of our proposed approach. We report separate results using the standard evaluation metrics for the two distance ranges: 1-50 meters and 0-80 meters. The quantitative results show that our method achieved excellent performance compared to the other state-of-the-art methods in every metrics category for depth estimation error and accuracy and these results indicate that the depth refinement module helps improve depth estimation. Moreover, our model generates a consistent scene layout with fewer outlier depths since we achieved excellent results in the square relative difference (SRD) and linear root mean squared error (RMSE linear) metrics. These metrics are sensitive to large depth errors and obtaining good results under these metrics is an excellent indicator that the model generates depth maps with few outliers.

We can also observe that methods using monocular video sequences as training data tend to perform worse than the models using rectified stereo image pairs and this can be attributed to the errors from the camera pose estimation whereas when using rectified stereo image pairs as training data, camera pose is not an issue because it is given. We can clearly see that our proposed approach which uses rectified stereo image pairs is substantially better than all models trained on monocular video sequences.

Moreover, our models that are pre-trained on the Cityscapes dataset and fine-tuned on the KITTI dataset significantly improve the results compared to our models trained on the KITTI dataset only. These results show that pre-training adds robustness to the model and leads to better generalization since the model can capture better features and learn more complex dependencies between the different parameters. Most importantly, these results suggest that our proposed approach can leverage different data sources with similar characteristics to obtain a more accurate depth estimation.

Another valuable insight is that distance plays a significant role in obtaining accurate depth estimates where a larger range causes the depth accuracy to drift noticeably since the depth cues become less reliable.

TABLE 2. Monocular depth estimation results on the KITTI test set using the Eigen split. For training, K means trained on the KITTI dataset, and CS+K means pre-trained on Cityscapes and fine-tuned on KITTI. For the training protocol, Depth means the methods used ground truth depths at training time, Mono means the methods used monocular video sequences for training, and Stereo means the methods used rectified stereo image pairs for training. The **bold** values indicate the best results.

Method	Training Dataset	Train	Error Metric (Lower is better)				Accuracy Metric (Higher is better)		
			ARD	*SRD*	*RMSE (linear)*	*RMSE (log)*	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Depth range: 0-80 meters									
Eigen et al. Coarse* [9]	K	Depth	0.194	1.531	7.216	0.273	0.679	0.897	0.967
Eigen et al. Coarse+Fine* [9]	K	Depth	0.190	1.515	7.156	0.270	0.692	0.899	0.967
Zhou et al. [10]	K	Mono	0.208	1.768	6.856	0.283	0.678	0.885	0.957
DDVO [19]	K	Mono	0.151	1.257	5.583	0.228	0.810	0.936	0.974
GeoNet [23]	K	Mono	0.155	1.296	5.857	0.233	0.793	0.931	0.973
Mahjourian et al. [20]	K	Mono	0.163	1.240	6.220	0.250	0.762	0.916	0.968
LEGO [21]	K	Mono	0.162	1.352	6.276	0.252	–	–	–
Yang et al. [22]	K	Mono	0.182	1.481	6.501	0.267	0.725	0.906	0.963
DF-Net [24]	K	Mono	0.150	1.124	5.507	0.223	0.806	0.933	0.973
Godard et al. [12]	K	Stereo	0.148	1.344	5.927	0.247	0.803	0.922	0.964
AsiANet [14]	K	Stereo	0.145	1.349	5.909	0.230	0.824	0.936	0.970
Ours	K	Stereo	**0.131**	**1.075**	**5.318**	**0.212**	**0.846**	**0.947**	**0.976**
Zhou et al. [10]	CS + K	Mono	0.198	1.836	6.565	0.275	0.718	0.901	0.960
DDVO [19]	CS + K	Mono	0.148	1.187	5.496	0.226	0.812	0.938	0.975
GeoNet [23]	CS + K	Mono	0.153	1.328	5.737	0.232	0.802	0.934	0.972
Mahjourian et al. [20]	CS + K	Mono	0.159	1.231	5.912	0.243	0.784	0.923	0.970
LEGO [21]	CS + K	Mono	0.159	1.345	6.254	0.247	–	–	–
DF-Net [24]	CS + K	Mono	0.146	1.182	5.215	0.213	0.818	0.943	0.978
Godard et al. [12]	CS + K	Stereo	0.124	1.076	5.311	0.219	0.847	0.942	0.973
AsiANet [14]	CS + K	Stereo	0.128	1.161	5.470	0.213	0.858	0.947	0.974
Ours	CS + K	Stereo	**0.120**	**1.012**	**5.106**	**0.200**	**0.867**	**0.954**	**0.979**
Depth range: 1-50 meters									
Zhou et al. [10]	K	Mono	0.201	1.391	5.181	0.264	0.696	0.900	0.966
GeoNet [23]	K	Mono	0.147	0.936	4.348	0.218	0.810	0.941	0.977
Mahjourian et al. [20]	K	Mono	0.155	0.927	4.549	0.231	0.781	0.931	0.975
Garg et al. [11] L12 Aug 8x	K	Stereo	0.169	1.080	5.104	0.273	0.740	0.904	0.962
Godard et al. [12]	K	Stereo	0.140	0.976	4.471	0.232	0.818	0.931	0.969
AsiANet [14]	K	Stereo	0.122	0.786	4.014	0.198	0.864	0.953	0.978
Ours	K	Stereo	**0.109**	**0.634**	**3.636**	**0.182**	**0.882**	**0.962**	**0.983**
Zhou et al. [10]	CS + K	Mono	0.190	1.436	4.975	0.258	0.735	0.915	0.968
Mahjourian et al. [20]	CS + K	Mono	0.151	0.949	4.383	0.227	0.802	0.935	0.974
Godard et al. [12]	CS + K	Stereo	0.117	0.762	3.972	0.206	0.860	0.948	0.976
AsiANet [14]	CS + K	Stereo	0.107	0.663	3.717	0.184	0.893	0.960	0.981
Ours	CS + K	Stereo	**0.100**	**0.589**	**3.473**	**0.172**	**0.900**	**0.966**	**0.985**

The qualitative results are depicted in Figure 3. The results show that our model can capture the general 3D scene layout, preserve scene structures, and suppress the visible artifacts around the image boundary and occluded regions. As a result, we can visually see that the predicted depth maps are smooth in planes and have sharp edges on the object boundaries. In other words, the model can generate smoother and more detailed depth maps without border artifacts around the image boundary, preserve the structure in depth boundaries, and recover the underlying geometry of objects even for objects with thin structures such as trees, poles, and traffic signs.

These results demonstrate that jointly performing depth estimation and refinement using our proposed unsupervised learning framework leads to better performance.

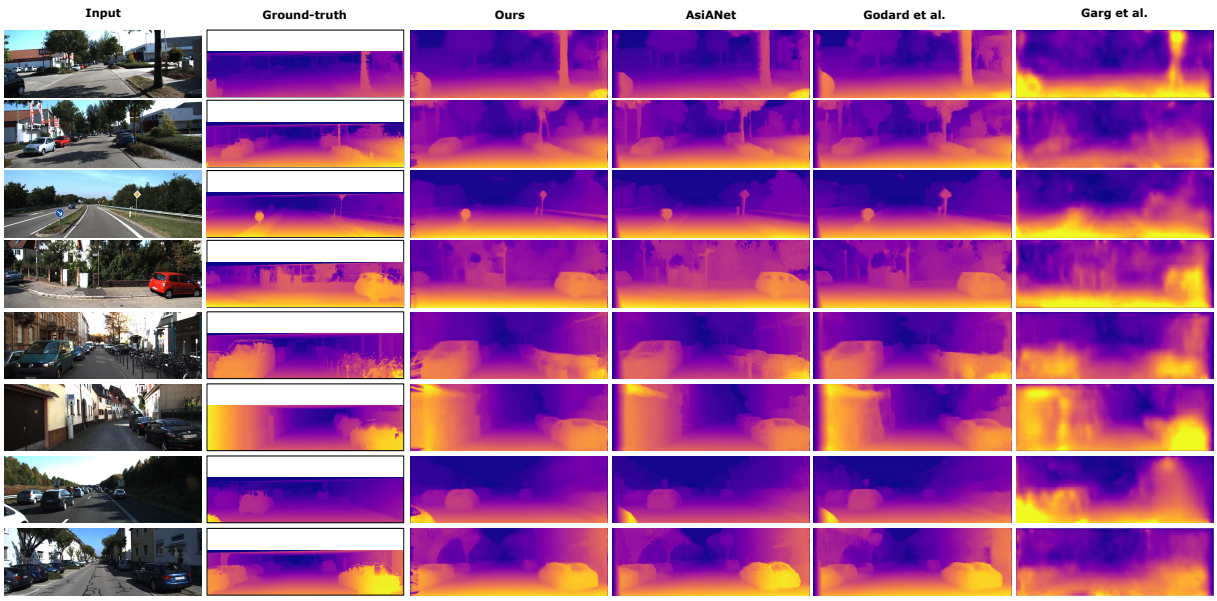

FIGURE 3. (color online) Qualitative results on the KITTI test set. We compared our results with the results of Garg et al. [11], Godard et al. [12], and AsiANet [14]. Our method generates depth maps without border artifacts around the image boundary, minimizes the ghosting effect around the object boundaries, and recovers the 3D scene layout. The ground truth depth maps are interpolated for visualization purpose only.

4.4. **Architectural analysis.** We also conducted an ablation study by removing the convLSTM layers in our proposed network and used the test set of the Eigen split to evaluate the performance. This setup aims to establish the importance of incorporating convLSTM layers to our proposed network. We compared three different models for monocular depth estimation, a non-Siamese U-net architecture, a Siamese U-net architecture, and our proposed Siamese convLSTM architecture. The non-Siamese U-net architecture uses a framework similar to the previous works [11, 12, 14], while the Siamese U-net architecture uses our proposed framework but adopts the original U-net architecture as the autoencoder. We also demonstrate the relevance of the refinement module in improving the quality of the predicted depth maps by comparing the non-Siamese U-net architecture (baseline) to the Siamese U-net architecture.

As shown in Table 3, using our Siamese framework to jointly perform estimation and refinement of depth maps with rectified stereo image pairs as training data leads to significant improvements compared to the non-Siamese framework in every metrics category for depth estimation error and accuracy. These results indicate that the refinement module is effective in improving the quality of the predicted depth maps. Moreover, comparing the results between the Siamese U-net and Siamese convLSTM reveals that adding convLSTM layers to the U-net model improves the model's performance significantly as these layers provide a way to capture long-range contextual information.

Hence, we can infer that the model's performance can be improved by introducing convLSTM layers at various stages in the network to extract more features and refining the predicted depth maps during training to enhance the quality of the depth maps.

5. **Conclusions.** In this paper, we presented an unsupervised learning framework for monocular depth estimation that jointly performs estimation and refinement of depth maps using rectified stereo image pairs as training data. In particular, we proposed to

TABLE 3. Architectural analysis. Results on the KITTI test set using the Eigen split. For training, K means trained on the KITTI dataset and Stereo means the methods used rectified stereo image pairs for training. The **bold** values indicate the best results.

Method	Training Dataset	Train	Error Metric (Lower is better)				Accuracy Metric (Higher is better)		
			ARD	*SRD*	*RMSE (linear)*	*RMSE (log)*	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Depth range: 0-80 meters									
Non-Siamese U-net (baseline)	K	Stereo	0.150	1.438	5.964	0.237	0.817	0.932	0.968
Siamese U-net	K	Stereo	0.139	1.186	5.569	0.221	0.830	0.941	0.974
Siamese convLSTM (ours)	K	Stereo	**0.131**	**1.075**	**5.318**	**0.212**	**0.846**	**0.947**	**0.976**
Depth range: 1-50 meters									
Non-Siamese U-net (baseline)	K	Stereo	0.126	0.833	4.067	0.204	0.857	0.950	0.977
Siamese U-net	K	Stereo	0.116	0.689	3.796	0.190	0.869	0.958	0.982
Siamese convLSTM (ours)	K	Stereo	**0.109**	**0.634**	**3.636**	**0.182**	**0.882**	**0.962**	**0.983**

tackle the monocular depth estimation problem as an image reconstruction task by using rectified stereo image pairs to train a deep learning model based on a unique Siamese convolutional long short-term memory network architecture to simultaneously learn and refine depth from a single RGB image. Our proposed approach exploits the ability of convolutional long short-term memory layers to reason long-range contextual information, incorporates a post-processing step as a trainable component of our model, and utilizes a modified image reconstruction loss function that optimizes these two tasks concurrently to improve performance and to produce a more accurate depth map for an image.

We conducted extensive experiments using the KITTI 2015 driving dataset to demonstrate quantitatively and qualitatively that our proposed Siamese convLSTM network performs better compared to the previous state-of-the-art unsupervised methods. Specifically, the quantitative results using the standard metrics show that the proposed approach substantially improves depth estimation performance, even outperforming the previous state-of-the-art unsupervised methods while the qualitative results reveal that the proposed approach generates more detailed and accurate depth maps of the scenes.

Although we incorporated a refinement module in our network, we did not explicitly handle occlusions during training, such as using occlusion masks for pixels that are occluded in both images. Hence, there are still some visible artifacts in the refined disparity maps. Moreover, since our framework requires a set of rectified stereo image pairs as inputs during training, we cannot use any training data consisting only of RGB or RGB-D images to train our Siamese convLSTM network.

In the future work, we can introduce high-level cues such as semantic segmentation to our model and validate the possible improvement on its performance. We can also hopefully improve the quality of the predicted depth maps by using multi-baseline stereo images as training data, which basically involves more than two views to estimate depth. With multi-baseline stereo images, the additional information can be exploited to estimate depth in occluded areas. Another area of interest is to redesign our Siamese convLSTM network for embedded systems to allow for real-time monocular depth estimation. Finally, we can extend our model to deal with monocular video sequences or random Internet videos with unknown camera parameters as training data.

Acknowledgment. This work has been supported by the Department of Science and Technology under the Engineering Research and Development for Technology Program.

REFERENCES

[1] C. Hazirbas, L. Ma, C. Domokos and D. Cremers, FuseNet: Incorporating depth into semantic segmentation via fusion-based CNN architecture, *Proc. of Asian Conference on Computer Vision (ACCV)*, 2016.

[2] Q. Wu, G. Xu, M. Li, L. Chen, X. Zhang and J. Xie, Human pose estimation method based on single depth image, *IET Computer Vision*, vol.12, no.6, pp.919-924, 2018.

[3] J. Xiao and C. Cai, Contour detection combined with depth information, *The 9th International Symposium on Multispectral Image Processing and Pattern Recognition (MIPPR2015)*, Enshi, China, 2015.

[4] Y. Xia, W. Xu, L. Zhang, X. Shi and K. Mao, Integrating 3D structure into traffic scene understanding with RGB-D data, *Neurocomputing*, vol.151, pp.700-709, 2015.

[5] A. Grigorev, F. Jiang, S. Rho, W. J. Sori, S. Liu and S. Sai, Depth estimation from single monocular images using deep hybrid network, *Multimedia Tools and Applications*, vol.76, no.18, pp.18585-18604, 2017.

[6] Y. Omae, M. Mori, T. Akiduki and H. Takahashi, A novel deep learning optimization algorithm for human motions anomaly detection, *International Journal of Innovative Computing, Information and Control*, vol.15, no.1, pp.199-208, 2019.

[7] K. S. Sim and F. Sammani, Deep convolutional networks for magnification of DICOM brain images, *International Journal of Innovative Computing, Information and Control*, vol.15, no.2, pp.725-739, 2019.

[8] X. Wang, Y. Sheng, H. Deng and Z. Zhao, CharCNN-SVM for Chinese text datasets sentiment classification with data augmentation, *International Journal of Innovative Computing, Information and Control*, vol.15, no.1, pp.277-246, 2019.

[9] D. Eigen, C. Puhrsch and R. Fergus, Depth map prediction from a single image using a multi-scale deep network, *Advances in Neural Information Processing Systems (NIPS)*, pp.2366-2374, 2014.

[10] T. Zhou, M. Brown, N. Snavely and D. G. Lowe, Unsupervised learning of depth and ego-motion from video, *Proc. of Conference on Computer Vision and Pattern Recognition*, pp.6612-6619, 2017.

[11] R. Garg, G. Carniero and I. Reid, Unsupervised CNN for single view depth estimation: Geometry to the rescue, *Proc. of European Conference on Computer Vision (ECCV)*, pp.740-756, 2016.

[12] C. Godard, O. Mac Aodha and G. J. Brostow, Unsupervised monocular depth estimation with left-right consistency, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.6602-6611, 2017.

[13] Y. Hua and H. Tian, Depth estimation with convolutional conditional random field network, *Neurocomputing*, vol.214, pp.546-554, 2016.

[14] J. P. Yusiong and P. C. Naval, Jr., AsiANet: Autoencoders in autoencoder for unsupervised monocular depth estimation, *Proc. of IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp.443-451, 2019.

[15] X. Shi, Z. Chen, H. Wang and D. Yeung, Convolutional LSTM network: A machine learning approach for precipitation nowcasting, *Advances in Neural Information Processing Systems (NIPS)*, pp.802-810, 2015.

[16] D. Kingma and J. Ba, Adam: A method for stochastic optimization, *Proc. of International Conference on Learning Representations (ICLR)*, 2015.

[17] Geiger, P. Lenz and R. Urtasun, Are we ready for autonomous driving? The KITTI vision benchmark suite, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3354-3361, 2012.

[18] S. Hochreiter and J. Schmidhuber, Long short-term memory, *Neural Computation*, vol.9, no.8, pp.1735-1780, 1997.

[19] C. Wang, J. M. Buenaposada, R. Zhu and S. Lucey, Learning depth from monocular videos using direct methods, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.2022-2030, 2018.

[20] R. Mahjourian, M. Wicke and A. Angelova, Unsupervised learning of depth and ego-motion from monocular video using 3D geometric constraints, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.5667-5675, 2018.

[21] Z. Yang, P. Wang, Y. Wang, W. Xu and R. Nevatia, LEGO: Learning edge with geometry all at once by watching videos, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.225-234, 2018.

[22] Z. Yang, P. Wang, W. Xu, L. Zhao and R. Nevatia, Unsupervised learning of geometry from videos with edge-aware depth-normal consistency, *Proc. of 32nd AAAI Conference on Artificial Intelligence (AAAI-18)*, 2018.

[23] Z. Yin and J. Shi, GeoNet: Unsupervised learning of dense depth, optical flow and camera pose, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1983-1992, 2018.

[24] Y. Zou, Z. Lou and J. Huang, DF-Net: Unsupervised joint learning of depth and flow using cross-consistency, *Proc. of European Conference on Computer Vision (ECCV)*, 2018.

[25] O. Ronneberger, P. Fischer and T. Brox, U-Net: Convolutional networks for biomedical image segmentation, *Proc. of International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015.

[26] M. Jaderberg, K. Simonyan, A. Zisserman and K. Kavukcuoglu, Spatial transformer networks, *Proc. of Annual Conference on Neural Information Processing Systems (NIPS)*, pp.2017-2025, 2015.

[27] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, Image quality assessment: From error measurement to structural similarity, *IEEE Trans. Image Processing*, vol.13, no.4, pp.600-612, 2004.

[28] Y. Zhong, Y. Dai and H. Li, Self-supervised learning for stereo matching with self-improving ability, *arXiv Preprint*, arXiv:1709.00930, 2017.

[29] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth and B. Schiele, The cityscapes dataset for semantic urban scene understanding, *Proc. of Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.3213-3223, 2016.

[30] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard et al., Tensorflow: A system for large-scale machine learning, *Operating Systems Design and Implementation (OSDI)*, 2016.