Pyramid Stereo Matching Network

Youngju Yoo

Introduction

- Depth estimation from stereo matching : compute the **disparity** d for each pixel in the reference image
- Recent work :
  - CNN compute the similarity.
  - Rely on patch-based Siamese networks, a limit to exploit context information in ill-posed regions
- Therefore, regional support from global context information must be incorporated into stereo matching.
- **Pyramid stereo matching network (PSMNet)** : exploit **global context information**

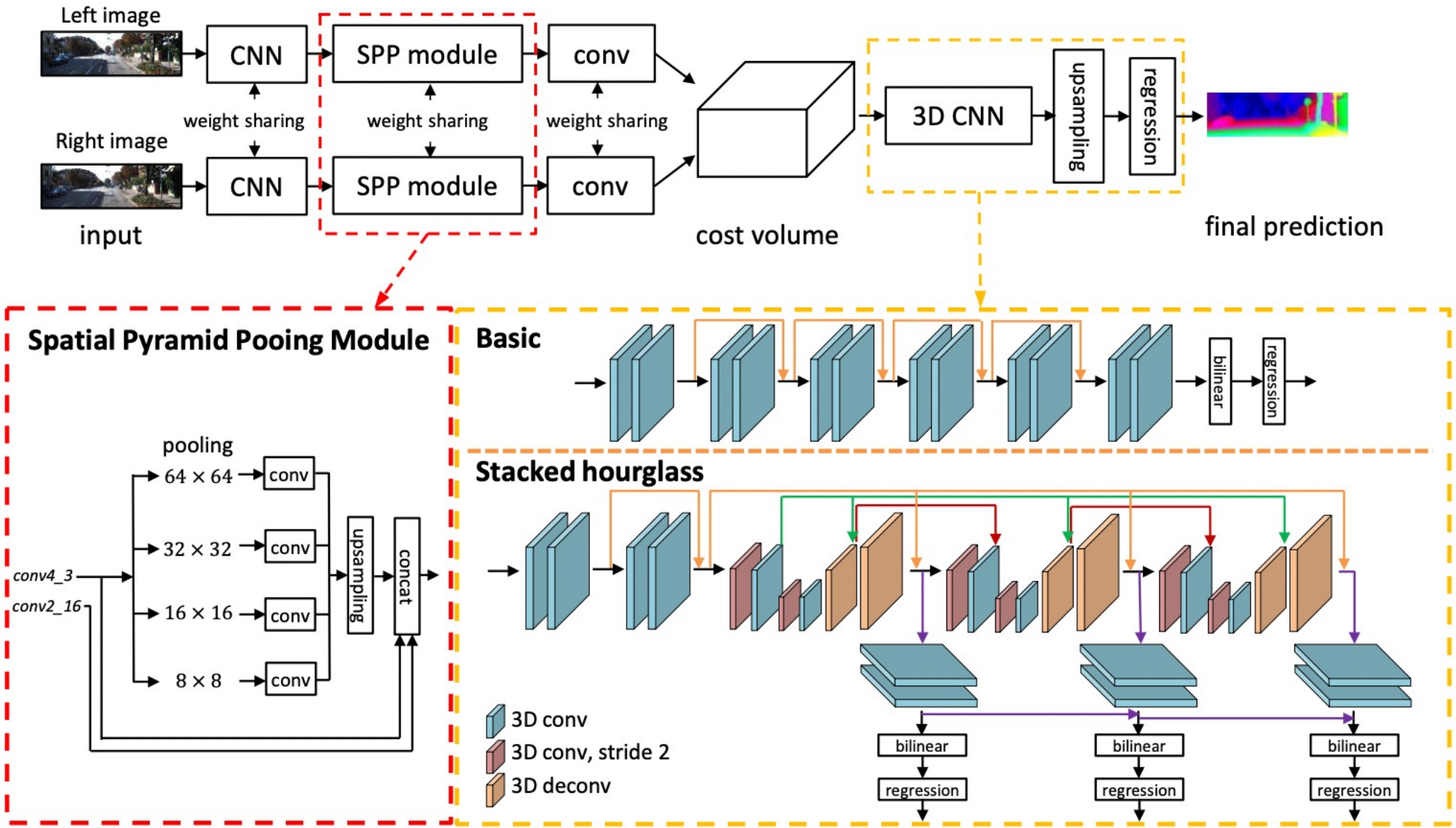
Introduction

- **Spatial pyramid pooling(SPP)**
  - Extends pixel-level features to region-level features with different scales of receptive fields
  - Form the cost volume using combined global and local feature
- **The stacked hourglass 3D CNN**
  - Regularize the cost volume.
  - Repeatedly processes the cost volume in a top-down/bottom-up manner to further improve the utilization of global context information.

Related work

- **MC-CNN** : a deep Siamese network to compute matching cost, the network is trained to lear n to predict the similarity between image patches.
- **DispNet** : an end-to-end networks for the estimation of disparity, it can predict whole disparit y maps without post-processing
- **CRL** : Extend DispNet and introduce a two-stage network, Cascade residual learning. The fir st and second stages calculate the disparity map. Final disparity map is sum of both
- **GC-Net** : an end-to-end network for cost volume regularization using 3D CNN with the encod er-decoder architecture to incorporate context information to reduce mismatch in ambiguous regions and thus improve depth estimation

PSMNet – Network Architecture

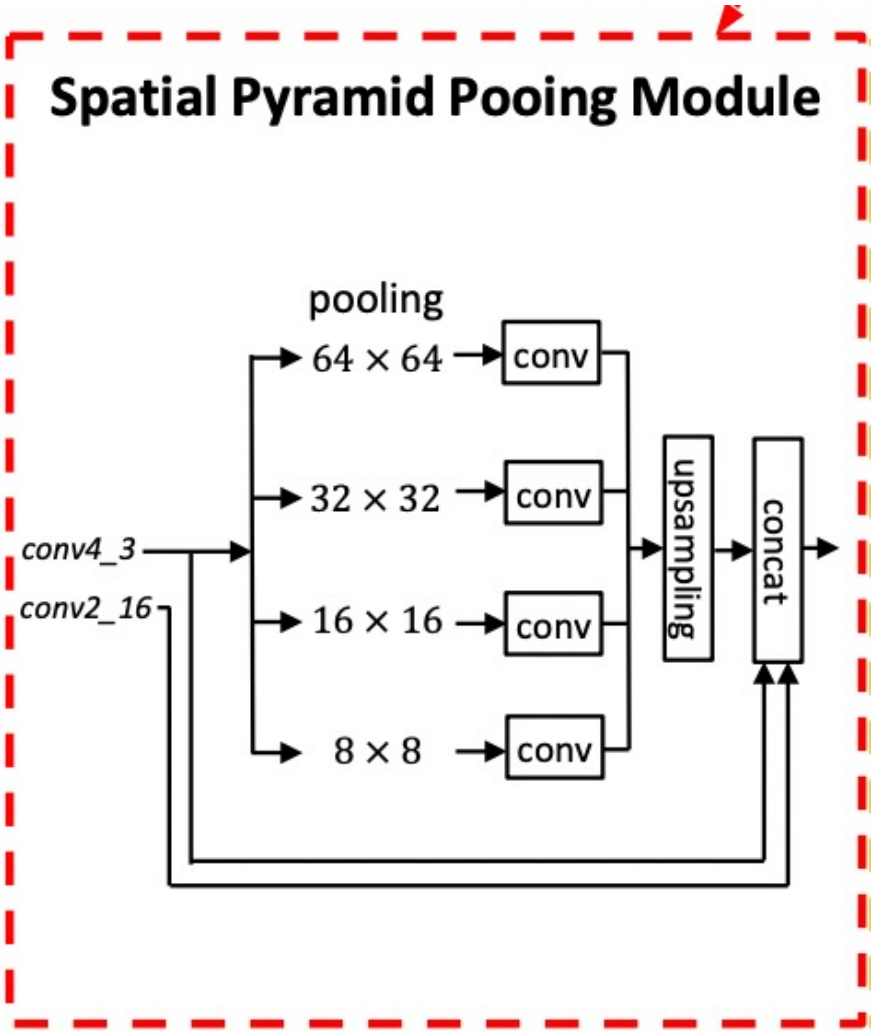


PSMNet – Network Architecture

- Three small convolution filters (3 × 3) are cascaded to construct a deeper network with the same receptive field.
- The basic residual blocks for learning the unary feature extraction and **dilated CNN is applied to enlarge the receptive field**.
- The output feature map size is ¼ of the input image size.

Name	Layer setting	Output dimension
input		$H \times W \times 3$
CNN		
conv0_1	$3 \times 3, 32$	$\frac{1}{2}H \times \frac{1}{2}W \times 32$
conv0_2	$3 \times 3, 32$	$\frac{1}{2}H \times \frac{1}{2}W \times 32$
conv0_3	$3 \times 3, 32$	$\frac{1}{2}H \times \frac{1}{2}W \times 32$
conv1_x	$\begin{bmatrix} 3 \times 3, 32 \\ 3 \times 3, 32 \end{bmatrix} \times 3$	$\frac{1}{2}H \times \frac{1}{2}W \times 32$
conv2_x	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 16$	$\frac{1}{4}H \times \frac{1}{4}W \times 64$
conv3_x	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 3$, dila = 2	$\frac{1}{4}H \times \frac{1}{4}W \times 128$
conv4_x	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 3$, dila= 4	$\frac{1}{4}H \times \frac{1}{4}W \times 128$

PSMNet – Spatial Pyramid Pooling Module

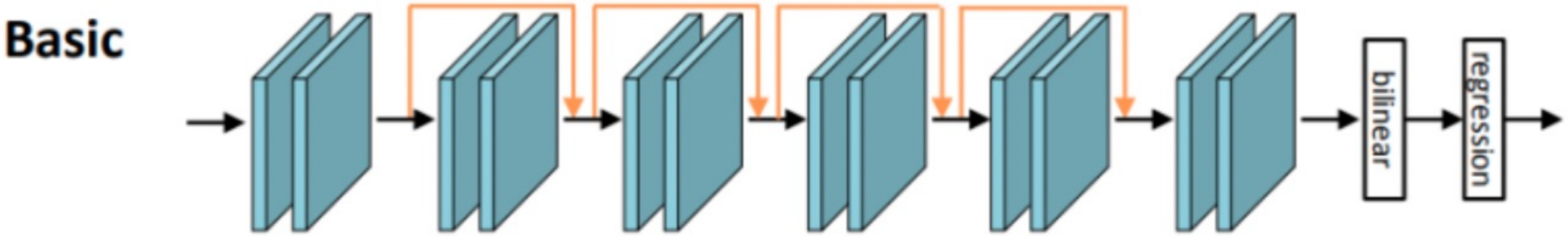


- Four fixed-size average pooling blocks for SPP : 64 × 64, 32 × 32, 16 × 16, and 8×8.
- The relationship between an object and sub-regio ns is learned by the SPP Module.
- **Object context information** can solve the limitati ons of obtaining context relationship from pixel int ensities and benefit estimation, particularly for ill-posed regions.
- Unsampling with bilinear interpolation and Fusion.

PSMNet – Cost Volume

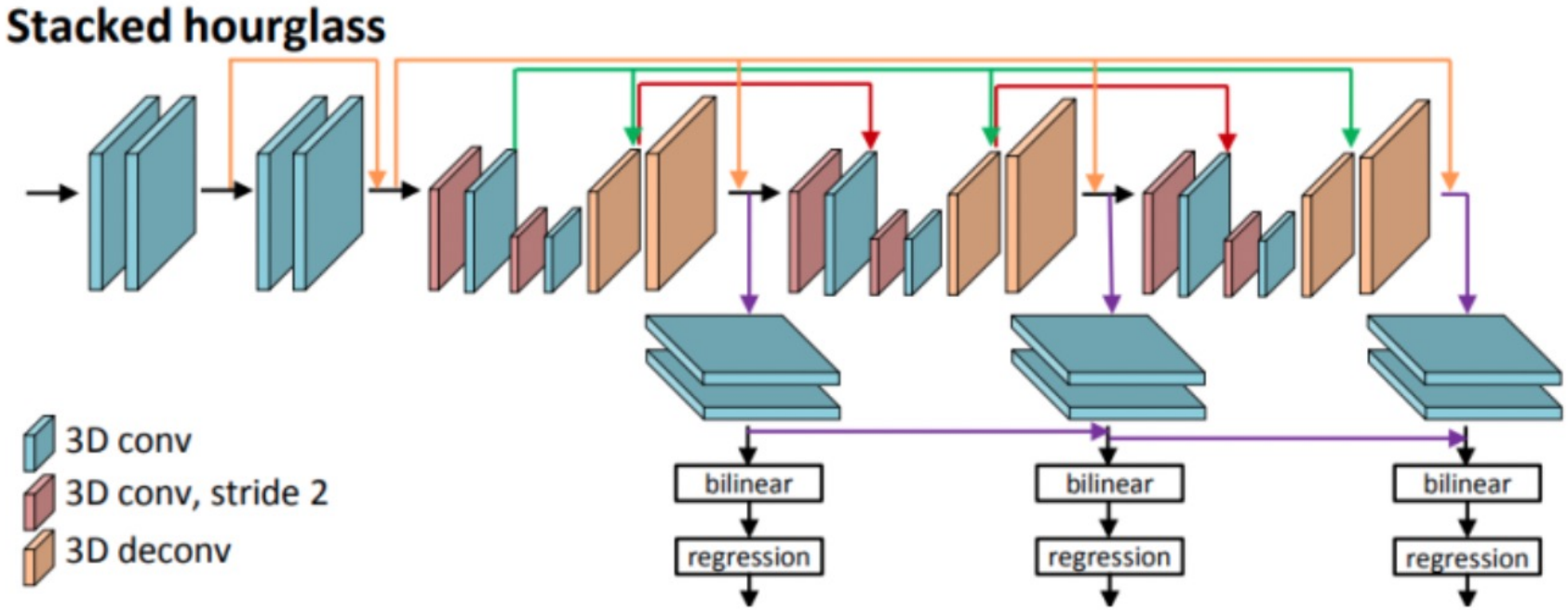
- PSMNet adopt SPP features to form a cost volume **by concatenating** left feature maps with their corresponding right feature maps across each disparity level
- **Cost volume** : 4D(height×width×disparity×feature size).

PSMNet – 3D CNN : Basic architecture



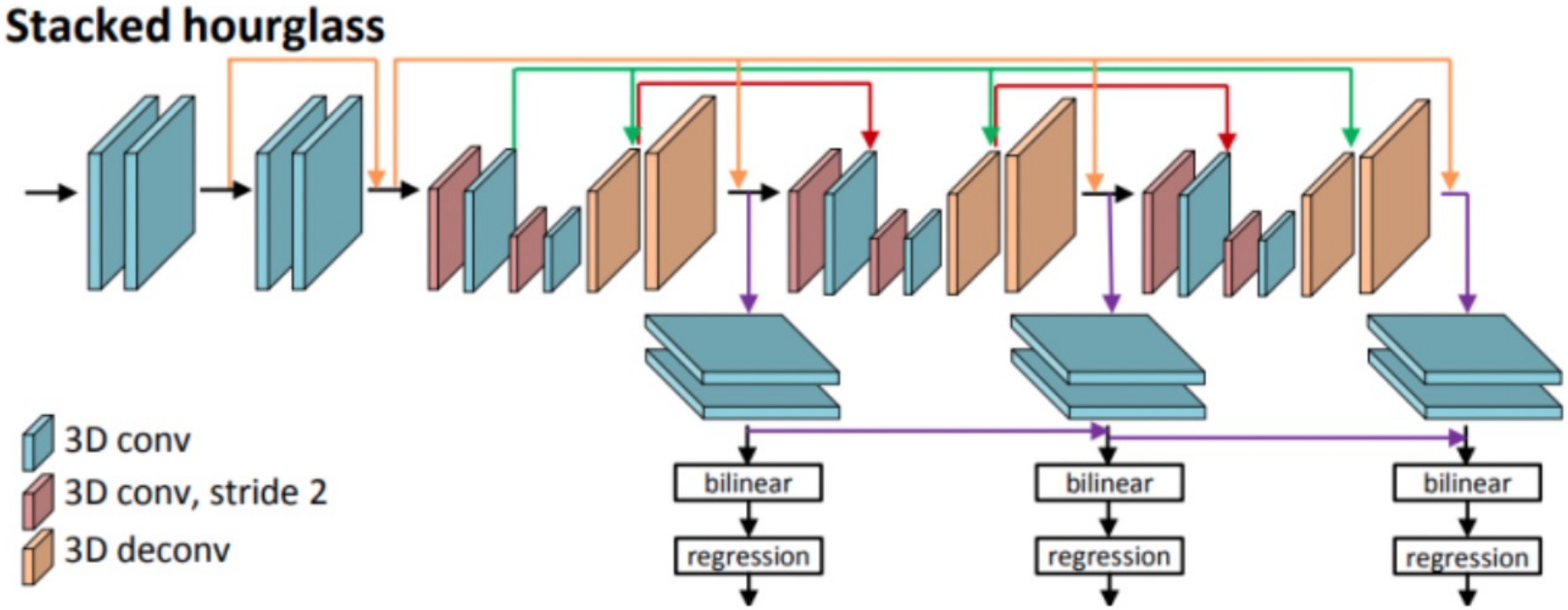
- In the basic architecture, the network is simply built using residual blocks with 3 × 3 × 3 convolutional layers.
- By bilinear interpolation, upsample the cost volume back to size H×W ×D.
- Finally, apply regression to calculate the disparity map with size H × W.
- Basic architecture was used to evaluate the performance of the SPP module.

PSMNet – 3D CNN : stacked hourglass



- to learn more context information
- consisting of repeated top-down/bottom-up processing in conjunction with intermedia te supervision

PSMNet – 3D CNN : stacked hourglass



- Each three main hourglass networks generates a disparity map and has **three outputs and losses** (Loss 1, Loss 2, and Loss 3).
- Train : the total loss is summation of the three losses.
- Test : the final disparity map is the last of three outputs

PSMNet – Disparity Regression

- Use disparity regression to estimate the continuous disparity map
- More robust than classification-based stereo matching methods(GC-Net)

$$\hat{d} = \sum_{d=0}^{D_{max}} d \times \sigma(-c_d).$$

- $\sigma(-c_d)$: The probability of each disparity d
- $\hat{d}$: predicted disparity, calculated as the sum of each disparity weighted by its probability

PSMNet – Loss

- The smooth L1 loss function is adopted to train the proposed PSMNet
- less sensitive to outliers, than other regression loss such as L2 loss

$$L(d, \hat{d}) = \frac{1}{N} \sum_{i=1}^{N} smooth_{L_1}(d_i - \hat{d}_i), \qquad (2)$$

in which

$$smooth_{L_1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases},$$

where N is the number of labeled pixels, d is the ground-truth disparity, and $\hat{d}$ is the predicted disparity.

Experiment - Datasets

- Scene Flow : A large scale synthetic dataset, the trained model was directly used for testing.
- KITTI 2015 : A real-world dataset with street views from a driving car, model trained with Scene Flow data after fine-tuning on the KITTI
- KITTI 2012 : A real-world dataset with street views from a driving car, model trained with Scene Flow data after fine-tuning on the KITTI

Experiment – KITTI 2015 : PSMNet

- Experiments including the usage of dilated convolution, pooling at different levels, a nd 3D CNN architectures to evaluate PSMNet.
- Dilated convolution with the SPP module works better.
- pooling with more levels works better
- The stacked hourglass 3D CNN with dilated convolution and the SPP module work s better.

Table 2. Evaluation of PSMNet with different settings. We computed the percentage of three-pixel-error on the KITTI 2015 validation set, and end-point-error on the Scene Flow test set. * denote that we use half the dilated rate of dilated convolution.

Network setting						KITTI 2015	Scene Flow
dilated conv	pyramid pooling size				stacked hourglass	Val Err (%)	End Point Err
	64 × 64	32 × 32	16 × 16	8 × 8			
						2.43	1.43
√						2.16	1.56
	√	√	√	√		2.47	1.40
√	√					2.17	1.30
√	√	√	√	√		2.09	1.28
√	√	√	√	√	√	1.98	**1.09**
√*	√	√	√	√	√	**1.83**	1.12

Experiment – KITTI 2015 : Loss Weight

- Experiments with various combinations of loss weights between 0 and 1.
- The weight settings of 0.5 for Loss 1, 0.7 for Loss 2, and 1.0 for Loss 3 yielded the best performance, which was a 1.98% error rate on the KITTI 2015 validation set.

Table 3. Influence of weight values for Loss_1, Loss_2, and Loss_3 on validation errors. We empirically found that 0.5/0.7/1.0 yielded the best performance.

Loss weight			KITTI 2015
Loss_1	Loss_2	Loss_3	val error(%)
0.0	0.0	1.0	2.49
0.1	0.3	1.0	2.07
0.3	0.5	1.0	2.05
0.5	0.7	1.0	**1.98**
0.7	0.9	1.0	2.05
1.0	1.0	1.0	2.01

Experiment – KITTI 2015 : Result

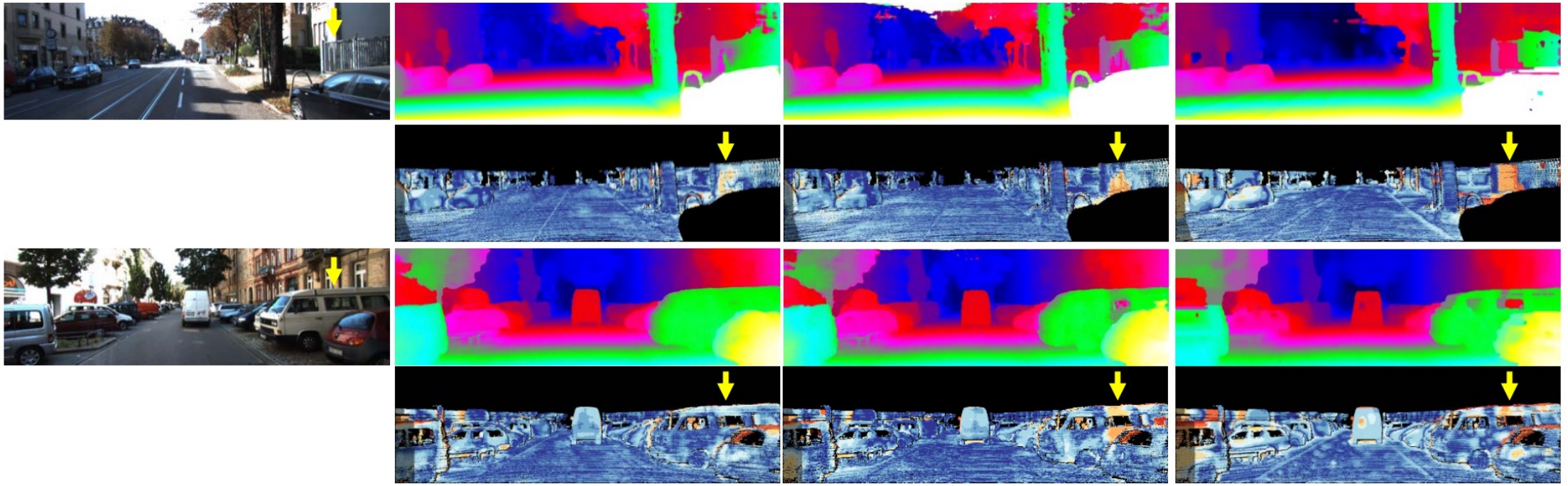
- Calculate the disparity maps for the 200 testing images in the KITTI 2015 dataset usi ng the best model trained in experiments.
- The overall three-pixel-error for the proposed PSMNet was 2.32% in D1-all, which su rpassed prior studies

Table 4. The KITTI 2015 leaderboard presented on March 18, 2018. The results show the percentage of pixels with errors of more than three pixels or 5% of disparity error from all test images. Only published methods are listed for comparison.

Rank	Method	All (%)			Noc (%)			Runtime (s)
		D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	
1	PSMNet (ours)	**1.86**	4.62	**2.32**	**1.71**	4.31	**2.14**	0.41
3	iResNet-i2e2 [14]	2.14	3.45	2.36	1.94	3.20	2.15	0.22
6	iResNet [14]	2.35	**3.23**	2.50	2.15	**2.55**	2.22	**0.12**
8	CRL [21]	2.48	3.59	2.67	2.32	3.12	2.45	0.47
11	GC-Net [13]	2.21	6.16	2.87	2.02	5.58	2.61	0.90

Experiment – KITTI 2015 : Evaluation

- PSMNet yields more robust results, particularly in ill-posed regions



(a) PSMNet (b) GC-Net (c) MC-CNN

Figure 2. Results of disparity estimation for KITTI 2015 test images. The left panel shows the left input image of stereo image pair. For each input image, the disparity maps obtained by (a) PSMNet, (b) GC-Net [13], and (c) MC-CNN [30] are illustrated together above their error maps.

Experiment – Scene Flow

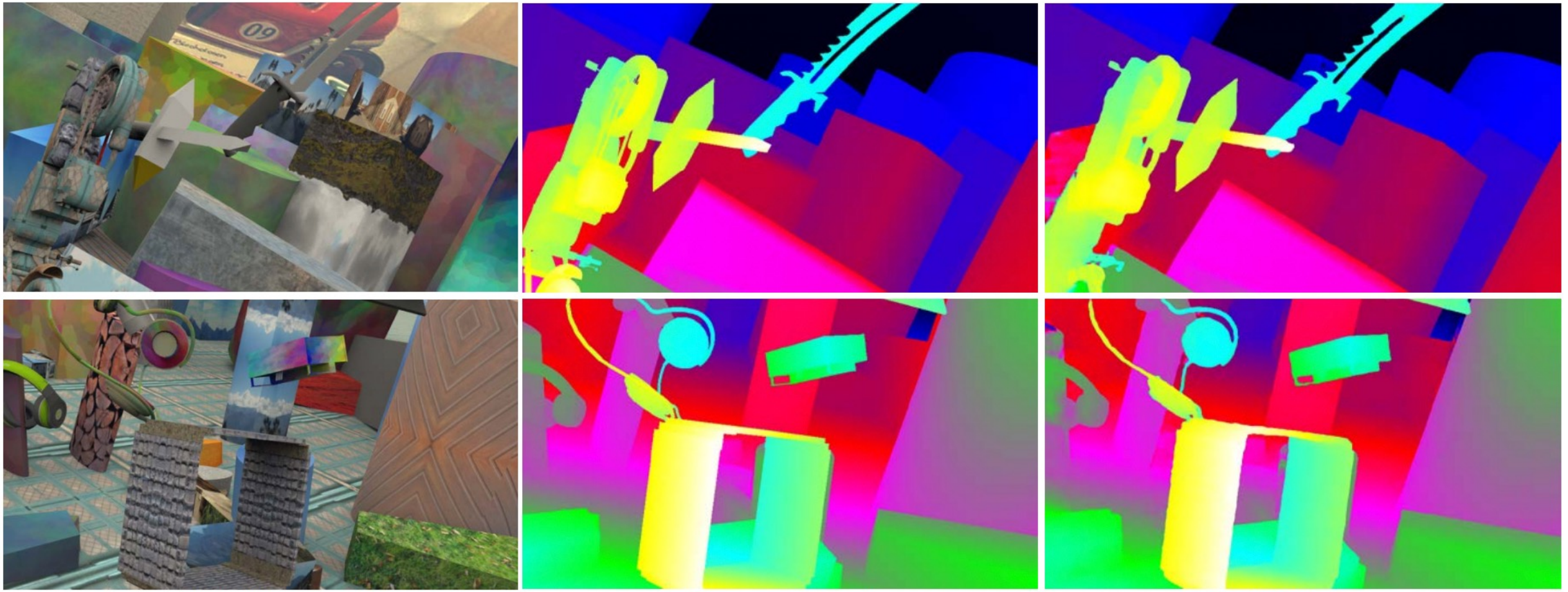
- compared the performance of PSMNet with other state-of-the-art methods using the Scene Flow test set
- PSMNet outperformed other methods in terms of accuracy

Table 5. Performance comparison with Scene Flow test set. EPE: End-point-error.

	PSMNet	CRL [21]	DispNetC [19]	GC-Net [13]
EPE	**1.09**	1.32	1.68	2.51

Experiment – Scene Flow

- PSMNet obtains accurate disparity maps for delicate and intricately overlapped objects.



(a) left image (b) ground truth (c) PSMNet output

Experiment – KITTI 2012

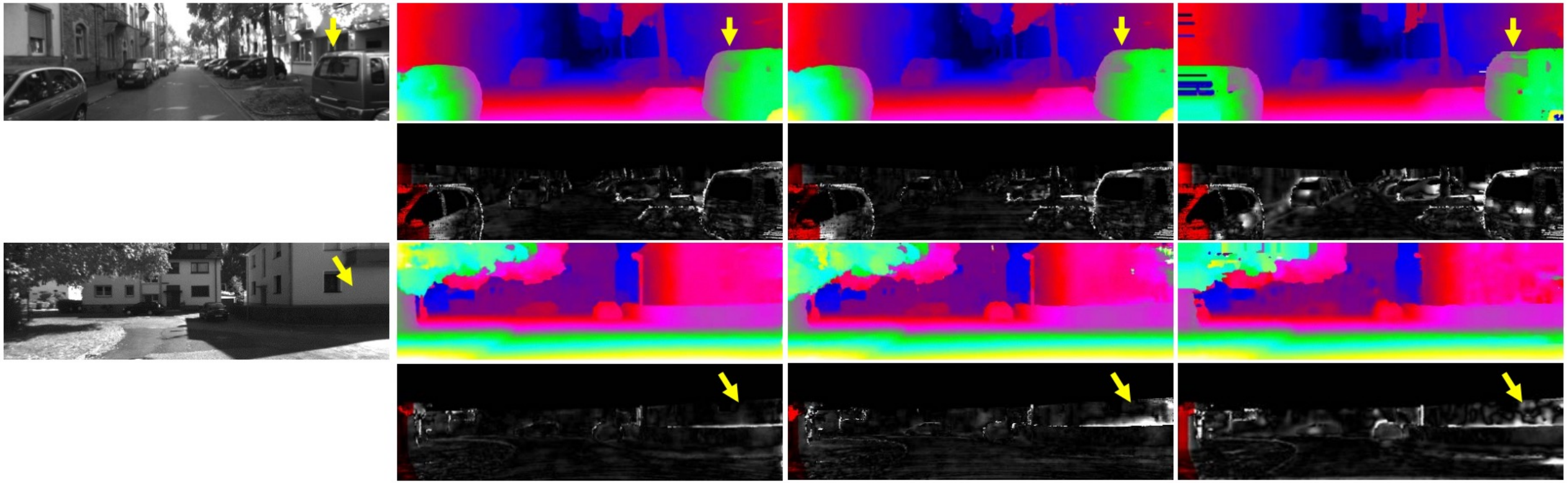
- According to the online leaderboard, the overall three-pixel-error for the proposed PS MNet was 1.89%, which surpassed prior studies.

Table 6. The leaderboard of KITTI 2012 presented on March 18, 2018. PSMNet achieves the best results under all evaluation criteria, except runtime. Only published methods are listed for comparison.

Rank	Method	>2 px		>3 px		>5 px		Mean Error		Runtime (s)
		Noc	All	Noc	All	Noc	All	Noc	All	
1	PSMNet (ours)	**2.44**	**3.01**	**1.49**	**1.89**	**0.90**	**1.15**	**0.5**	**0.6**	0.41
2	iResNet-i2 [14]	2.69	3.34	1.71	2.16	1.06	1.32	0.5	0.6	**0.12**
4	GC-Net [13]	2.71	3.46	1.77	2.30	1.12	1.46	0.6	0.7	0.9
11	L-ResMatch [27]	3.64	5.06	2.27	3.40	1.50	2.26	0.7	1.0	48
14	SGM-Net [26]	3.60	5.15	2.29	3.50	1.60	2.36	0.7	0.9	67

Experiment – KITTI 2012

- PSMNet obtains more robust results, particularly in regions of car windows and walls



(a) PSMNet (b) GC-Net (c) MC-CNN

Figure 4. Results of disparity estimation for KITTI 2012 test images. The left panel shows the left input image of the stereo image pair. For each input image, the disparity obtained by (a) PSMNet, (b) GC-Net [13], and (c) MC-CNN [30], is illustrated above its error map.

Conclusion

- PSMNet, a CNN architecture for stereo vision which consists of two main modules, t he SPP module and the 3D CNN, to exploit context information.
- PSMNet outperforms other state-of-the-art methods.
- The estimated disparity maps clearly demonstrate that PSMNet significantly reduces errors in ill-posed regions.

Group-wise Correlation Stereo Network

Youngju Yoo

Introduction & Related work

- **The full correlation cost volume :**

$$\mathbf{C}_{corr}(d,x,y) = \frac{1}{N_c}\langle \mathbf{f}_l(x,y), \mathbf{f}_r(x-d,y)\rangle$$

  - An efficient way for measuring feature similarities
  - It loses much information because it produces only a single-channel correlation map for each disp arity level.

- **The concatenation volume :**

$$\mathbf{C}_{concat}(d,x,y,\cdot) = \text{Concat}\{\mathbf{f}_l(x,y), \mathbf{f}_r(x-d,y)\}$$

  - no information about the feature similarities
  - more parameters are required in the following aggregation network to learn the similarity measure ment function

- **Group-wise correlation stereo network(GwcNet)** : combining advantages of the c oncatenation volume and the correlation volume.

GwcNet – Network architecture

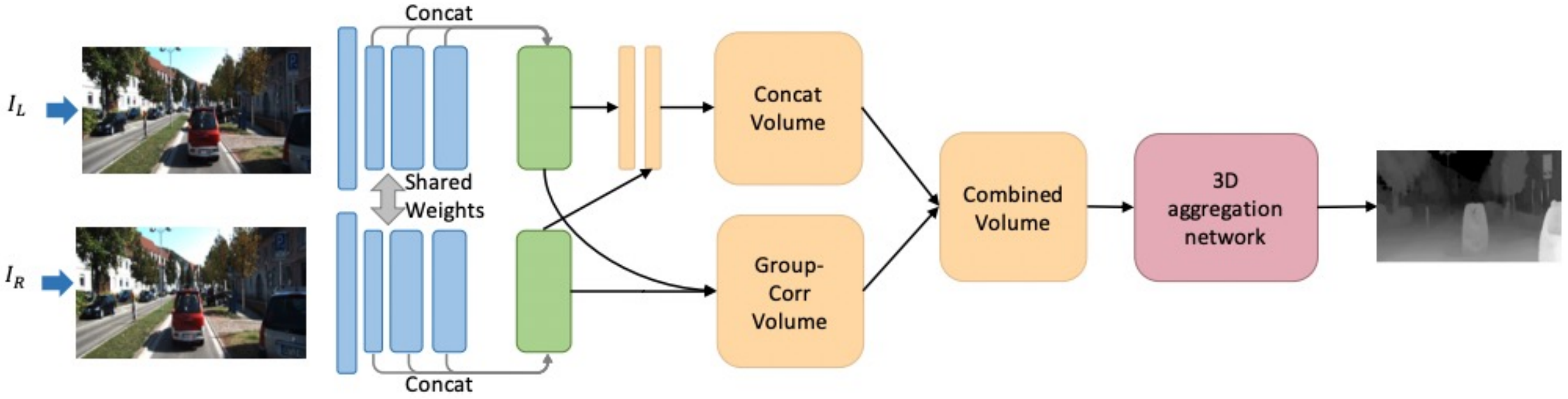


Figure 1: The pipeline of the proposed group-wise correlation network. The whole network consists of four parts, unary feature extraction, cost volume construction, 3D convolution aggregation, and disparity prediction. The cost volume is divided into two parts, concatenation volume (*Cat*) and group-wise correlation volume (*Gwc*). Concatenation volume is built by concatenating the compressed left and right features. Group-wise correlation volume is described in Section 3.2.

- **Group-wise correlation stereo network(GwcNet)** : group-wise correlation cost volume and improved 3D stacked hourglass networks.

GwcNet – Group-wise correlation volume

- Split the features into groups and computing correlation maps **group by group**
- All the channels are divided into N_g groups along the cannel dimension, thus each fe ature group has N_c/N_g channels.(N_c : the channels of unary features)
- The gth feature group f_l^g, f_r^g : $g\frac{N_c}{N_g}$, $g\frac{N_c}{N_g}$ + 1, … , $g\frac{N_c}{N_g}$ + ($\frac{N_c}{N_g}$ − 1) th channels
- The group-wire correlation is computed as

$$\mathbf{C}_{gwc}(d,x,y,g) = \frac{1}{N_c/N_g}\langle \mathbf{f}_l^g(x,y), \mathbf{f}_r^g(x-d,y)\rangle$$

- Each C_{gwc} is computed from the corresponding feature group.
- Cost volume shape : [D_{max}/4, H/4, W/4, N_g]

GwcNet – Group-wise correlation volume

- Proposed group-wise correlation volume requires **less 3D aggregation parameters** to achieve favorable results.
- To further improve the performance, the group correlation cost volume can be **combined with the concatenation volume.**

Cost Volume		
unary_l/r	N/A, S2	$H/4 \times W/4 \times 320$
volume_g	group-wise cost volume	$D/4 \times H/4 \times W/4 \times 40$
volume_c	concatenation cost volume	$D/4 \times H/4 \times W/4 \times 24$
volume	volume_g,volume_c: Concat	$D/4 \times H/4 \times W/4 \times 64$

GwcNet – Improved 3D CNN

- Modify the stacked hourglass architecture in PSMNet to make it suitable for group-wise correlation and improve the inference speed.

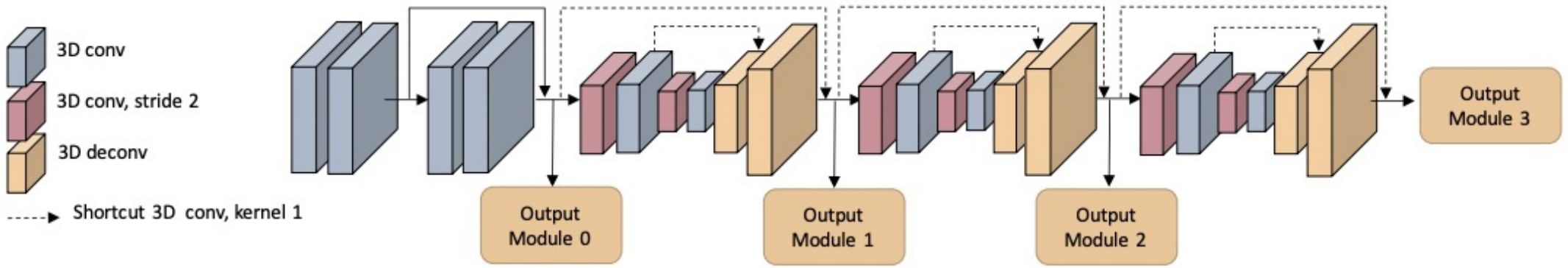
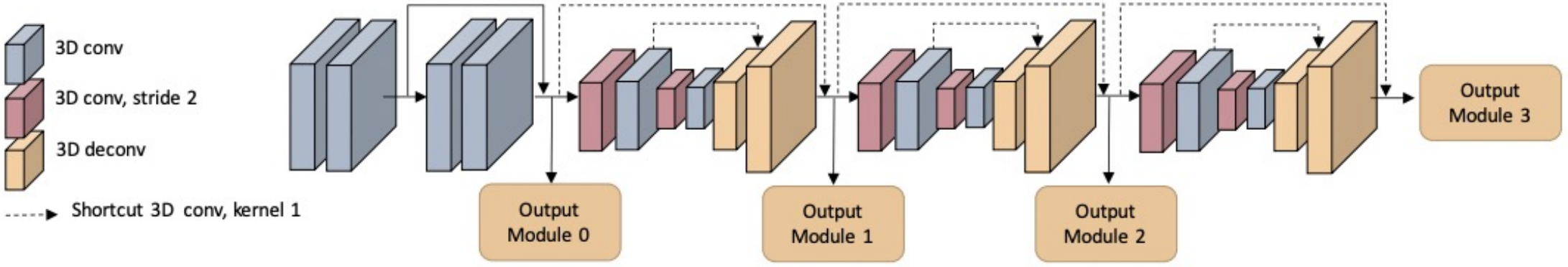


Figure 2: The structure of our proposed 3D aggregation network. The network consists of a pre-hourglass module (four convolutions at the beginning) and three stacked 3D hourglass networks. Compared with PSMNet [2], we remove the shortcut connections between different hourglass modules and output modules, thus output modules 0,1,2 can be removed during inference to save time. $1\times1\times1$ 3D convolutions are added to the shortcut connections within hourglass modules.

GwcNet – Improved 3D CNN

- Add one more auxiliary output module (output module 0) : it's loss makes the networ k learn better features at lower layers, which also benefits the final prediction.
- The residual connections between different output modules are removed : output mo dule 0,1,2 can be removed during inference to save computational cost.
- Add 1×1×1 3D convolutions to the shortcut connections : to improve the performance without increasing much computational cost.



GwcNet – Output Module

- For each output module, two 3D CNN are employed to output a 1-channel 4D volume, and then the volume is upsampled and converted into a probability volume with softmax function along the disparity dimension.

Output Module 0, 1, 2, 3		
input	N/A	$D/4 \times H/4 \times W/4 \times 32$
conv1	32×32, $3 \times 3 \times 3$, S1	$D/4 \times H/4 \times W/4 \times 32$
conv2**	32×1, $3 \times 3 \times 3$, S1	$D/4 \times H/4 \times W/4 \times 1$
score	Upsample	$D \times H \times W \times 1$
prob	Softmax (at disparity dimension)	$D \times H \times W \times 1$
disparity	Soft Argmin (Equ. 4)	$H \times W \times 1$

GwcNet – Output Module

- Each pixel has a D_{max}-length vector which contains the probability p for all disparity l evels.
- The disparity estimation is computed by the soft argmin function

$$\tilde{d} = \sum_{k=0}^{D_{max}-1} k \cdot p_k$$

- κ : possible disparity level
- p_κ : corresponding probability
- It is both fully differentiable and able to regress a smooth disparity estimate.

GwcNet – Loss

- The smooth L1 loss function same as PSMNet is adopted.

$$L = \sum_{i=0}^{i=3} \lambda_i \cdot \text{Smooth}_{L_1}(\tilde{\mathbf{d}}_i - \mathbf{d}^*)$$

$$\text{Smooth}_{L_1}(x) = \begin{cases} 0.5x^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise} \end{cases}$$

- $\tilde{d}_i$: the predicted disparity maps from the four output modules
- d^* : groud-truth
- λ_i : the coefficients for the ith disparity prediction

Experiment - Datasets

- Scene Flow : A large scale synthetic dataset, the trained model was directly used for testing.
- KITTI 2015 : A real-world dataset with street views from a driving car, model trained with Scene Flow data after fine-tuning on the KITTI
- KITTI 2012 : A real-world dataset with street views from a driving car, model trained with Scene Flow data after fine-tuning on the KITTI
- The coefficients of four outputs : λ_0 = 0.5, λ_1 = 0.5, λ_2 = 0.7, λ_3 = 1.0,

Experiment – The effectiveness of Gwc

- Base model : remove the stacked hourglass network, only use the output module 0
- the performance of the Gwc-Base network increases **as the group number increases**.
- All the Gwc-Base models except Gwc1-Base outperform the Cat-Base model
- Gwc volume provide accurate matching feature, and the cat volume provide complementary semantic information

Model	Concat Volume	Group Corr Volume	Stack Hour-glass	Groups × Channels	Init Volume Channel	>1px (%)	>2px (%)	>3px (%)	EPE (px)	Time (ms)
Cat64-Base	✓			-	64	12.78	8.05	6.33	1.308	117.1
Gwc1-Base		✓		1×320	1	13.32	8.37	6.62	1.369	104.0
Gwc10-Base		✓		10×32	10	11.82	7.31	5.70	1.230	112.8
Gwc20-Base		✓		20×16	20	11.84	7.29	5.67	1.216	116.3
Gwc40-Base		✓		40×8	40	11.68	7.18	5.58	1.212	122.2
Gwc80-Base		✓		80×4	80	11.69	7.17	5.57	1.214	133.3
Gwc160-Base		✓		160×2	160	11.58	7.08	5.49	1.188	157.3
Gwc40-Cat24-Base	✓	✓		40×8	40+24	**11.26**	**6.87**	**5.31**	**1.127**	135.1
PSMNet [2]	✓		[2]	-	64	9.46	5.19	3.80	0.887	246.1
Cat64-original-hg	✓		[2]	-	64	9.47	5.13	3.74	0.876	241.0
Cat64	✓		Ours	-	64	8.41	4.63	3.41	0.808	198.3
Gwc40 (GwcNet-g)		✓	Ours	40×8	40	8.18	4.57	3.39	0.792	200.3
Gwc40-Cat24 (GwcNet-gc)	✓	✓	Ours	40×8	40+24	**8.03**	**4.47**	**3.30**	**0.765**	210.7

Experiment – Stacked hourglass

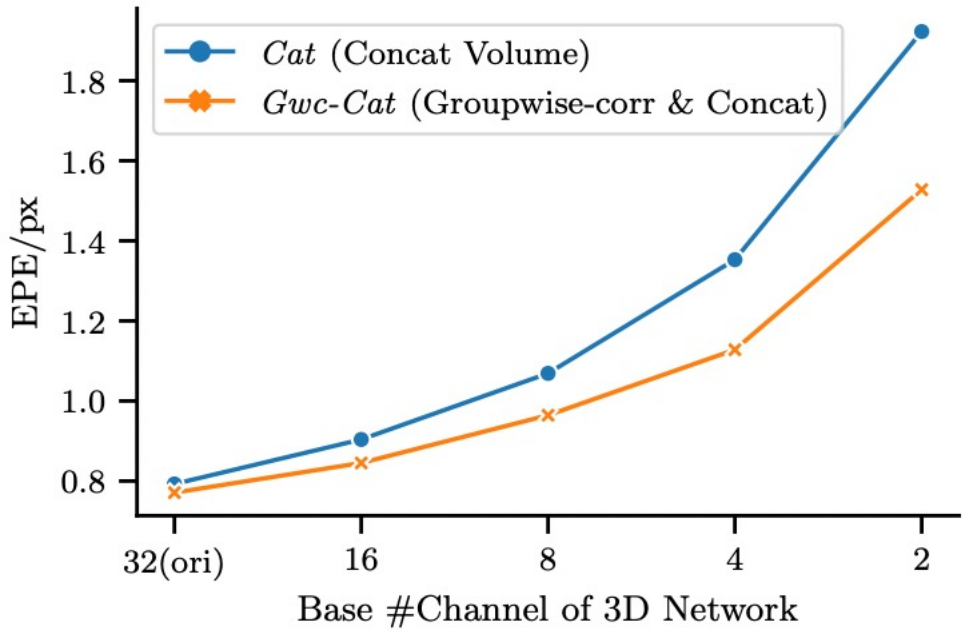
- Model with proposed hourglass network increases EPE by 7.8% in Scene Flow dataset and 5.8% in KITTI 2015
- The inference time also decreases by 42.7ms since the auxiliary output modules can be removed during inference to save time.(The residual connections between differe nt output modules are removed)

Model	KITTI 12 EPE (px)	KITTI 12 D1-all(%)	KITTI 15 EPE (px)	KITTI 15 D1-all (%)
PSMNet [2]	0.713	2.53	0.639	1.50
Cat64-original-hg	0.740	2.72	0.652	1.76
Cat64	0.691	2.41	0.615	1.55
Gwc40	0.662	2.30	**0.602**	**1.41**
Gwc40-Cat24	**0.659**	**2.10**	0.613	1.49

Table 3: Ablation study results of our networks on KITTI 2012 validation and KITTI 2015 validation sets.

Experiment – Computational cost of 3D CNN

- As the number of channels decreasing, Gwc-cat perform better than Cat, and the per formance gain enlarges.
- Gwc provides good matching cost representations for the 3D aggregation network
- Concatenation volume needs to learn the matching similarity function → require mor e parameters and computational cost.



Experiment – KITTI 2012 & 2015

- With validation set, the performance of both Gwc40 and Gwc40-Cat24 is better than the models without Gwc in both KITTI 2012 and KITTI 2015.

Model	KITTI 12 EPE (px)	KITTI 12 D1-all(%)	KITTI 15 EPE (px)	KITTI 15 D1-all (%)
PSMNet [2]	0.713	2.53	0.639	1.50
Cat64-original-hg	0.740	2.72	0.652	1.76
Cat64	0.691	2.41	0.615	1.55
Gwc40	0.662	2.30	**0.602**	**1.41**
Gwc40-Cat24	**0.659**	**2.10**	0.613	1.49

Table 3: Ablation study results of our networks on KITTI 2012 validation and KITTI 2015 validation sets.

Experiment – KITTI 2012 & 2015

	All (%)			Noc (%)			Time
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all	(s)
DispNetC [14]	4.32	4.41	4.34	4.11	3.72	4.05	0.06
GC-Net [6]	2.21	6.16	2.87	2.02	5.58	2.61	0.9
CRL [17]	2.48	3.59	2.67	2.32	3.12	2.45	0.47
iResNet-i2e2 [12]	2.14	3.45	2.36	1.94	3.20	2.15	0.22
PSMNet [6]	1.86	4.62	2.32	1.71	4.31	2.14	0.41
SegStereo [26]	1.88	4.07	2.25	1.76	3.70	2.08	0.6
GwcNet-g (Gwc40)	**1.74**	**3.93**	**2.11**	**1.61**	**3.49**	**1.92**	0.32

Table 4: KITTI 2015 test set results. The dataset contains 200 images for training and 200 images for testing.

	>2px (%)		>3px (%)		>5px (%)		Mean Error (px)		Time
	Noc	All	Noc	All	Noc	All	Noc	All	(s)
DispNetC [14]	7.38	8.11	4.11	4.65	2.05	2.39	0.9	1.0	0.06
MC-CNN-acrt [29]	3.90	5.45	2.43	3.63	1.64	2.39	0.7	0.9	67
GC-Net [6]	2.71	3.46	1.77	2.30	1.12	1.46	0.6	0.7	0.9
iResNet-i2 [12]	2.69	3.34	1.71	2.16	1.06	1.32	0.5	0.6	0.12
SegStereo [26]	2.66	3.19	1.68	2.03	1.00	1.21	0.5	0.6	0.6
PSMNet [6]	2.44	3.01	1.49	1.89	0.90	1.15	0.5	0.6	0.41
GwcNet-gc (Gwc40-Cat24)	**2.16**	**2.71**	**1.32**	**1.70**	**0.80**	**1.03**	**0.5**	**0.5**	0.32

Table 5: KITTI 2012 test set results. The dataset contains 194 images for training and 195 images for testing.

Conclusion

- The paper proposed GwcNet to estimate disparity maps for stereo matching, which i ncorporates groupwise correlation to build up the cost volumes.
- The groupwise correlation volumes provide good matching features for the 3D aggre gation network.
- Gwc volume with improved stacked hourglass improve the performance and reduce t he inference time, and also computational cost.
- GwcNet overcomes both drawbacks of PSMNet and full correlation well.