北京理工大學
BEIJING INSTITUTE OF TECHNOLOGY

Beijing Engineering Research Center
of Mixed Reality and Advanced Display

IGTA2019

Multi-Level Context Ultra-Aggregation for Stereo Matching

Guang-Yu Nie[1], Ming-Ming Cheng[2], Yun Liu[2], Zhengfa Liang[3], Deng-Ping Fan[2], Yue Liu[1,4], and Yongtian Wang[1,4]

[1] Beijing Institute of Technology [2] TKLNDST, CS, Nankai University
[3] National Key Laboratory of Science and Technology on Blind Signal Processing
[4] AICFVE, Beijing Film Academy

Depth from Stereo
What is stereo?

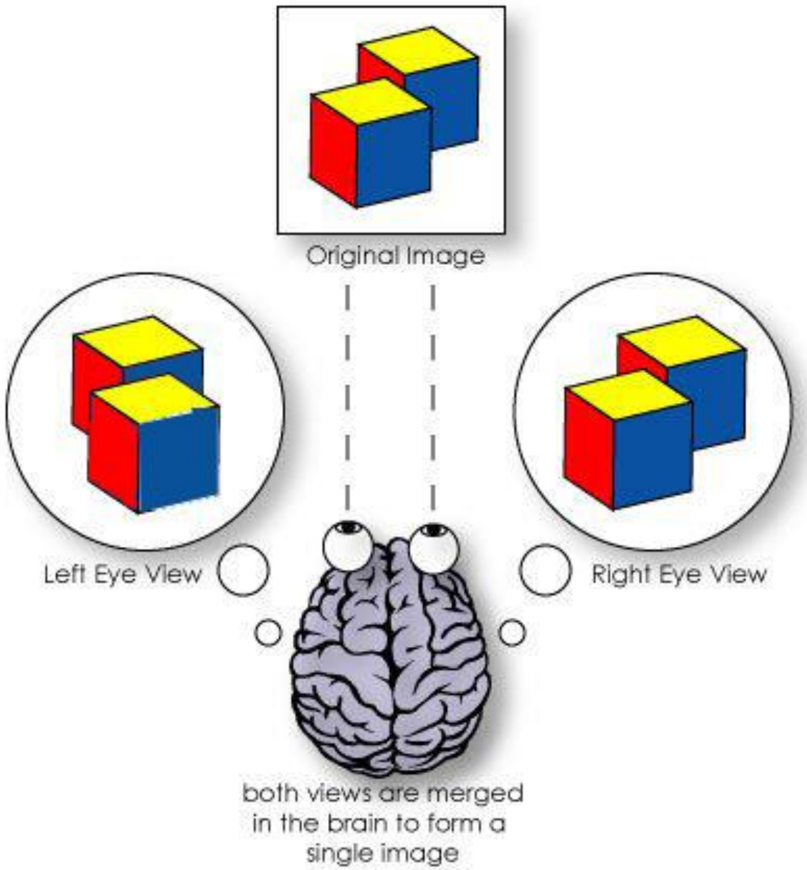
Depth from images is a very intuitive ability

- Given two images of a scene from (slightly) different viewpoints, we are able to infer depth

Can we do the same using computers?

- Yes

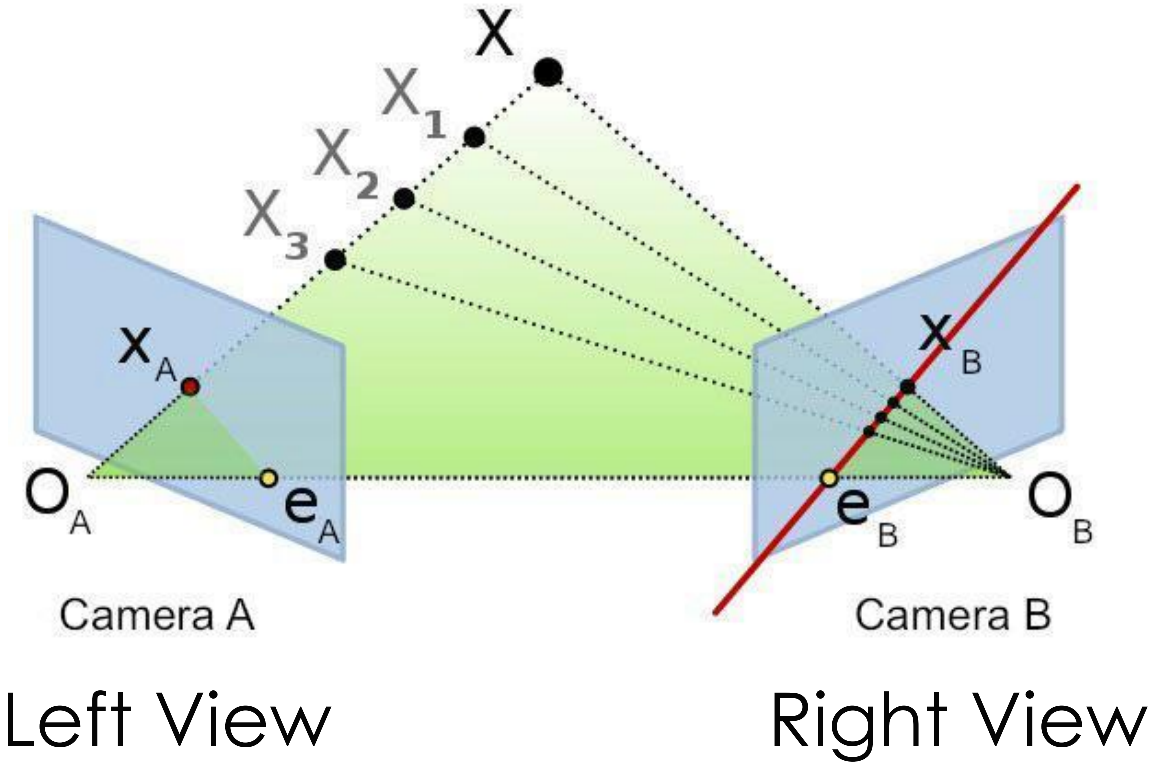
3D Stereo Scene



Source: http://www.vudream.com/reasons-why-virtual-reality-is-happening-now/3d-brain/

Depth from Stereo
Geometry in stereo

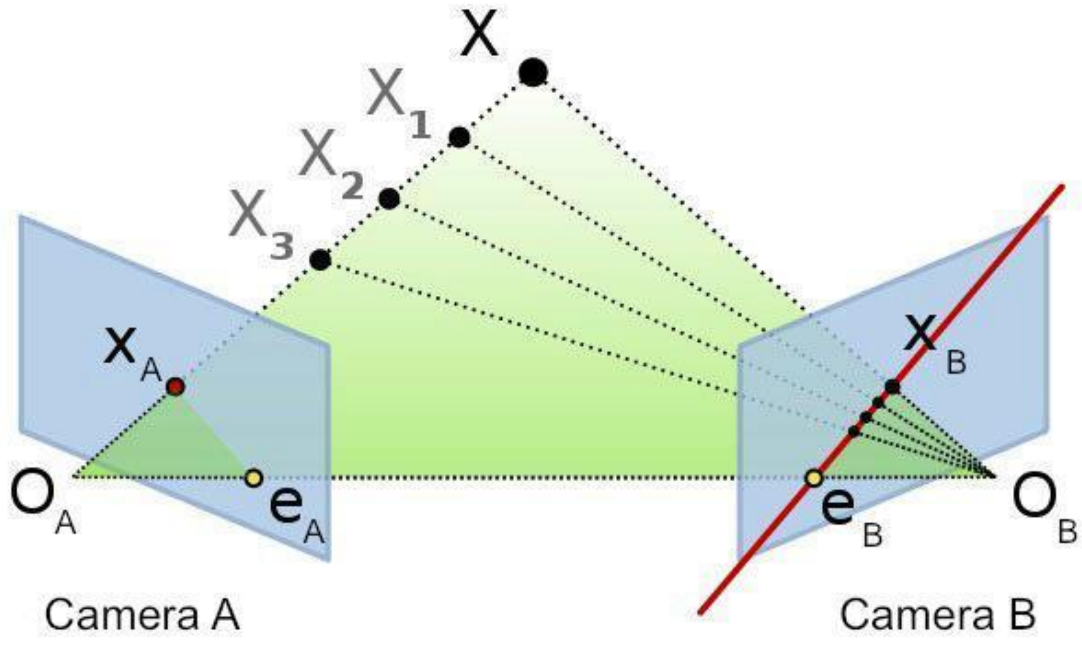
- Think of images as projections of 3D points (in the real world) onto a 2D surface (image plane)
- X_A is the projection of X, X_1, X_2, X_3, ... onto the left image
- X, X_1, X_2, X_3 will also project onto the right image



Source: Schairer, Edward, et al. "Measurements of tip vortices from a full-Scale UH-60A rotor by retro-reflective background oriented schlieren and stereo photogrammetry." (2013).

Depth from Stereo
Geometry in stereo

- Projections of X_1, X_2, X_3 on right image all lie on a line
- This line is known as an **epipolar line**
  - Projections of cameras' optical centers O_A, O_B onto the images
  - Points e_A, e_B are known as **epipoles**
  - All epipolar lines will intersect at epipoles
  - Left image has corresponding epipolar line



Left View

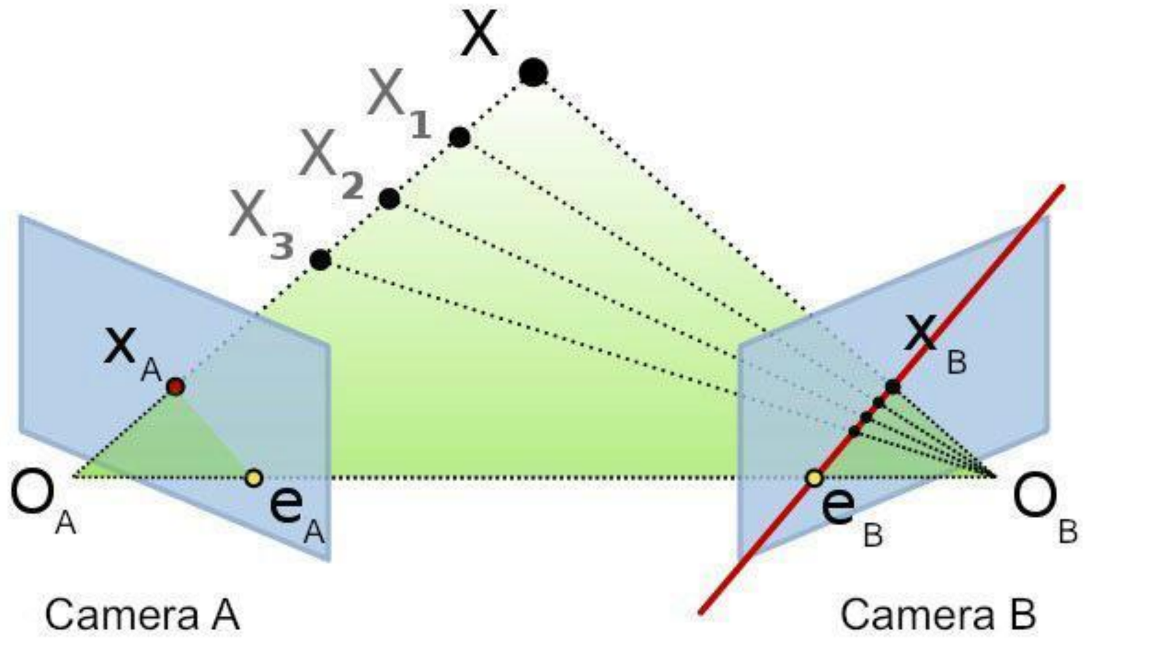
Right View

Source: Schairer, Edward, et al. "Measurements of tip vortices from a full-Scale UH-60A rotor by retro-reflective background oriented schlieren and stereo photogrammetry." (2013).

Depth from Stereo
Geometry in stereo

What does this give us?

- All 3D points that could have resulted in X_A must have a projection on the right image, and must be on the epipolar line $e_B - x_B$
- Given just the left/right images and X_A, you can search on the corresponding epipolar line in the right image. **If you can find the corresponding match X_B, you can uniquely determine the 3D position of X.**

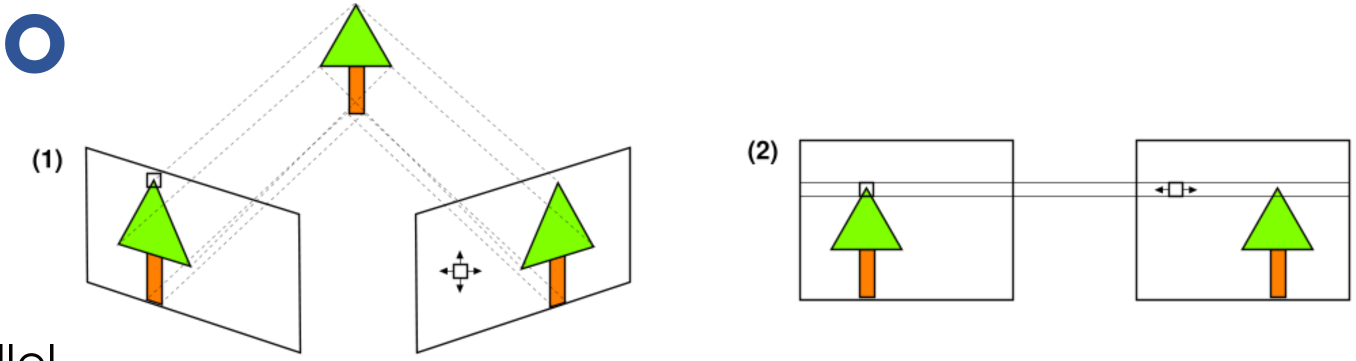


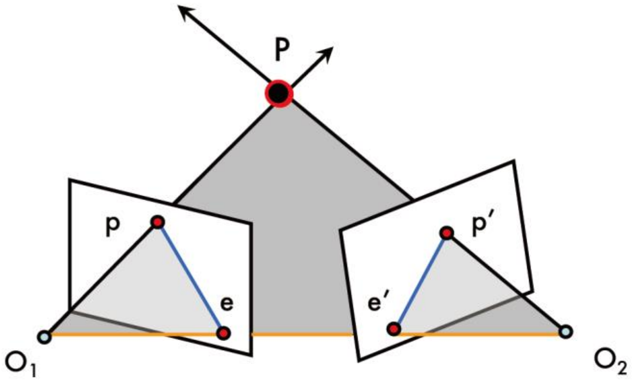
Left View Right View

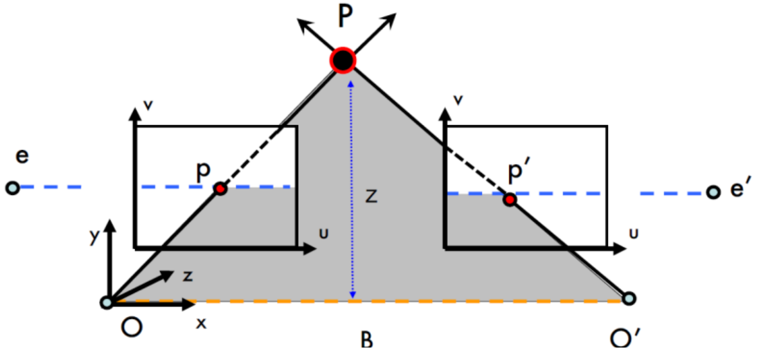
Source: Schairer, Edward, et al. "Measurements of tip vortices from a full-Scale UH-60A rotor by retro-reflective background oriented schlieren and stereo photogrammetry." (2013).

Depth from Stereo
Geometry in stereo

- Epipolar lines can be made parallel through a process called **rectification**
- Simplifies the process of finding a match and calculating the 3D point





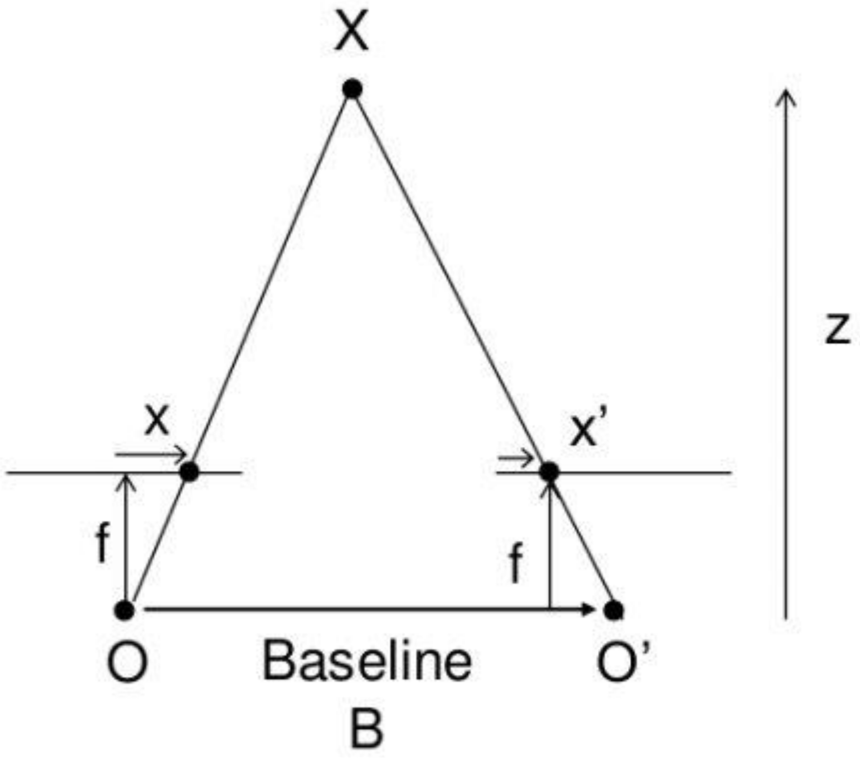


Source: https://www.ivs.auckland.ac.nz/web/calibration.php http://web.stanford.edu/class/cs231a/lectures/lecture6_stereo_systems.pdf

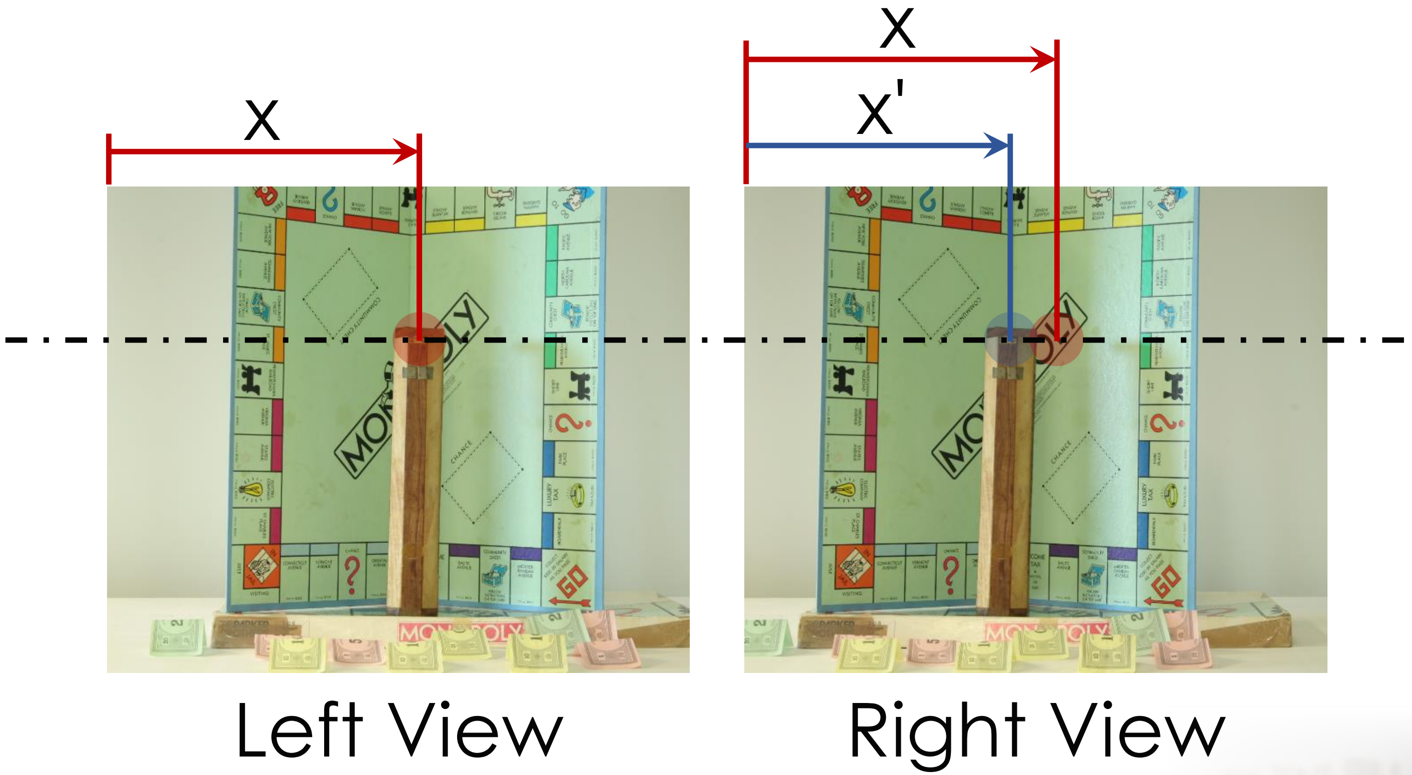
Depth from Stereo
Geometry in stereo

Problem statement, reformulated:
Find the disparity for every pixel in the left (or right) image by finding matches in the right (or left) image

$$disparity = x - x' = \frac{Bf}{Z} \qquad \frac{x - x'}{O - O'} = \frac{f}{Z}$$



Source: https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_calib3d/py_depthmap/py_depthmap.html



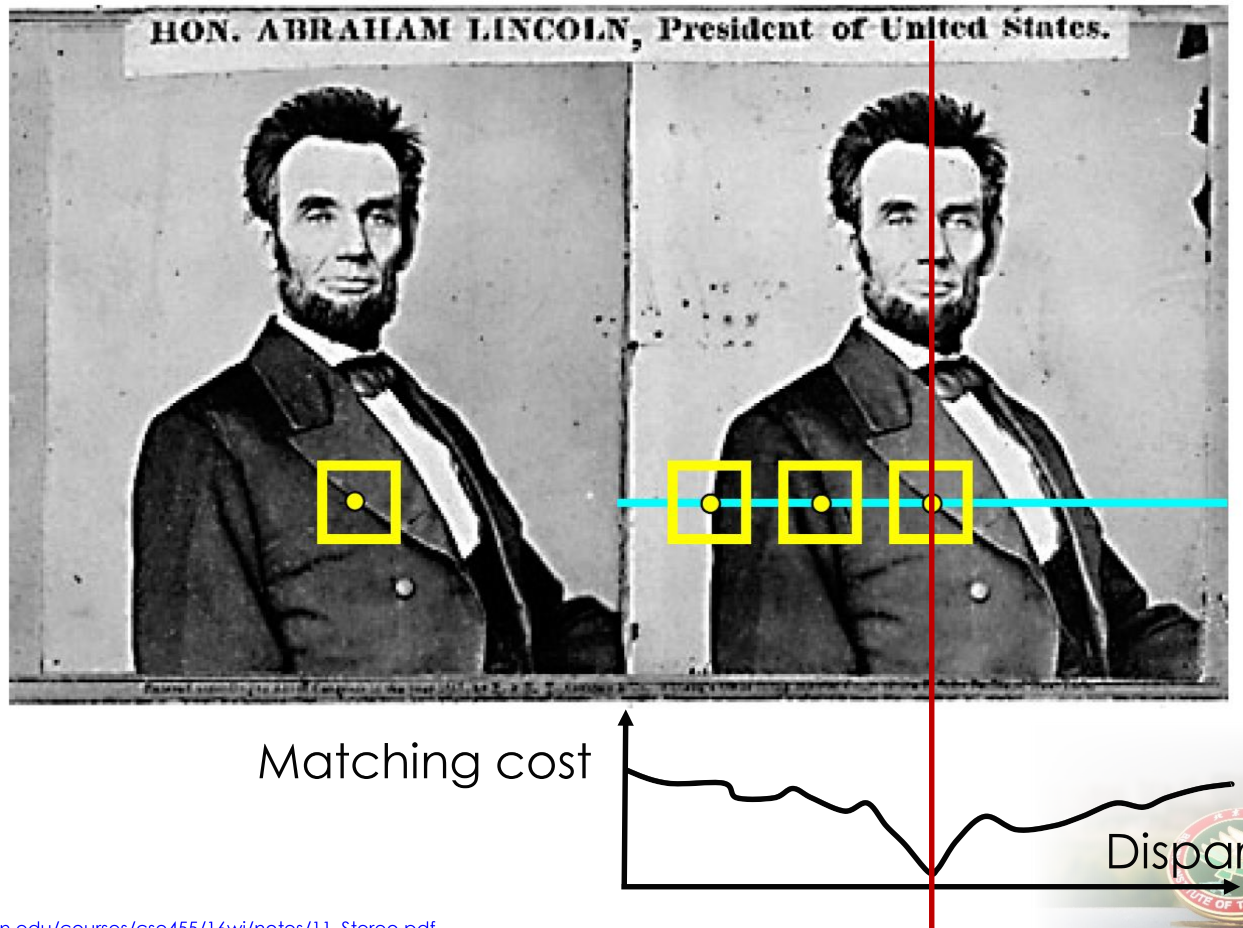
Related Research
Basic stereo matching algorithm

1. If necessary, rectify the two stereo images to transform epipolar lines into scanlines

2. For each pixel x in the first image:

- Find corresponding epipolar scanline in the right image
- Search the scanline and pick the best match x'
- Compute disparity x-x' and set depth(x) = Bf/(x-x')

Correspondence search

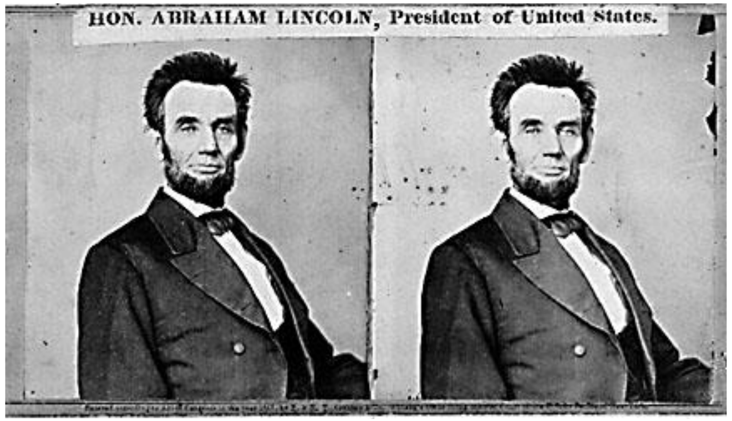


IGTA2019

Source: https://courses.cs.washington.edu/courses/cse455/16wi/notes/11_Stereo.pdf

Related Research

Failures of correspondence search



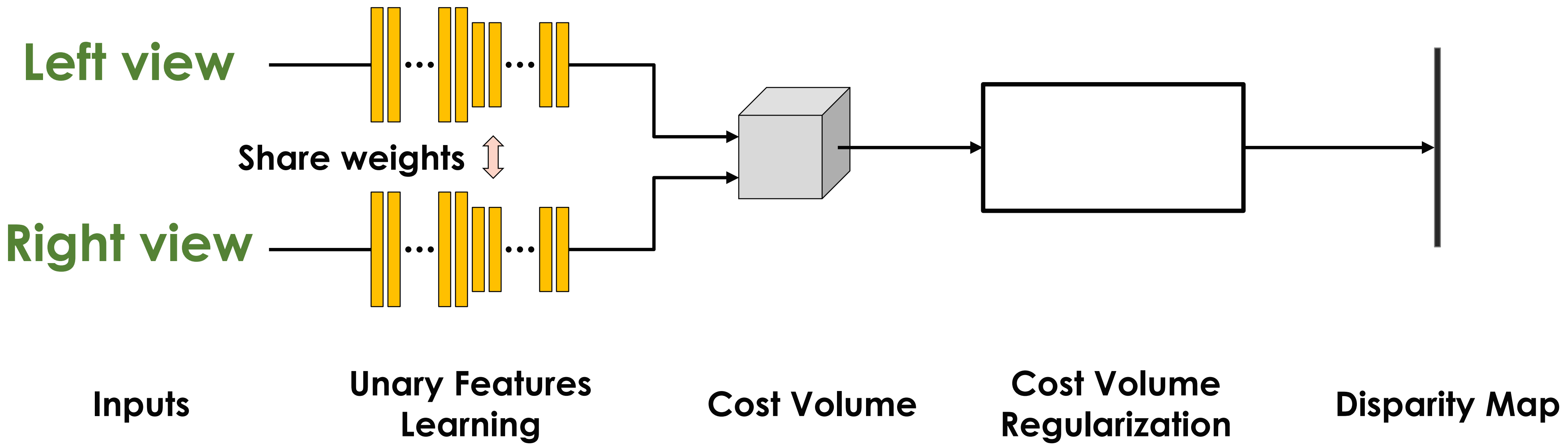
Textureless surfaces

Occlusions, repetition

Non-Lambertian surfaces, specularities

Source: https://courses.cs.washington.edu/courses/cse455/16wi/notes/11_Stereo.pdf

Related Research
Learning-Based Stereo Matching



End-to-end training network

Related Research
GC-Net by Kendall et al.

End-to-End Learning of Geometry and Context for Deep Stereo Regression (ICCV'17)

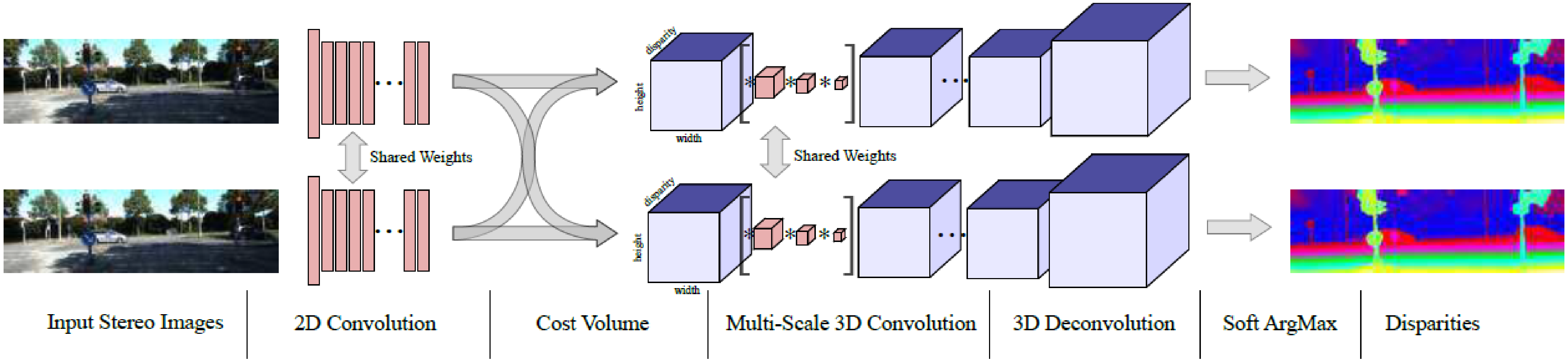
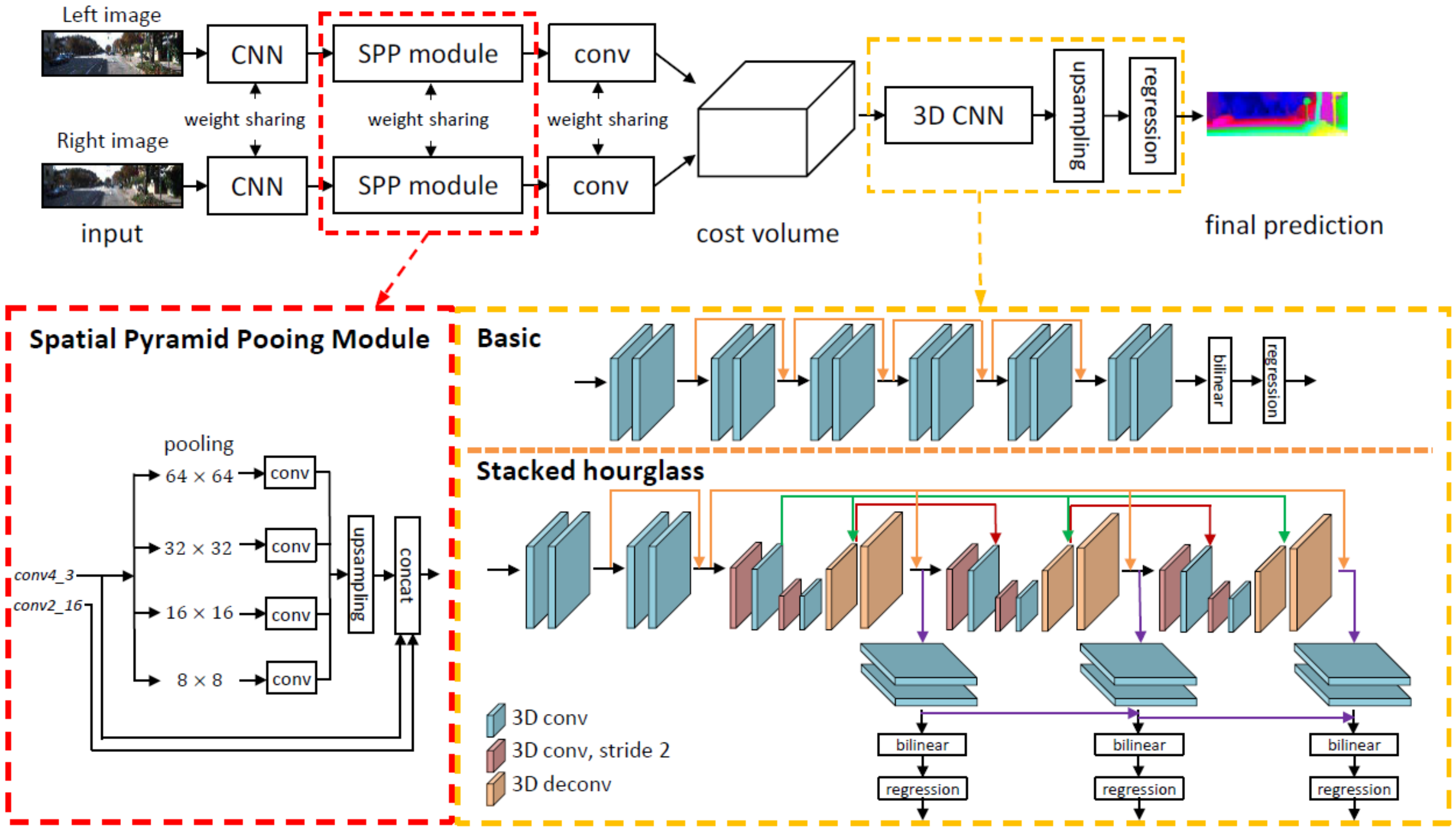


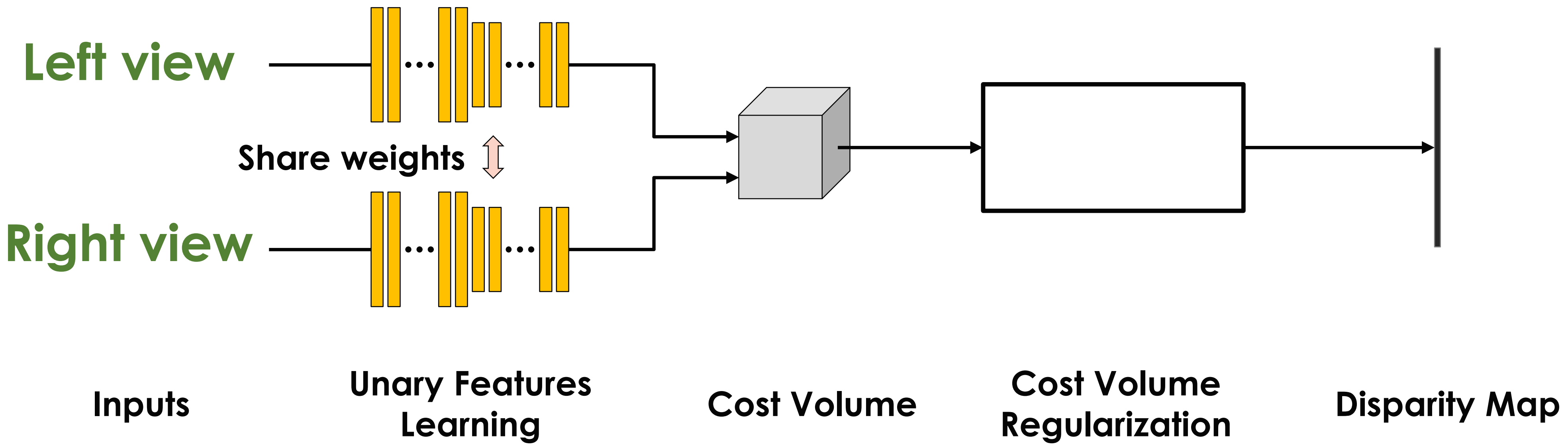
Figure 1: **Our end-to-end deep stereo regression architecture, GC-Net** (Geometry and Context Network).

Related Research
PSM-Net by Chang et al.

Pyramid Stereo Matching Network (CVPR'18)

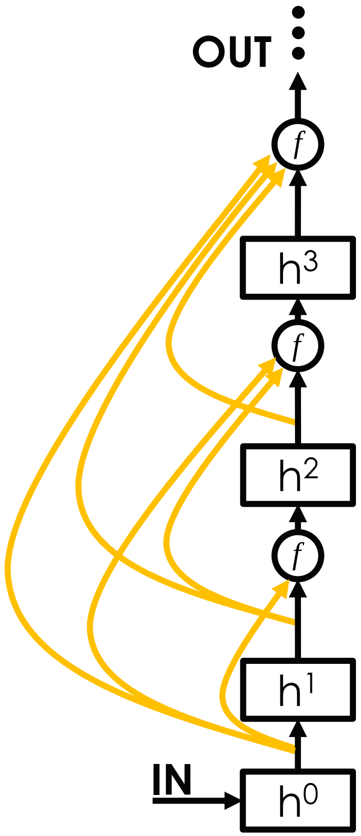


Related Research
Learning-Based Stereo Matching



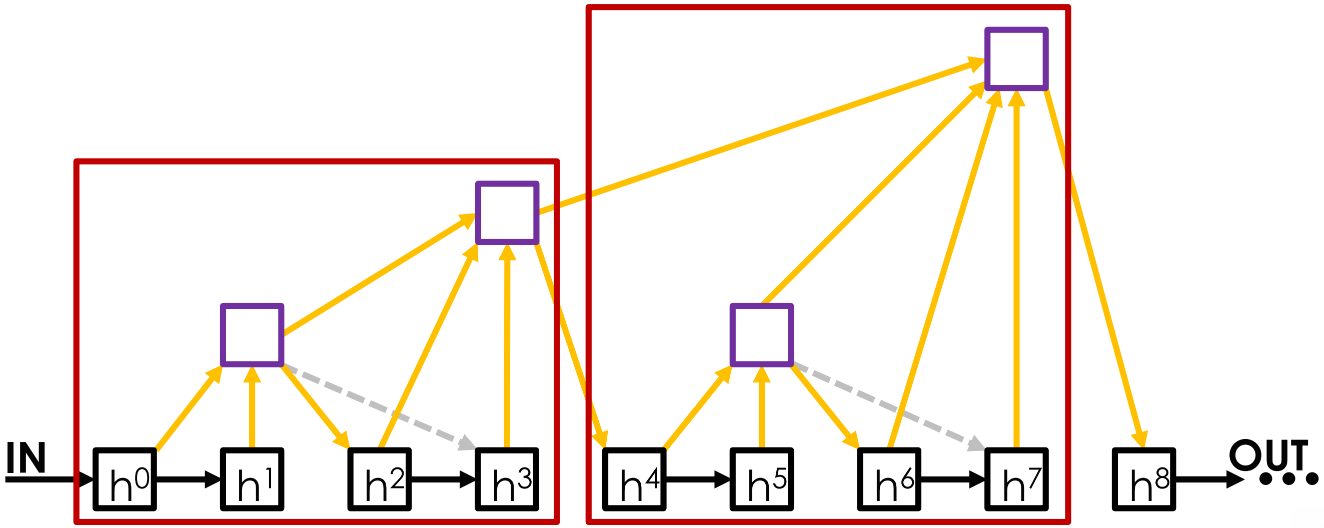
End-to-end training network

Related Research
Different aggregation patterns



(a) DenseNets

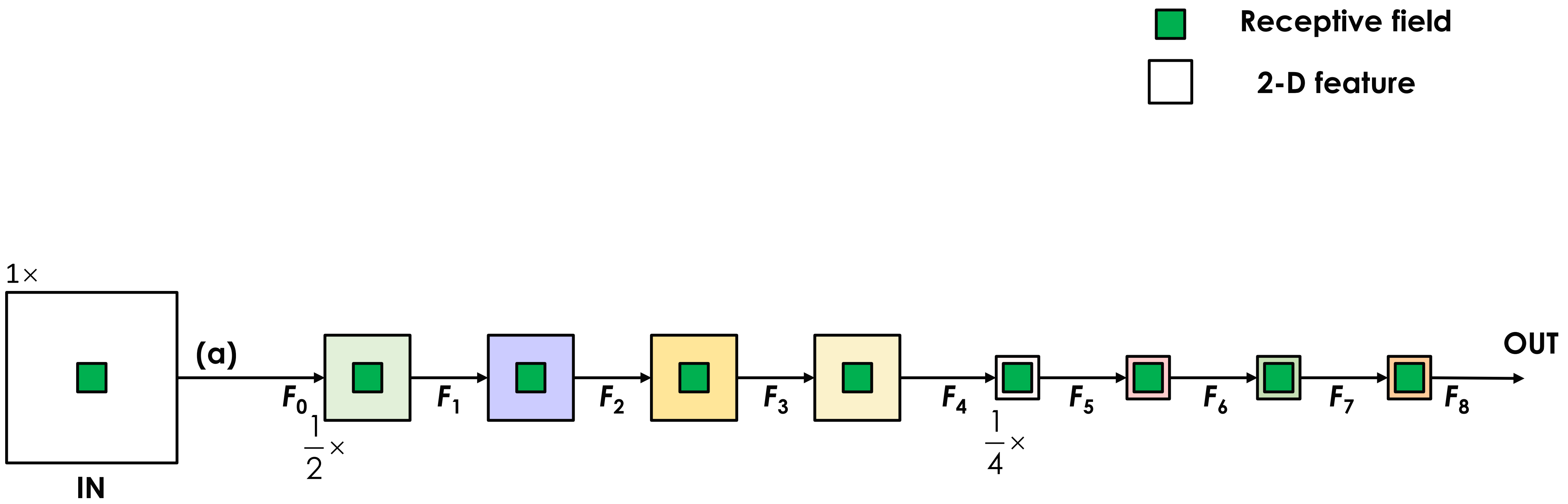
Intra-Level combination



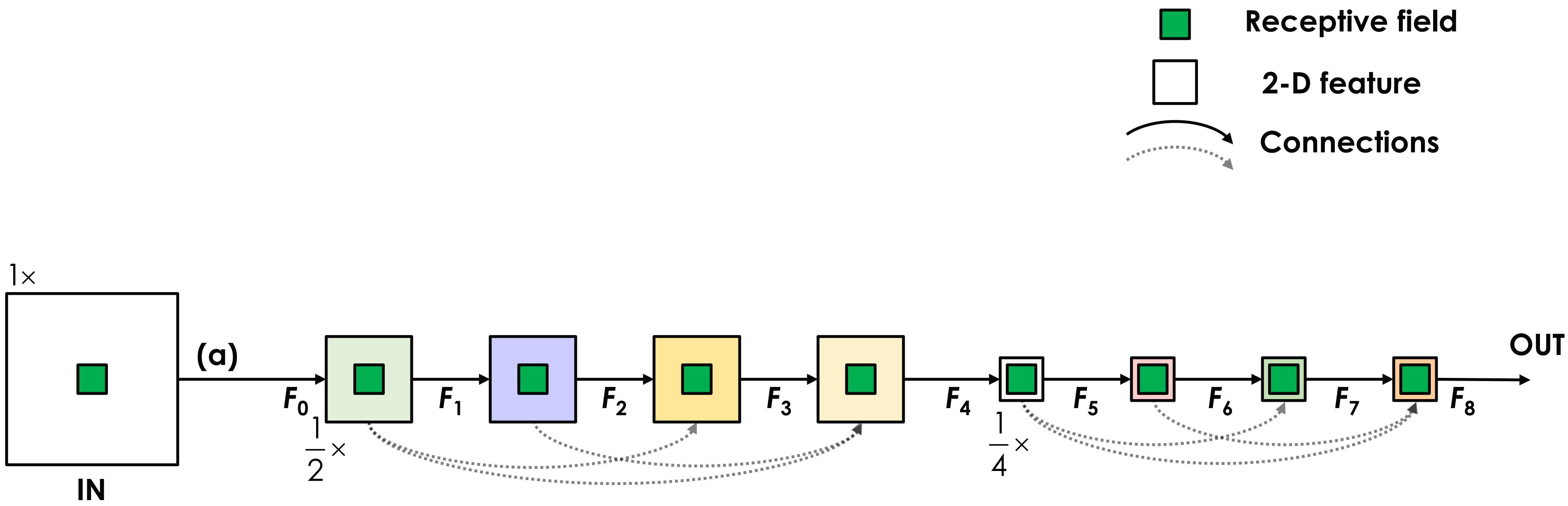
(b) Deep Layer Aggregation

Method/MCUA

Multi-Level **C**ontext **U**ltra-**A**ggregation

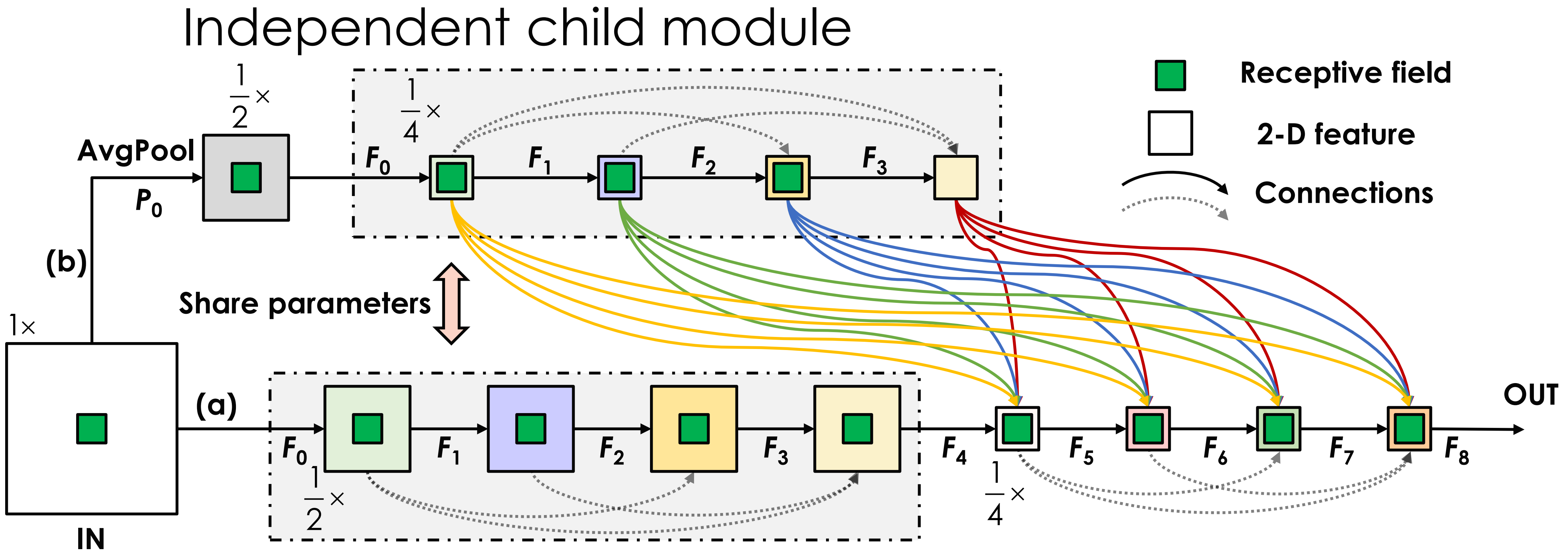


MCUA Intra-Level Combination



IGTA2019

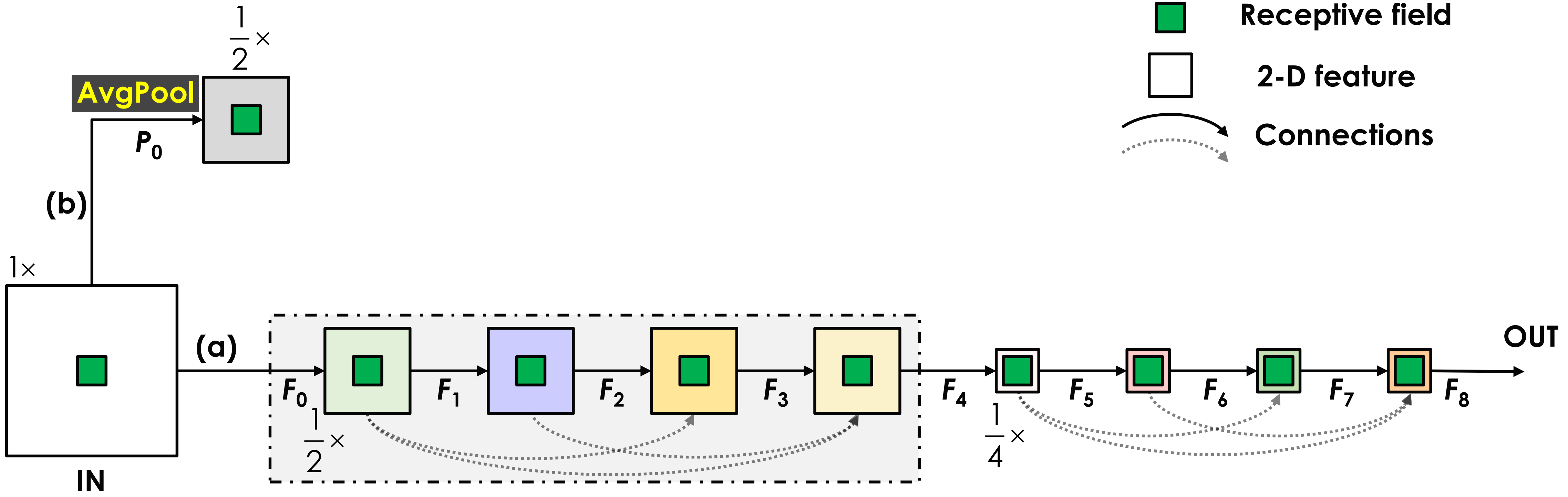
MCUA Inter-Level Combination



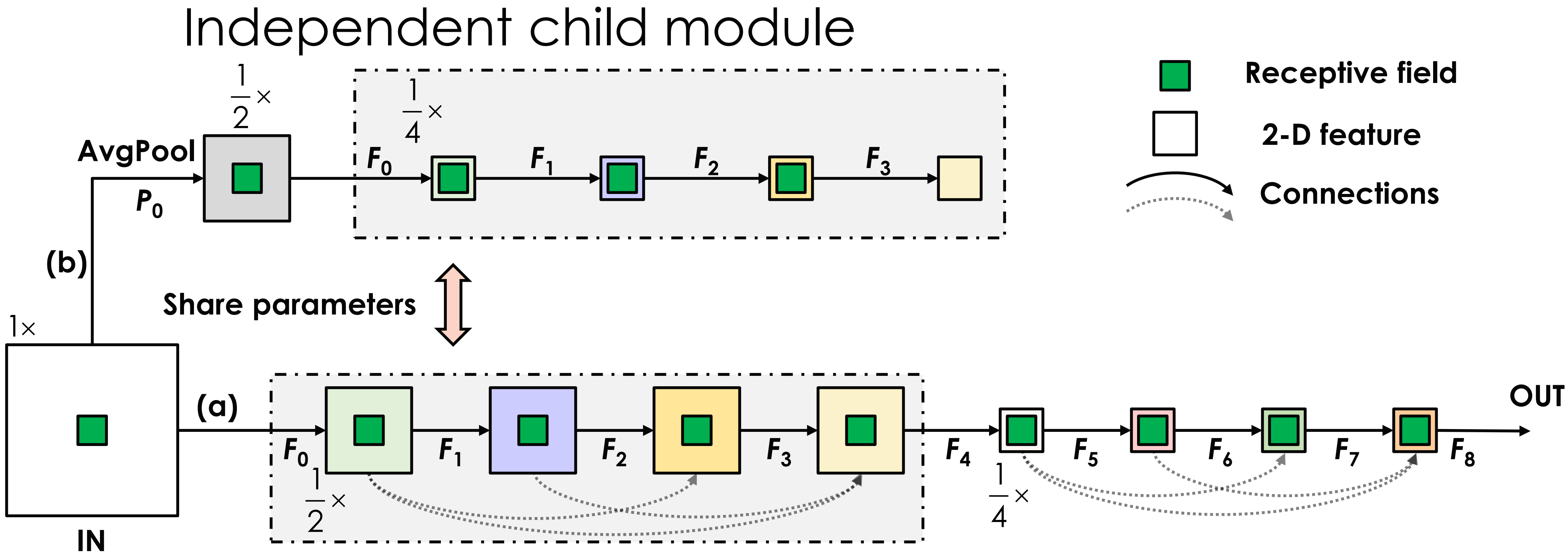
IGTA2019

MCUA Inter-Level Combination

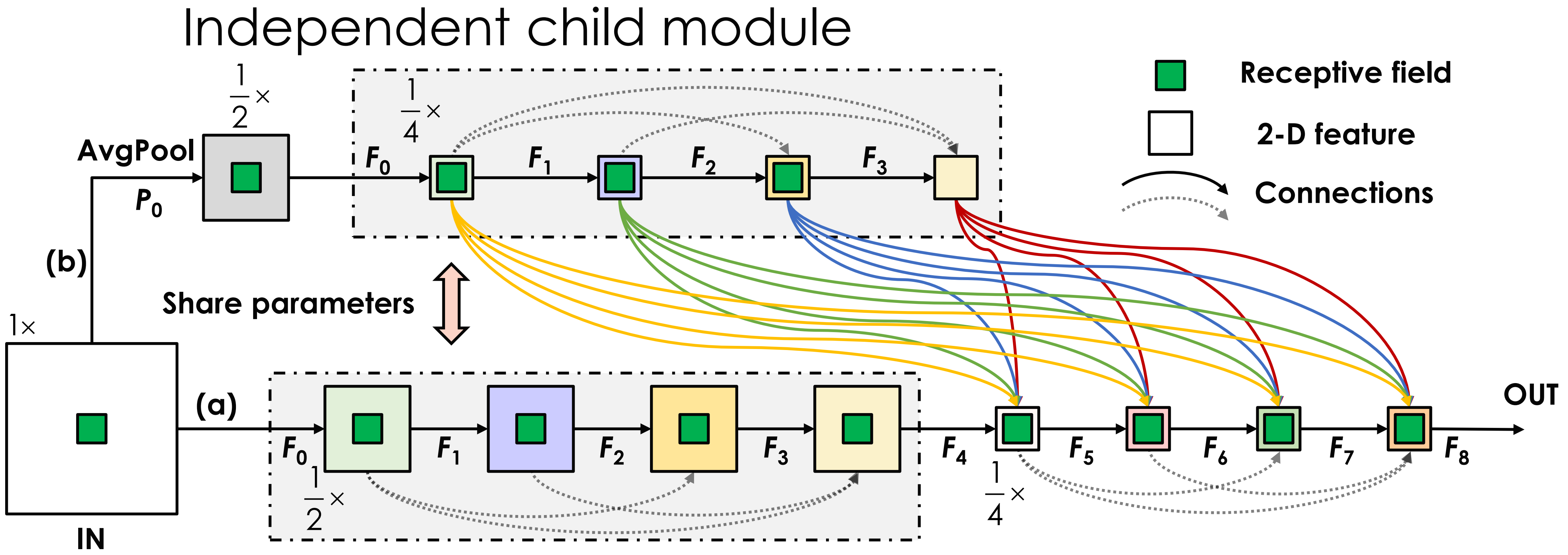
Independent child module



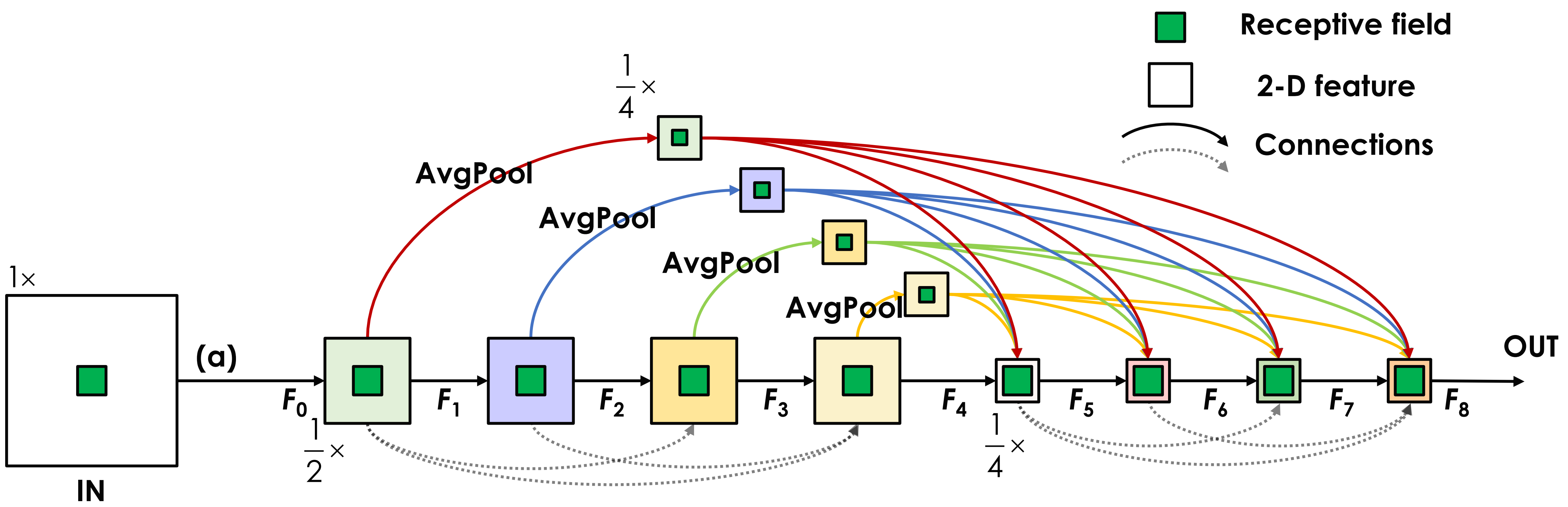
MCUA Inter-Level Combination



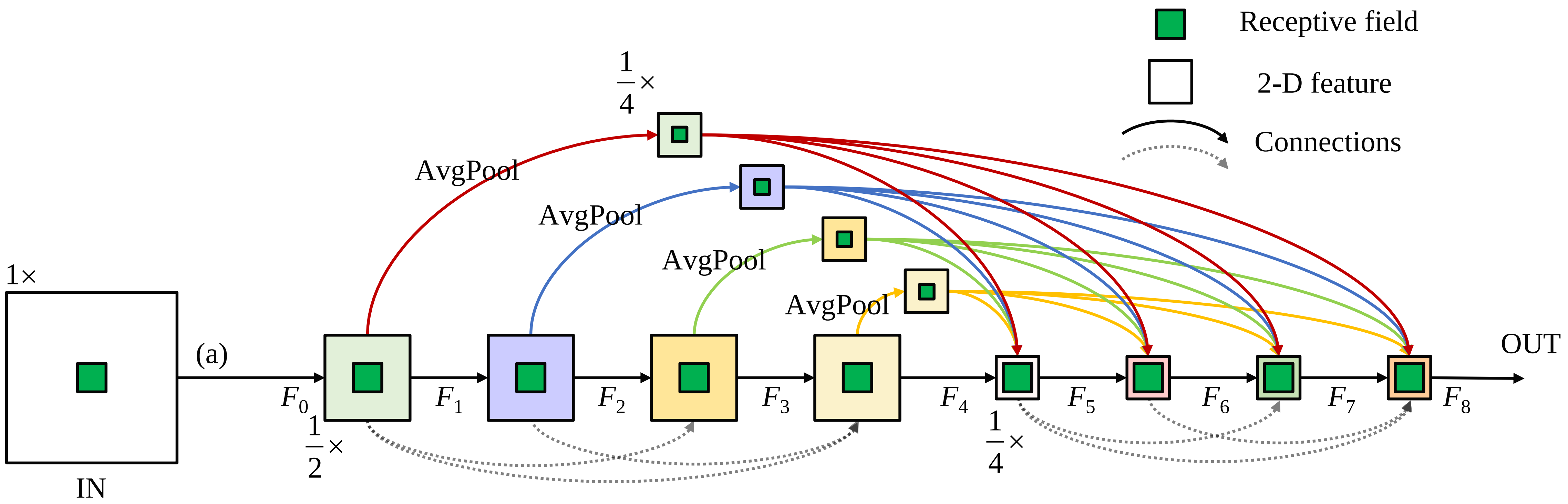
MCUA Inter-Level Combination



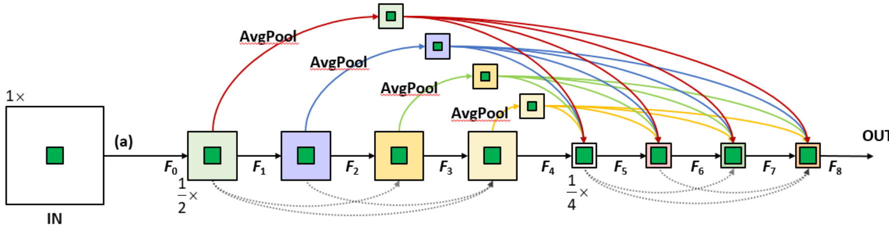
MCUA Dense Connection

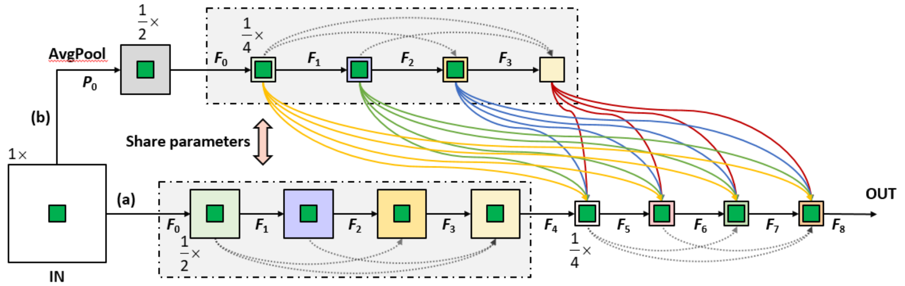


MCUA Dense Connection



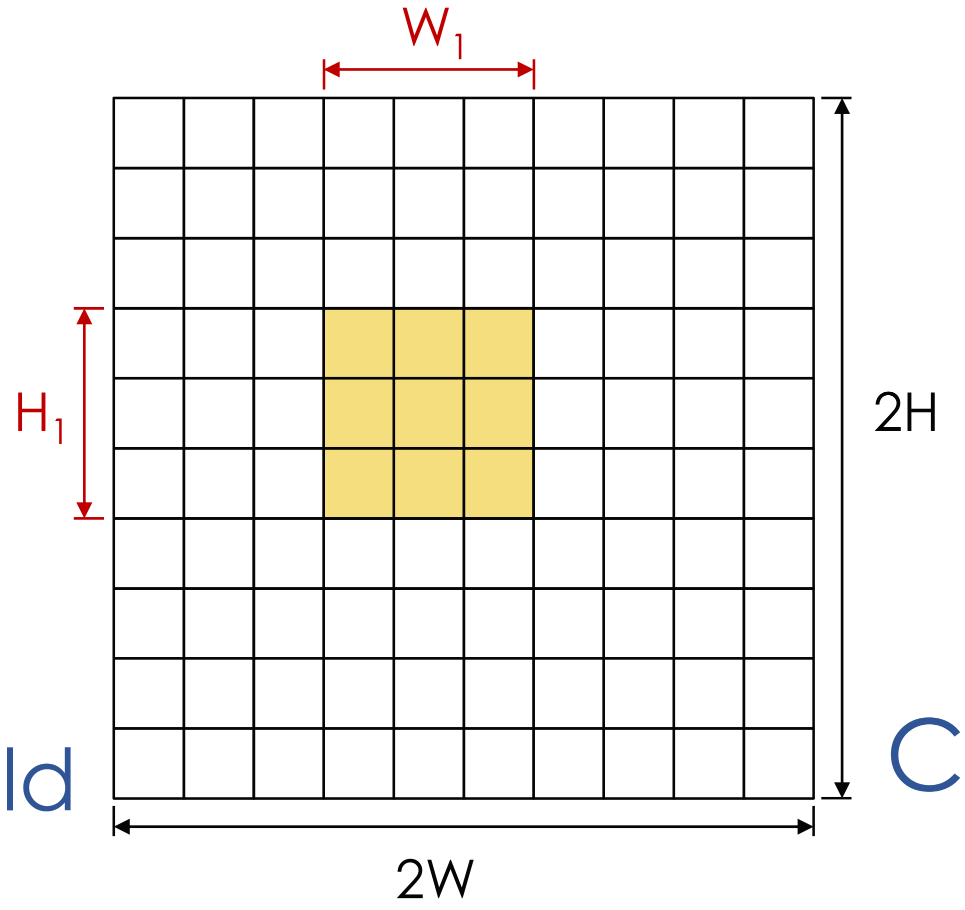
MCUA

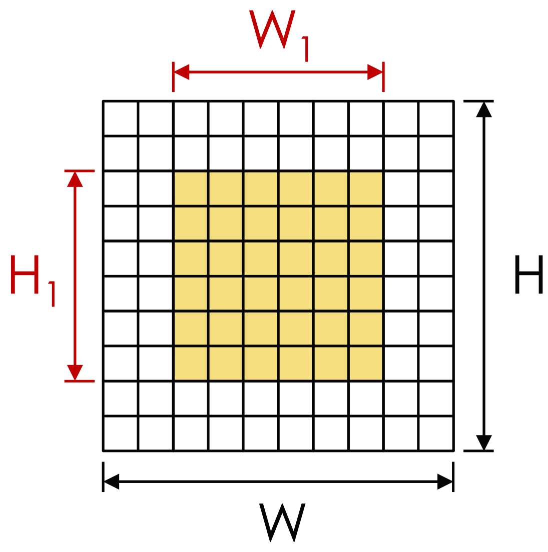




1×1

$\frac{1}{2} \times \frac{1}{2}$

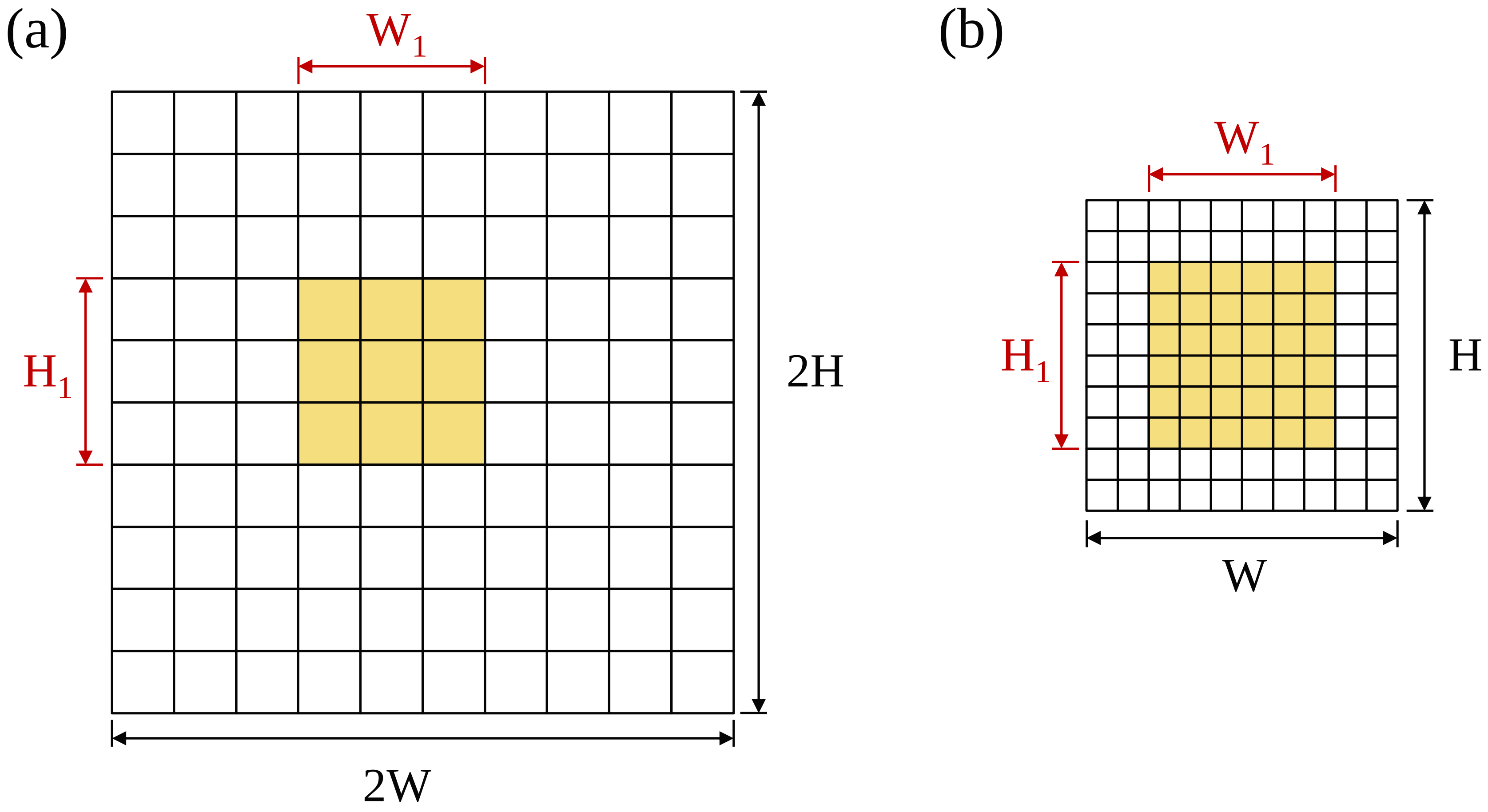




Receptive Field

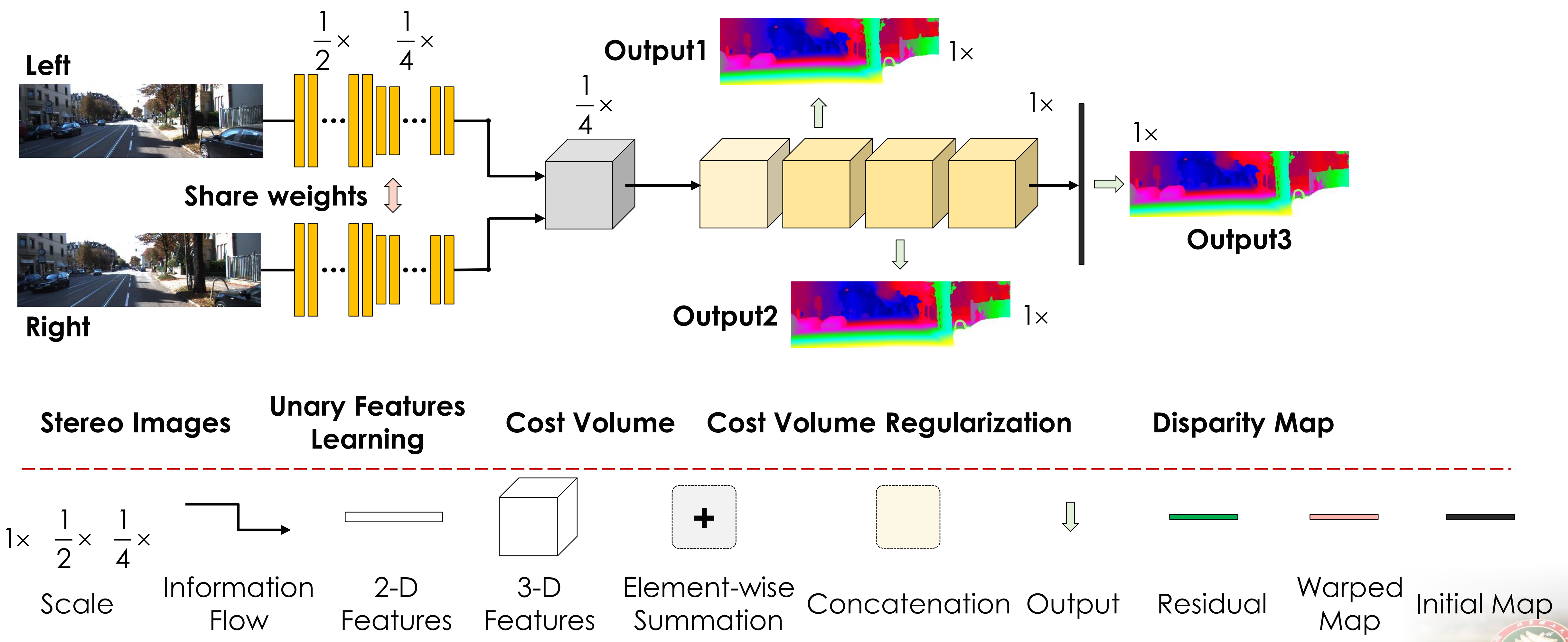
Capture more area

IGTA2019



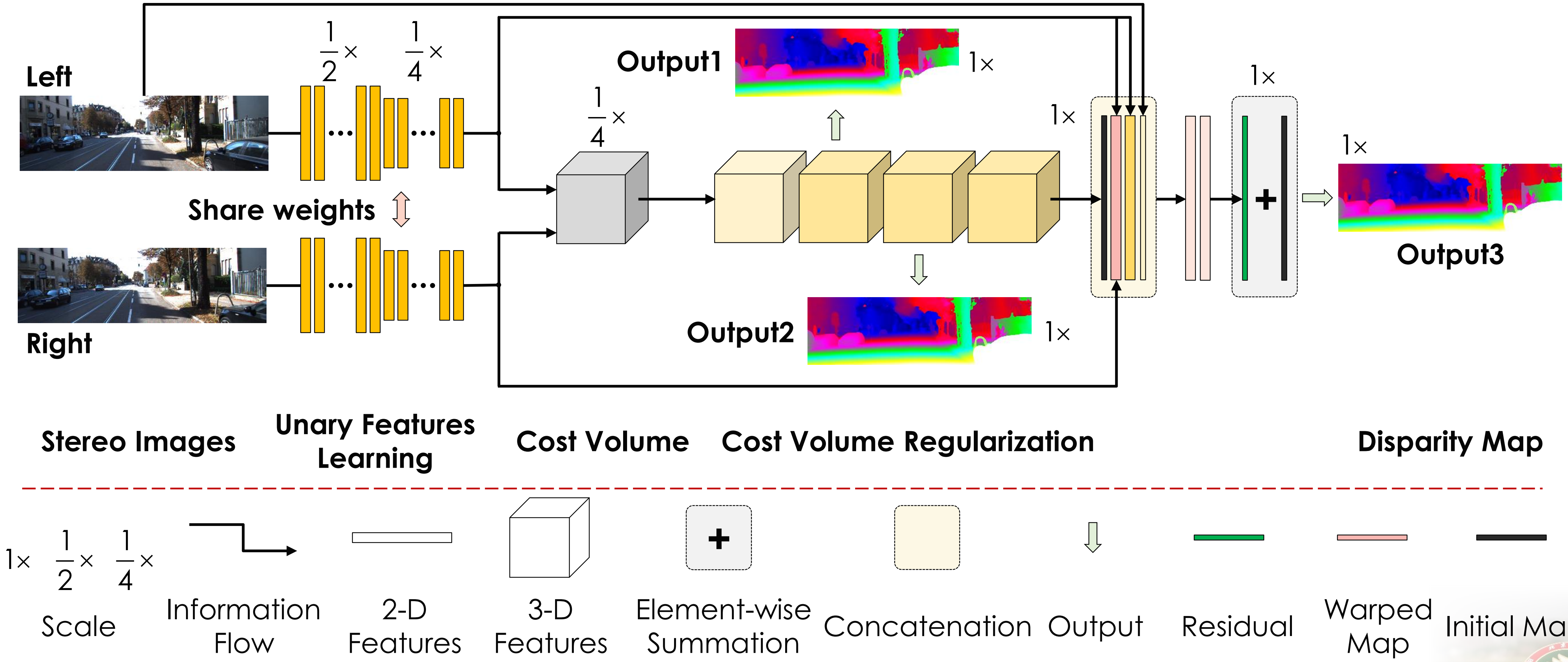
IGTA2019

MCUA Stereo Matching



IGTA2019

EMCUA Stereo Matching



IGTA2019

Experiment **Datasets**

Scene Flow dataset:

FlyingThings3D, Driving, Monkaa

>39000(35454/4370 train/test) stereo frames

960×540 pixel resolution

KITTI2015/2012 datasets

Left view Right view

Disparity map (Ground truth)

KITTI2015: 200/200 train/test stereo images

KITTI2012: 194/200 train/test stereo images

1242×375 pixel resolution

Source: https://lmb.informatik.uni-freiburg.de/resources/datasets/SceneFlowDatasets.en.html
http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=stereo
http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo

Experiment Implementation Details

Train on a lot of data:

- Scene Flow datasets
- Finetuning on KITTI

Test on Flying Things and KITTI

Input: 256×512 pixel resolution

Optimizer: Adam

The training process of EMCUA contains two steps:

- Train MCUA:

  20+50 epochs on SF dataset (lr=0.01)

  600 (lr=0.001) + 400 (lr=0.0001) epochs on KITTI datasets

- Train EMCUA (+ Residual module)

  1 epoch on SF dataset (lr=0.01)

  600 (lr=0.001) + 400 (lr=0.0001) epochs on KITTI datasets

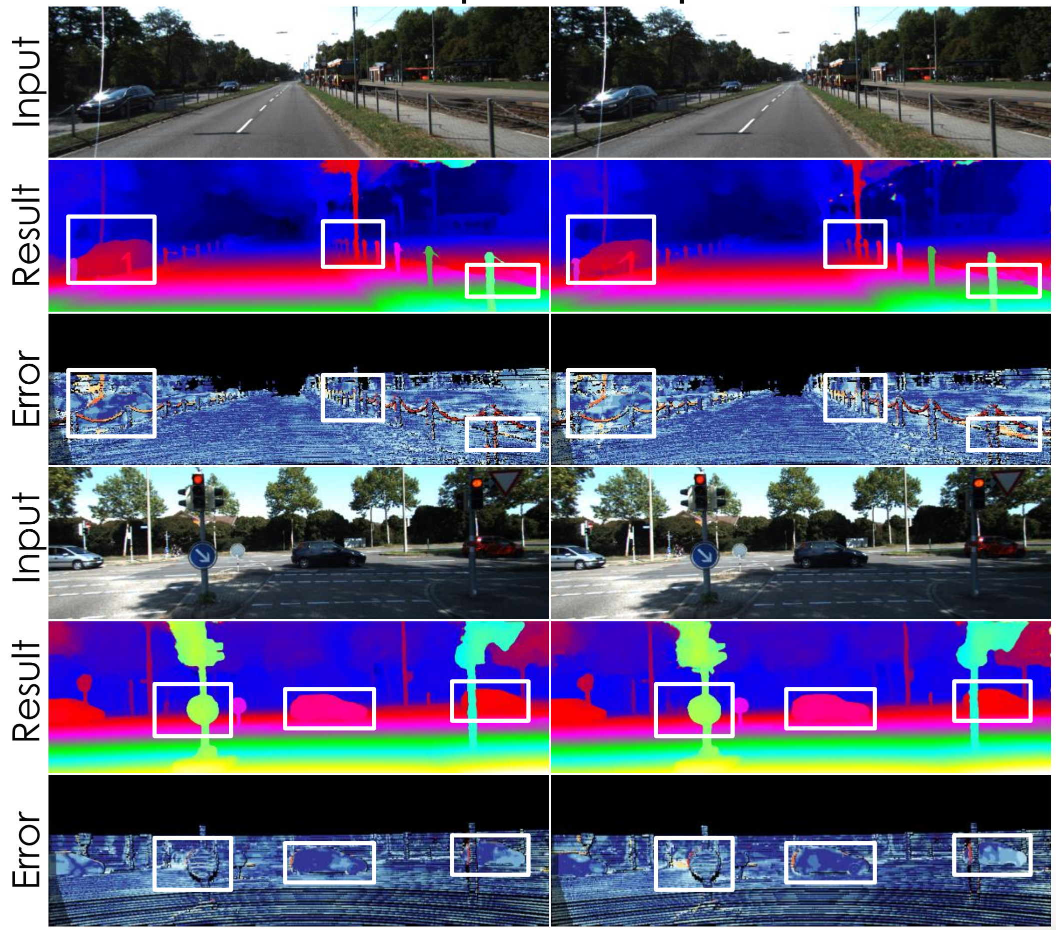
Performance KITTI2015 dataset

Table 2. KITTI2015 Results

Mod.	All (%)			Noc (%)		
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all
SegStereo	1.88	4.07	2.25	1.76	3.70	2.08
iResNet	2.25	**3.40**	2.44	2.07	**2.76**	2.19
CRL	2.48	3.59	2.67	2.32	3.12	2.45
GC-Net [9]	2.21	6.16	2.87	2.02	5.58	2.61
PSM-Net	1.86	4.62	2.32	1.71	4.31	2.14
MCUA	1.69	4.38	2.14	1.55	3.90	1.93
EMCUA	**1.66**	4.27	**2.09**	**1.50**	3.88	**1.90**

"**All**" and "**Noc**" : percentage of outliers averaged over ground truth pixels of all/non-occluded regions. "**D1-bg**", "**D1-fg**", and "**D1-all**": percentage of outliers averaged only over background regions, foreground regions, and all ground truth pixels.

Sample output



(a) EMCUA (b) PSM-Net

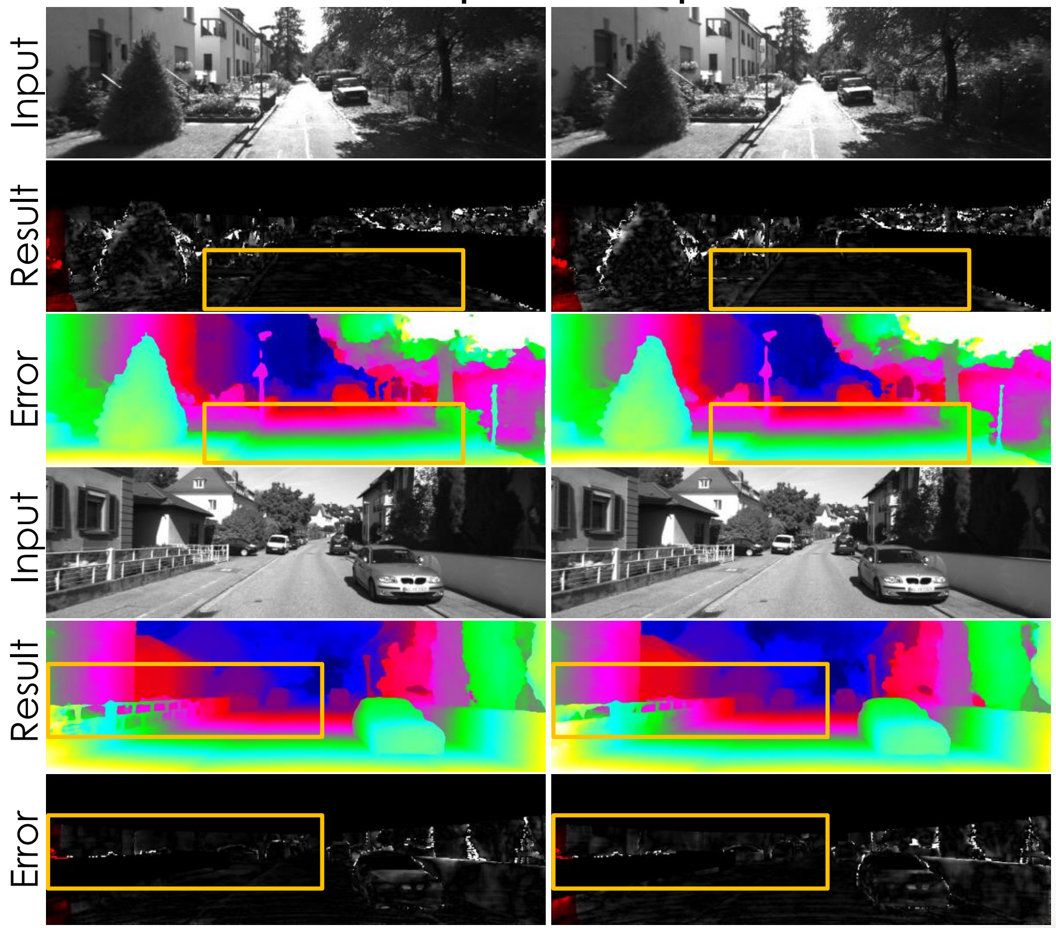
Performance **KITTI2012 dataset**

Table 3. KITTI2012 Results

Mod	$>2px$		$>3px$		$>4px$		$>5px$		ME(px)	
	Noc	All	Noc	All	Noc	All	Noc	All	AN	AA
SegStereo	2.66	3.19	1.68	2.03	1.25	1.52	1.00	1.21	0.5	0.6
iResNet	2.69	3.34	1.71	2.16	1.30	1.63	1.06	1.32	0.5	0.6
GC-Net	2.71	3.46	1.77	2.30	1.36	1.77	1.12	1.46	0.6	0.7
PSM-net	2.44	3.01	1.49	1.89	1.12	1.42	0.90	1.15	0.5	0.6
MCUA	2.07	2.64	1.30	1.70	0.98	1.29	0.80	1.04	0.5	0.5
EMCUA	**2.02**	**2.56**	**1.26**	**1.64**	**0.95**	**1.24**	**0.76**	**0.99**	**0.4**	**0.5**

"**Noc**" and "**All**": percentage of erroneous pixels in non-occluded areas, and in total. "**AN**" and "**AA**": average disparity/end-point error in non-occluded areas, and in total. "**ME**": mean error.

Sample output



(a) EMCUA

(b) PSM-Net

IGTA2019

Performance Residual Module

Table 2. KITTI2015 Results

Mod.	All (%)			Noc (%)		
	D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all
SegStereo	1.88	4.07	2.25	1.76	3.70	2.08
iResNet	2.25	**3.40**	2.44	2.07	**2.76**	2.19
CRL	2.48	3.59	2.67	2.32	3.12	2.45
GC-Net [9]	2.21	6.16	2.87	2.02	5.58	2.61
PSM-Net	1.86	4.62	2.32	1.71	4.31	2.14
MCUA	1.69	4.38	2.14	1.55	3.90	1.93
EMCUA	**1.66**	4.27	**2.09**	**1.50**	3.88	**1.90**

"**All**" and "**Noc**" : percentage of outliers averaged over ground truth pixels of all/non-occluded regions. "**D1-bg**", "**D1-fg**", and "**D1-all**": percentage of outliers averaged only over background regions, foreground regions, and all ground truth pixels.

Table 3. KITTI2012 Results

Mod	$> 2px$		$> 3px$		$> 4px$		$> 5px$		ME(px)	
	Noc	All	Noc	All	Noc	All	Noc	All	AN	AA
SegStereo	2.66	3.19	1.68	2.03	1.25	1.52	1.00	1.21	0.5	0.6
iResNet	2.69	3.34	1.71	2.16	1.30	1.63	1.06	1.32	0.5	0.6
GC-Net	2.71	3.46	1.77	2.30	1.36	1.77	1.12	1.46	0.6	0.7
PSM-net	2.44	3.01	1.49	1.89	1.12	1.42	0.90	1.15	0.5	0.6
MCUA	2.07	2.64	1.30	1.70	0.98	1.29	0.80	1.04	0.5	0.5
EMCUA	**2.02**	**2.56**	**1.26**	**1.64**	**0.95**	**1.24**	**0.76**	**0.99**	**0.4**	**0.5**

"**Noc**" and "**All**": percentage of erroneous pixels in non-occluded areas, and in total. "**AN**" and "**AA**": average disparity/end-point error in non-occluded areas, and in total. "**ME**": mean error.

Residual module is mainly used to improve the performance of the accuracy of the foreground.

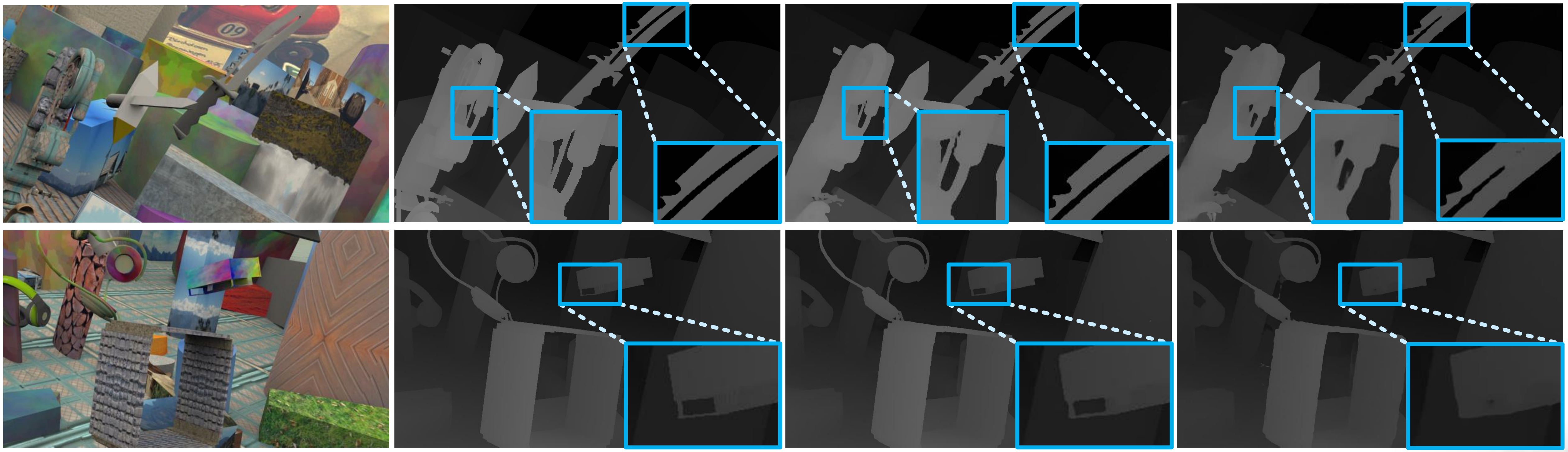
Performance Scene Flow Datasets

Table 4. Performance comparison on Scene Flow test set

Mod.	EPE	Mod.	EPE	Mod.	EPE
MCUA	**0.56**	PSM-Net [2]	1.09	StereoNet [10]	1.10
CRL. [18]	1.32	iResNet [11]	1.40	SegStereo [24]	1.45

Mod.: model; **EPE**: end-point-error;

Sample output



Inputs Ground truth **MCUA** PSM-Net

IGTA2019

Discussion

Different aggregation schemes

- Dense connection
- Deep Layer Aggregation
- MCUA

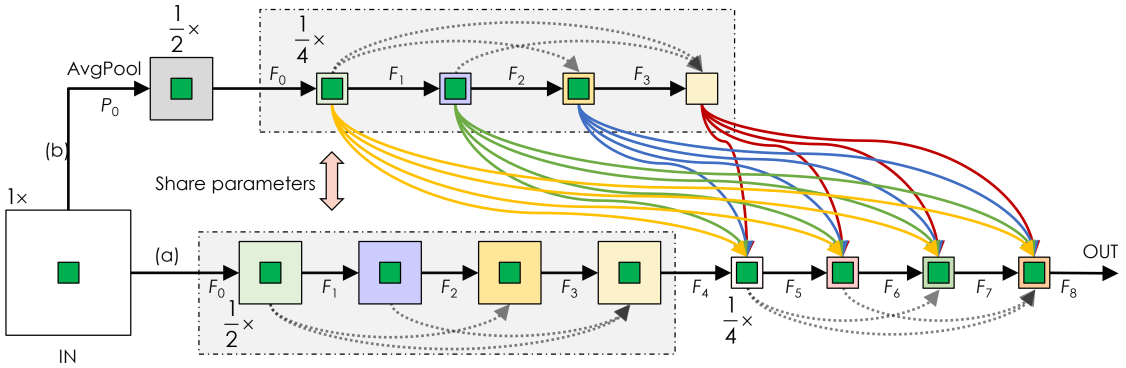
Table 5. Ablation study

Mod.	Scene Flow				KITTI2015	Para.
	$> 1px$	$> 3px$	$> 5px$	EPE	VE (%)	
Compare of aggregation patterns						
PSM-Net	–	–	–	1.119	1.83	5.22M
DenseNets	8.526	3.329	2.286	0.794	1.698	5.27M
DLA	8.586	3.337	2.280	0.806	1.685	5.32M
MCUA	**7.885**	**3.108**	**2.148**	**0.758**	**1.579**	5.31M
Compare of architecture components						
UChi	8.185	3.153	**2.147**	**0.755**	1.635	5.39M
Chi	8.133	3.242	2.226	0.777	1.642	5.29M
DenPool	8.187	3.187	2.179	0.761	1.628	5.31M
MCUA	**7.885**	**3.108**	2.148	0.758	**1.579**	5.31M

$>$ ***tpx***: EPE; **VE**: three-pixel-error; **Para.**: number of parameters.

Discussion

Effect of MCUA



MCUA

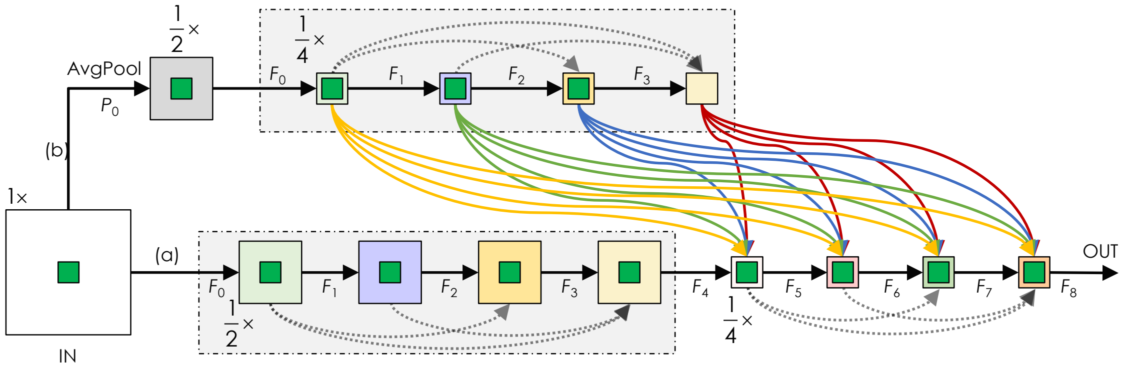
Table 5. Ablation study

Mod.	Scene Flow				KITTI2015	Para.
	$> 1px$	$> 3px$	$> 5px$	EPE	VE (%)	
Compare of aggregation patterns						
PSM-Net	–	–	–	1.119	1.83	5.22M
DenseNets	8.526	3.329	2.286	0.794	1.698	5.27M
DLA	8.586	3.337	2.280	0.806	1.685	5.32M
MCUA	**7.885**	**3.108**	**2.148**	**0.758**	**1.579**	5.31M
Compare of architecture components						
UChi	8.185	3.153	**2.147**	**0.755**	1.635	5.39M
Chi	8.133	3.242	2.226	0.777	1.642	5.29M
DenPool	8.187	3.187	2.179	0.761	1.628	5.31M
MCUA	**7.885**	**3.108**	2.148	0.758	**1.579**	5.31M

$>$ ***tpx***: EPE; **VE**: three-pixel-error; **Para.**: number of parameters.

Discussion

Effect of MCUA



UChi

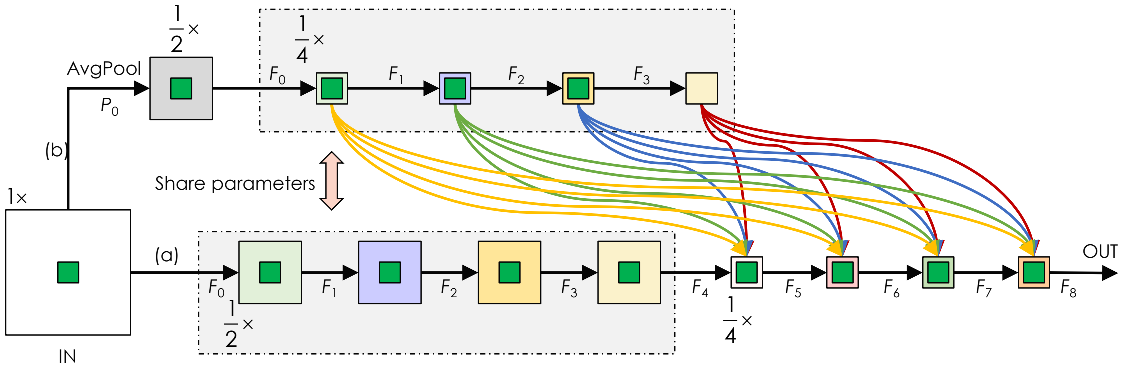
Table 5. Ablation study

Mod.	Scene Flow				KITTI2015	Para.
	$> 1px$	$> 3px$	$> 5px$	EPE	VE (%)	
Compare of aggregation patterns						
PSM-Net	–	–	–	1.119	1.83	5.22M
DenseNets	8.526	3.329	2.286	0.794	1.698	5.27M
DLA	8.586	3.337	2.280	0.806	1.685	5.32M
MCUA	**7.885**	**3.108**	**2.148**	**0.758**	**1.579**	5.31M
Compare of architecture components						
UChi	8.185	3.153	**2.147**	**0.755**	1.635	5.39M
Chi	8.133	3.242	2.226	0.777	1.642	5.29M
DenPool	8.187	3.187	2.179	0.761	1.628	5.31M
MCUA	**7.885**	**3.108**	2.148	0.758	**1.579**	5.31M

$>$ ***tpx***: EPE; **VE**: three-pixel-error; **Para.**: number of parameters.

Discussion

Effect of MCUA



Chi

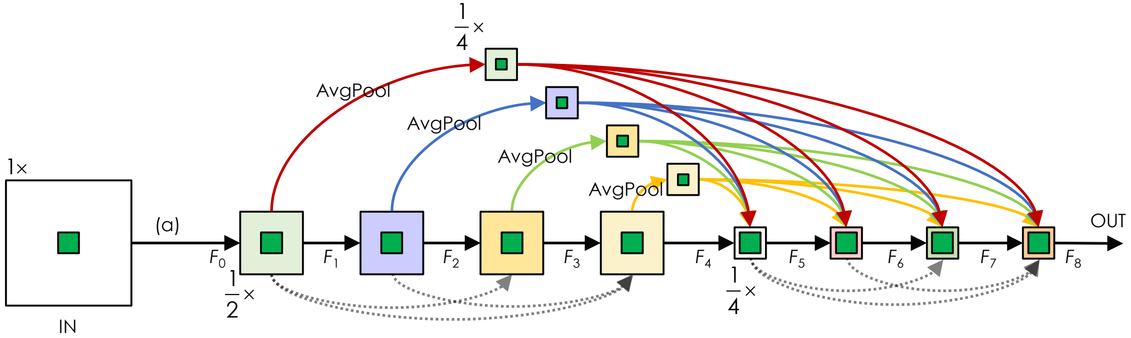
Table 5. Ablation study

Mod.	Scene Flow				KITTI2015	Para.
	$> 1px$	$> 3px$	$> 5px$	EPE	VE (%)	
Compare of aggregation patterns						
PSM-Net	–	–	–	1.119	1.83	5.22M
DenseNets	8.526	3.329	2.286	0.794	1.698	5.27M
DLA	8.586	3.337	2.280	0.806	1.685	5.32M
MCUA	**7.885**	**3.108**	**2.148**	**0.758**	**1.579**	5.31M
Compare of architecture components						
UChi	8.185	3.153	**2.147**	**0.755**	1.635	5.39M
Chi	8.133	3.242	2.226	0.777	1.642	5.29M
DenPool	8.187	3.187	2.179	0.761	1.628	5.31M
MCUA	**7.885**	**3.108**	2.148	0.758	**1.579**	5.31M

$>$ ***tpx***: EPE; **VE**: three-pixel-error; **Para.**: number of parameters.

Discussion

Effect of MCUA



DenPool

Table 5. Ablation study

Mod.	Scene Flow				KITTI2015	Para.
	$> 1px$	$> 3px$	$> 5px$	EPE	VE (%)	
Compare of aggregation patterns						
PSM-Net	–	–	–	1.119	1.83	5.22M
DenseNets	8.526	3.329	2.286	0.794	1.698	5.27M
DLA	8.586	3.337	2.280	0.806	1.685	5.32M
MCUA	**7.885**	**3.108**	**2.148**	**0.758**	**1.579**	5.31M
Compare of architecture components						
UChi	8.185	3.153	**2.147**	**0.755**	1.635	5.39M
Chi	8.133	3.242	2.226	0.777	1.642	5.29M
DenPool	8.187	3.187	2.179	0.761	1.628	5.31M
MCUA	**7.885**	**3.108**	2.148	0.758	**1.579**	5.31M

$>$ ***tpx***: EPE; **VE**: three-pixel-error; **Para.**: number of parameters.

Conclusion

- We propose a general feature aggregation scheme, MCUA, which contains both intra- and inter-level feature aggregation, while DenseNets and DLA contain only intra-level aggregation.
- We use an independent child module to introduce inter-level aggregation, which enlarges the receptive fields and captures more context information.

Future work

- Dataset bias (~~Stereo matching~~ Depth estimation)
- Real-time stereo matching

Future work **Datasets**

Scene Flow dataset:

FlyingThings3D, Driving, Monkaa

>39000(35454/4370 train/test) stereo frames

960×540 pixel resolution

KITTI2015/2012 datasets

Left view Right view

Disparity map (Ground truth)

KITTI2015: 200/200 train/test stereo images

KITTI2012: 194/200 train/test stereo images

1242×375 pixel resolution

Source: https://lmb.informatik.uni-freiburg.de/resources/datasets/SceneFlowDatasets.en.html
http://www.cvlibs.net/datasets/kitti/eval_scene_flow.php?benchmark=stereo
http://www.cvlibs.net/datasets/kitti/eval_stereo_flow.php?benchmark=stereo

Future work

- Dataset bias (~~Stereo matching~~ Depth estimation)
- Real-time stereo matching

Framework of traditional stereo vision algorithm

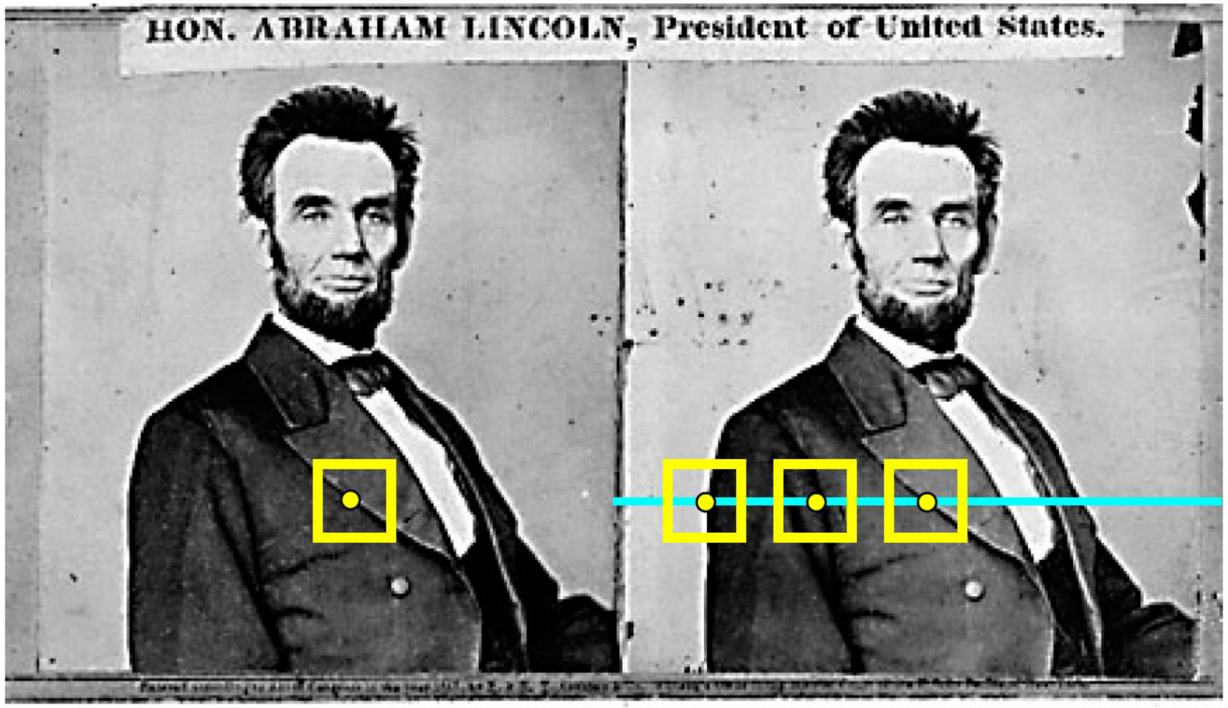
Input images ⇒ Step1: Matching cost computaion ⇒ Step2: Cost Aggregation ⇒ Step3: Disparity Selection ⇒ Step4: Disparity refinement ⇒ Disparity map

Matching cost: SSD, SAD, or normalized correlation

$$SSD(x,y,d) = \sum_{(x,y)\in w} \left| I_l(x,y) - I_r(x-d,y) \right|^2$$

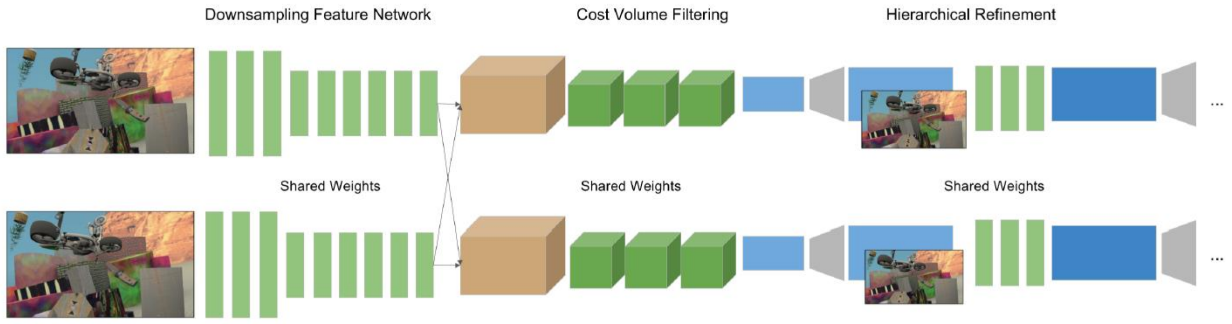
Source: A. Fusiello, U. Castellani, and V. Murino, “Relaxing symmetric multiple windows stereo using Markov Random Fields,” in *Computer Vision and Pattern Recognition*, vol. 2134 of *Lecture Notes in Computer Science*, pp. 91–105, Springer, 2001.

Correspondence search



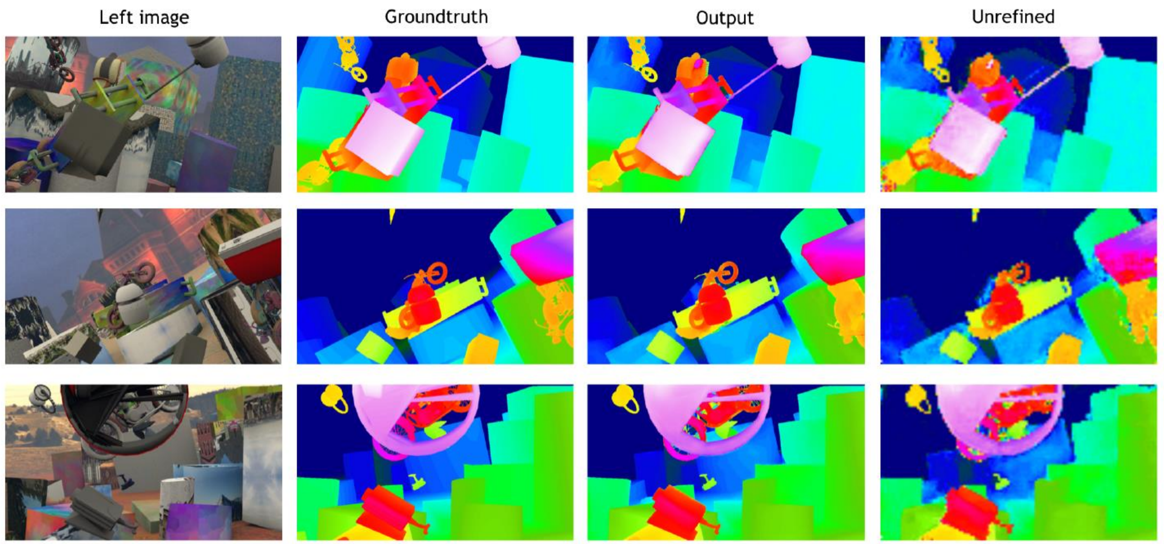
Future work

- Dataset bias (~~Stereo matching~~ Depth estimation)
- Real-time stereo matching



StereoNet architecture (ECCV'18)

Source: Khamis, Sameh, et al. "Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction." Proceedings of the European Conference on Computer Vision (ECCV). 2018.



Qualitative results on the FlyingThings3D test set

北京理工大學
BEIJING INSTITUTE OF TECHNOLOGY

Beijing Engineering Research Center
of Mixed Reality and Advanced Display

德以明理
学以精工

Thanks for your watching.

北京理工大學
BEIJING INSTITUTE OF TECHNOLOGY

Beijing Engineering Research Center
of Mixed Reality and Advanced Display

德以明理 学以精工

Q&A

Guang-Yu Nie

guyuneeeee@outlook.com

IGTA2019

04/19-20/2019