IMAGE PROCESSING ON LINE

Published in Image Processing On Line on 2023–07–16.
Submitted on 2022–10–12, accepted on 2022–10–13.
ISSN 2105–1232
This article is available online with supplementary materials, software, datasets and online demo at
https://doi.org/10.5201/ipol.2023.441

An Overview of GANet – Guided Aggregation Net for End-to-end Stereo Matching

Alvaro Gómez

Facultad de Ingeniería, Universidad de la República, Uruguay
agomez@fing.edu.uy

Communicated by Jean-Michel Morel *Demo edited by* Alvaro Gómez

Abstract

Guided Aggregation Net for End-to-end Stereo Matching (GANet) is a stereo matching method that uses Deep Neural Networks (DNN) to compute a disparity map from a pair of images of a scene. As other classic and DNN stereo methods, it follows the traditional stereo steps: dense features are extracted from both images, the cost of matching the features at different disparities is organized in a Cost Volume (CV) which is regularized by aggregation and local filtering and finally a map with minimal cost is derived from the CV. In GANet, the aggregation of the CV is done by a Semi-Global Guided Aggregation layer (SGA) which implements a differentiable approximation of the well known Semi-Global Matching (SGM) algorithm. SGA is followed by a Local Guided Aggregation layer (LGA) that performs a local filtering. SGA and LGA weights are generated by an auxiliary guidance subnet fed with the original reference image and its extracted features. This article presents an overview of GANet. An online demo, running on CPU, is made available.

Source Code

The source code and documentation for this algorithm are available from the web page of this article[1]. Usage instructions are included in the README file of the archive. The original implementation of the method is available here[2].
This is an MLBriefs article, the source code has not been reviewed!

Keywords: stereo matching; disparity map; cost volume; aggregation

[1] https://doi.org/10.5201/ipol.2023.441
[2] https://github.com/feihuzhang/GANet

1 Introduction

Stereo vision is an area that has been extensively researched and multiple algorithms have been proposed over the last decades [23, 12, 16]. Given two images of a scene from different known viewpoints, the objective of stereo is to estimate the most likely 3D shape or depth that explains those images. The change in viewpoint induces a relative displacement of the objects in the scene causing that closer objects move more than far ones in the images of the pair. This apparent motion between the two views (disparity) is inversely proportional to the depth.

In [23], the authors point out that most stereo algorithms perform these four steps: (1) matching cost computation, (2) cost aggregation, (3) disparity computation, (4) disparity refinement.

The first step implies finding sparse or dense correspondences between the images. In the sparse case, characteristic points along with their local features are extracted and compared. In the dense approach, image patches in both images are compared computing the cost of matching the patches for different possible disparities. The search of corresponding patches is simplified by the geometric constraints of the stereo pair (epipolar constraints). Instead of a 2D search for correspondences, the epipolar constraints restrict the search for corresponding image points from the entire image plane to a single line. Moreover, the images can be resampled (stereo-rectification) in such a way that corresponding points are located on the same row.

The matching information is organized usually in a cost volume that stores the costs $C_p(d)$ of matching the position p of the reference image with $p + d$ in the second image for all the considered possible disparity values d.

Matching at the correct disparity is challenging in real life due to the photometric and geometric distortions introduced by the change of viewpoint and by ambiguities due to occlusions, low texture or repetitive patterns in the scene. The step of cost aggregation tries to overcome this difficulty by imposing spatial coherence to the matching. This can be done by a simple local filtering of the cost volume or, in a more comprehensive approach, by formulating a global energy minimization problem with a regularization term that enforces the regularity of the disparity map.

Once the cost volume has been regularized, the disparity values can be estimated by processing the volume using argmin (usually mentioned as winner-takes-all), soft-argmin or a maximum a posteriori approximation.

The resulting disparity map may still have erroneous and missing values and several algorithms (filtering, interpolation, inpainting and others) for the post-processing of depth and/or disparity maps have been proposed in the literature [1].

1.1 Global Energy Minimization Methods

This section presents an overview of global energy minimization methods based on [8], where the reader is referred to for more details.

Global methods formulate stereo matching as a global energy minimization problem that includes a regularity term. The energy E is defined on the graph $G = (\mathcal{V}, \mathcal{E})$

$$E(\mathrm{D}) = \sum_{\mathbf{p}\in\mathcal{V}} C_{\mathbf{p}}(\mathrm{D}_{\mathbf{p}}) + \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{E}} V(\mathrm{D}_{\mathbf{p}}, \mathrm{D}_{\mathbf{q}}), \tag{1}$$

where $C_{\mathbf{p}}(d)$ is a unary data term that represents the pixel-wise cost of matching $\mathbf{p}$ with disparity $d \in \mathcal{D}$ (the cost volume), where $\mathcal{D} = \{d_{\min}, \cdots, d_{\max}\}$ defined on a discrete search space (often denoted label set). The pairwise terms $V(\mathrm{D}_{\mathbf{p}}, \mathrm{D}_{\mathbf{q}})$ enforce smoothness of the solution by penalizing changes of neighboring disparities on the edge set $\mathcal{E}$, which is usually the 4-connected image graph. Popular choices of regularity are

$$V(d, d) = |d - d'|, \tag{2}$$

or

$$V(d,d') = \begin{cases} 0 & \text{if } d = d' \\ P1 & \text{if } |d-d'| = 1 \\ P2 & \text{otherwise} \end{cases}. \tag{3}$$

The latter imposes a small penalty $P1$ for small jumps in disparity (up to one pixel), which are common on slanted surfaces, and a constant penalty $P2$ (with $P2 > P1$) accounts for larger disparity jumps.

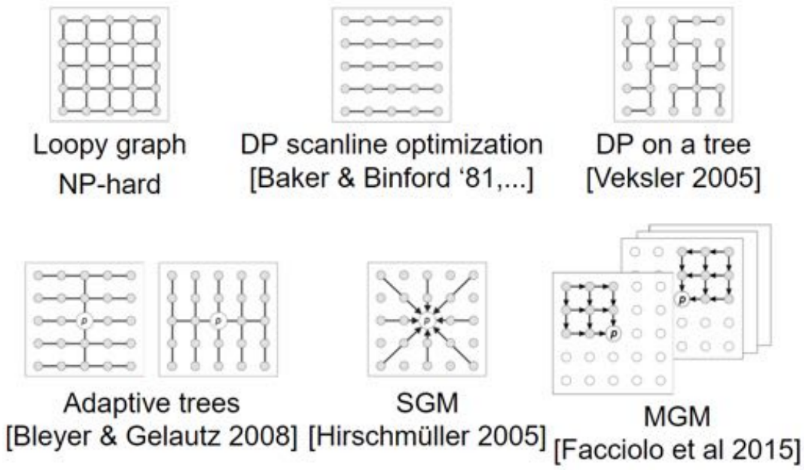


Figure 1: Approximations of the 2D MRF energy using trees [2, 4, 13, 9, 24]. Reproduced from [8].

The exact minimization of energy (1) on a 2D graph is NP-hard, except for some particular cases [20, 15].

On the other hand, when defined on acyclic graphs, the energy (1) can be minimized exactly in polynomial time using dynamic programming.

Tree-based dynamic programming approaches allow to incorporate more regularity (illustrated in Figure 1), leading to better approximations of the problem (1). Some methods build a single tree that spans the entire image [24]. Others construct trees that vary their grid structure with the position of the pixel [4, 13, 9]. The Semi-Global Matching (SGM) algorithm [13] is equivalent to optimizing an energy restricted to a star-shaped graph centered at the current pixel. Even though these algorithms do not yield the most accurate reconstructions, they produce very fast and high-quality results.

1.2 Semi-Global Matching Algorithm

Semi-Global matching [13] proposes to approximately minimize energy (1) with the smoothness term of (3). Semi-Global matching approximation consists in dividing the grid-shaped problem into multiple (N_{dir}) one-dimensional problems defined on scanlines, which are straight lines that run through the image in 4, 8 or 16 cardinal directions (illustrated in Figure 2). For simplicity, here we will consider only $N_{dir} = 4$ directions.

For each cardinal direction $\mathbf{r} \in \{(1,0),(-1,0),(0,1),(0,-1)\}$ SGM computes a matrix of costs $C^A_{\mathbf{r}}$. The costs $C^A_{\mathbf{r}}(\mathbf{p},d)$ are computed recursively starting from the image borders along a path in the direction $\mathbf{r}$

$$C^A_{\mathbf{r}}(\mathbf{p},d) = C_{\mathbf{p}}(d) + \min_{d' \in \mathcal{D}}(C^A_{\mathbf{r}}(\mathbf{p}-\mathbf{r},d') + V(d,d')). \tag{4}$$

This recursion is in fact a dynamic programming algorithm that solves the problem restricted to the directed graph induced by the scanline $\mathbf{p} - \mathbb{N}\mathbf{r} = \{\mathbf{p} - k\mathbf{r} | k \in \mathbb{N}\}$.

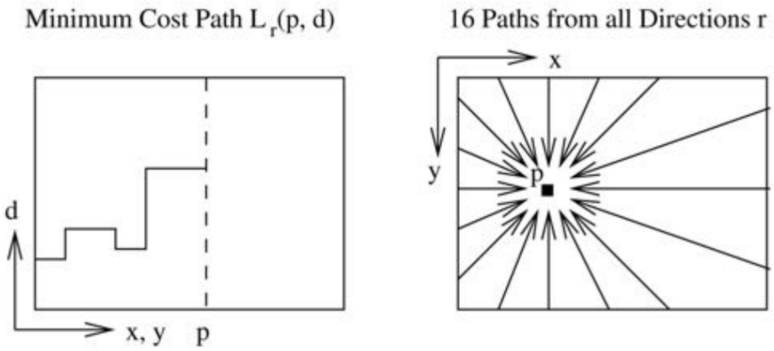


Figure 2: Semi-Global matching aggregates the results of scanline optimization performed along 8 or 16 different orientations. This is equivalent to solving the problem restricted to a star-shaped graph associated to each pixel. Figures reproduced from [13]. Caption text reproduced from [8].

In the case of SGM, with the regularity term as in (3), the aggregated cost volume along each of the directions can be computed as

$$C^A_{\mathbf{r}}(\mathbf{p}, d) = C(\mathbf{p}, d) + \min \begin{cases} C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d), \\ C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d-1) + P_1, \\ C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d+1) + P_1, \\ \min_i C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, i) + P_2. \end{cases} \tag{5}$$

These costs computed in each direction $\mathbf{r}$ are then added to obtain an aggregated cost volume

$$S(\mathbf{p}, d) = \sum_{\mathbf{r}} C^A_{\mathbf{r}}(\mathbf{p}, d) - (N_{dir} - 1) C_{\mathbf{p}}(d). \tag{6}$$

The subtraction of $(N_{dir} - 1) C_{\mathbf{p}}(d)$ is an over-counting correction analogous to the correction proposed by Drory et al. in [7] and that is not present in the original SGM description [13].

The final disparity for each pixel is then selected by winner-takes-all with respect to d on the aggregated cost $S(\mathbf{p}, d)$. This amounts to minimizing a different problem at each pixel defined as a restriction of energy (1) to the star-shaped graph illustrated in Figure 2.

2 GANet Method

The method addressed in this article, Guided Aggregation Net for End-to-end Stereo Matching (GANet) [25] is a stereo matching method that uses Deep Neural Networks (DNN) to compute a disparity map. Figure 3 depicts the architecture overview.

As other DNN methods [16] it follows the traditional stereo steps: dense features are extracted from both images, the cost of matching the features at different disparities is organized in a Cost Volume (CV), which is regularized by aggregation and local filtering and finally a map with minimal cost is derived from the CV.

In most DNN based stereo methods, cost aggregation is done by 3D convolutions, usually in an hourglass configuration [16]. 3D convolutions imply large memory requirements; the computational burden restricts the size of the images that can be processed.

GANet, despite using also some 3D convolutions, takes a different approach for the aggregation by introducing a Semi-Global Guided Aggregation layer (SGA) which implements a differentiable approximation of Semi-Global Matching (SGM) [14]. SGA is followed by a Local Guided Aggregation

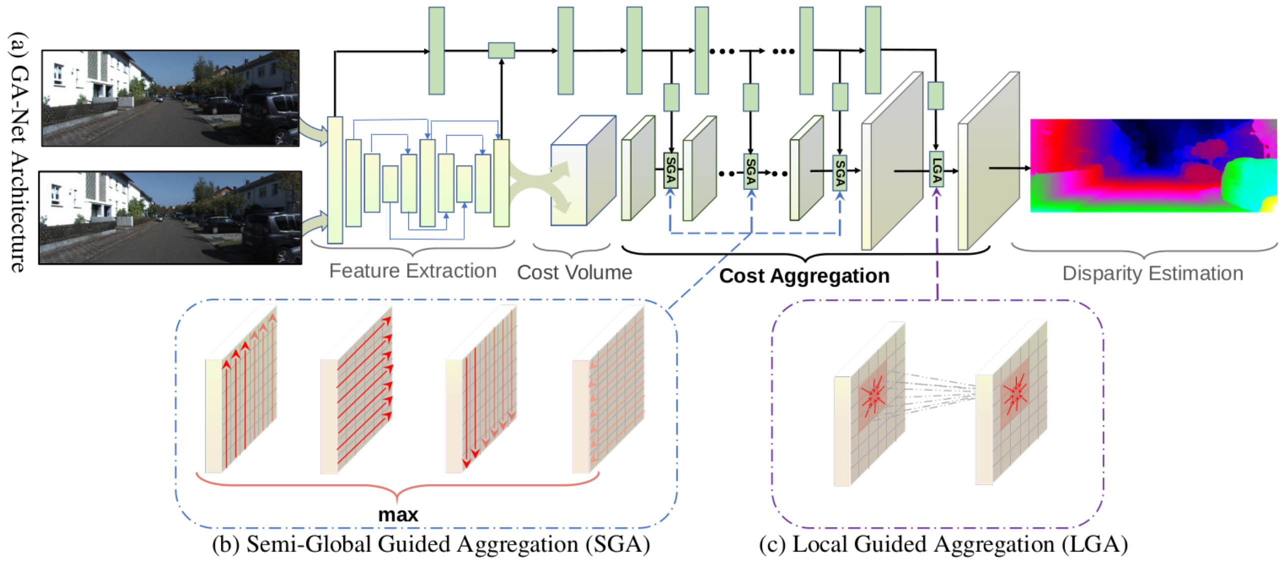


Figure 3: GANet architecture overview.
Reproduced from [25].

layer (LGA) that performs a local filtering. SGA and LGA weights are generated by an auxiliary "guidance subnet" fed with the input reference image and its extracted features.

2.1 Semi-Global Guided Aggregation (SGA)

Inspired by SGM, GANet introduces the SGA step which supports backpropagation. The SGA step that aggregates along a direction is

$$C^A_{\mathbf{r}}(\mathbf{p}, d) = C(\mathbf{p}, d) + \mathrm{sum}\begin{cases} \mathbf{w}_1(\mathbf{p}, \mathbf{r}) \cdot C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d), \\ \mathbf{w}_2(\mathbf{p}, \mathbf{r}) \cdot C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d - 1), \\ \mathbf{w}_3(\mathbf{p}, \mathbf{r}) \cdot C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d + 1), \\ \mathbf{w}_4(\mathbf{p}, \mathbf{r}) \cdot \max_i C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, i). \end{cases} \tag{7}$$

and presents several differences with respect to (5).

The main difference with the SGM approach is that the weights are learnt and hence adaptive and more flexible compared to the user-defined parameters from (3). Other changes can be noted between (5) and (7): (a) the outer min is changed to a weighted sum making the step all convolutional, (b) noting that the learning target of GANet is to maximize the probabilities at the ground truth depths and not to directly minimize the matching costs, the authors also change the inner min to a max.

Considering that the sum on a path can lead to large values, the weights are normalized. In practice, (7) is finally implemented as

$$C^A_{\mathbf{r}}(\mathbf{p}, d) = \mathrm{sum}\begin{cases} \mathbf{w}_0(\mathbf{p}, \mathbf{r}) \cdot C(\mathbf{p}, d), \\ \mathbf{w}_1(\mathbf{p}, \mathbf{r}) \cdot C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d), \\ \mathbf{w}_2(\mathbf{p}, \mathbf{r}) \cdot C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d - 1), \\ \mathbf{w}_3(\mathbf{p}, \mathbf{r}) \cdot C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, d + 1), \\ \mathbf{w}_4(\mathbf{p}, \mathbf{r}) \cdot \max_i C^A_{\mathbf{r}}(\mathbf{p} - \mathbf{r}, i). \end{cases} \quad \text{s.t.} \quad \sum_{i=0,1,2,3,4} \mathbf{w}_i(\mathbf{p}, \mathbf{r}) = 1. \tag{8}$$

2.2 Network Architecture

Figure 4 shows the main blocks of the GANet architecture and Table 1 lists their layers and parameters for the "GANet-deep" model.

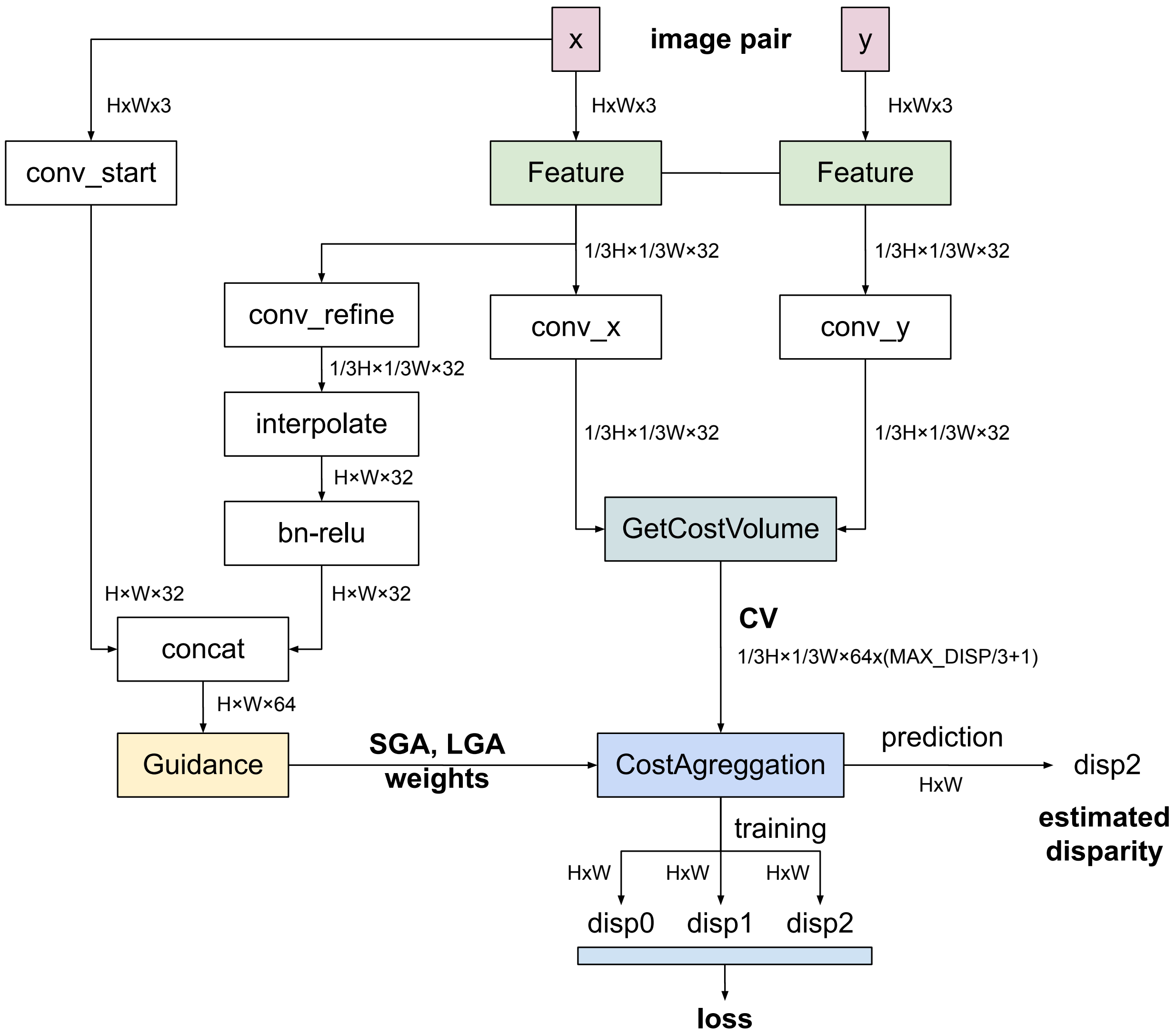


Figure 4: GANet architecture overview. The main blocks of the net are depicted in color.

2.3 Data

GANet developers present in [25] the evaluation on three datasets: SceneFlow [18], KITTI2012 and KITTI2015 [10, 19]. The SceneFlow dataset contains stereo frames rendered from various synthetic sequences. The KITTI datasets comprise images from urban and road scenes taken from the viewpoint of a car. In all the cases they are close range images where the camera, real or virtual, is close to the scene. The main characteristics of the images of these datasets are shown in Tables 2 and 3.

Layer id	Inputs	Layer description	Output tensor	Output
Feature extraction				
input		image	H×W×3	
1	image	conv	H×W×32	
2	1	conv	1/3H×1/3W×32	
3	2	conv	1/3H×1/3W×32	
4	3	conv	1/6H×1/6W×48	
5	4	conv	1/12H×1/12W×64	
6	5	conv	1/24H×1/24W×96	
7	6	conv	1/48H×1/48W×128	
8	7,6	deconv / concat / conv	1/24H×1/24W×96	
9	8,5	deconv / concat / conv	1/12H×1/12W×64	
10	9,4	deconv / concat / conv	1/6H×1/6W×48	
11	10,3	deconv / concat / conv	1/3H×1/3W×32	
12	11,10	deconv / concat / conv	1/6H×1/6W×48	
13	12,9	deconv / concat / conv	1/12H×1/12W×64	
14	13,8	deconv / concat / conv	1/24H×1/24W×96	
15	14,7	deconv / concat / conv	1/48H×1/48W×128	
16	15,14	deconv / concat / conv	1/24H×1/24W×96	
17	16,13	deconv / concat / conv	1/12H×1/12W×64	
18	17,12	deconv / concat / conv	1/6H×1/6W×48	
19	18,11	deconv / concat / conv	1/3H×1/3W×32	feature
Guidance branch				
input		concat 1 and up-sampled feature as input	H×W×64	
(1)		3×3 conv	H×W×16	
(2)		5×5 conv, stride 3	1/3H×1/3W×32	
(3)		3×3 conv	1/3H×1/3W×32	
(4)		3×3 conv (no bn & relu)	1/3H×1/3W×640	
(5)		split, reshape, normalize	4×1/3H×1/3W×5×32	sg1
(6)		from (3), 3×3 conv	1/3H×1/3W×32	
(7)		3×3 conv (no bn & relu)	1/3H×1/3W×640	
(8)		split, reshape, normalize	4×1/3H×1/3W×5×32	sg2
(9)-(11)	(6)	from (6), repeat (6)-(8)	4×1/3H×1/3W×5×32	sg3
(12)	(9)	from (9), 3x3 conv, stride 2	1/6H×1/6W×48	
(13)		3x3 conv	1/6H×1/6W×48	
(14)		3×3 conv (no bn & relu)	1/6H×1/6W×960	
(15)		split, reshape, normalize	4×1/3H×1/3W×5×48	sg11
(16)	(13)	from (13), 3x3 conv	1/6H×1/6W×48	
(17)		3×3 conv (no bn & relu)	1/6H×1/6W×960	
(18)		split, reshape, normalize	4×1/6H×1/6W×5×48	sg12
(19)-(21)	(16)	from (16), repeat (16)-(18)	4×1/6H×1/6W×5×48	sg13
(22)-(24)	(19)	from (19), repeat (19)-(21)	4×1/6H×1/6W×5×48	sg14
(25)	(1)	from (1), 3×3 conv	H×W×16	
(26)		3×3 conv (no bn & relu)	H×W×75	lg1
(27)-(28)		repeat (25)-(26)	H×W×75	lg2
Cost aggregation				
input		4D cost volume	1/3H×1/3W×64x(MAX_DISP/3+1)	
[1]	CV	3×3×3, 3D conv	1/3H×1/3W×32x(MAX_DISP/3+1)	
[2]	[1]	SGA layer: weight matrices from (5)	1/3H×1/3W×32x(MAX_DISP/3+1)	
		3×3×3, 3D to 2D conv, upsampling	H×W×(MAX_DISP+1)	
output		softmax, regression	H×W×1	disp0 (for training loss)
[3]	[2]	3×3×3, 3D conv, stride 2	1/6H×1/6W×48x(MAX_DISP/6+1)	
[4]	[3]	SGA layer: weight matrices from (15)	1/6H×1/6W×48x(MAX_DISP/6+1)	
[5]	[4]	3×3×3, 3D conv, stride 2	1/12H×1/12W×64x(MAX_DISP/12+1)	
[6]	[5],[4]	3×3×3, 3D deconv, stride 2	1/6H×1/6W×48x(MAX_DISP/6+1)	
[7]	[6]	SGA layer: weight matrices from (18)	1/6H×1/6W×48x(MAX_DISP/6+1)	
[8]	[7],[2]	3×3×3, 3D deconv, stride 2	1/3H×1/3W×32x(MAX_DISP/3+1)	
[9]	[2]	SGA layer: weight matrices from (8)	1/3H×1/3W×32x(MAX_DISP/3+1)	
		3×3×3, 3D to 2D conv, upsampling	H×W×(MAX_DISP+1)	
output		softmax, regression	H×W×1	disp1 (for training loss)
[10]	[9]	3×3×3, 3D conv, stride 2	1/6H×1/6W×48x(MAX_DISP/6+1)	
[11]	[10]	SGA layer: weight matrices from (21)	1/6H×1/6W×48x(MAX_DISP/6+1)	
[12]	[11]	3×3×3, 3D conv, stride 2	1/12H×1/12W×64x(MAX_DISP/12+1)	
[13]	[12],[11]	3×3×3, 3D deconv, stride 2	1/6H×1/6W×48x(MAX_DISP/6+1)	
[14]	[13]	SGA layer: weight matrices from (24)	1/6H×1/6W×48x(MAX_DISP/6+1)	
[15]	[13],[9]	3×3×3, 3D deconv, stride 2	1/3H×1/3W×32x(MAX_DISP/3+1)	
[16]	[15]	SGA layer: weight matrices from (11)	1/3H×1/3W×32x(MAX_DISP/3+1)	
[17]	[16]	3×3×3, 3D to 2D conv, upsampling	H×W×(MAX_DISP+1)	
[18]	[17]	LGA layer: weight matrices from (26)	H×W×(MAX_DISP+1)	
[19]	[18]	softmax	H×W×(MAX_DISP+1)	
[20]	[19]	LGA layer: weight matrices from (28)	H×W×(MAX_DISP+1)	
output	[20]	normalization, regression	H×W×1	disp2 (estimated disparity)

Table 1: Network layers of the main blocks of the "GANet Deep" model

	Input stereo pair	Target disparity
Product name	"RGB images (finalpass)"	"Disparity"
File format	PNG	PFM
Channels	3 (RGB)	1
Pixel depth (type)	8 bits (unsigned byte)	32 bits (floating point)
Image size	960x540	960x540

Table 2: SceneFlow data characteristics

	Input stereo pair	Target disparity
Channels	3 (RGB)	1
Pixel depth (type)	8 bits (unsigned byte)	32 bits (floating point)
Image size	1240x376	1240x376

Table 3: KITTI2012 and KITTI2015 data characteristics

2.4 Training

Table 4 presents the training parameters on the SceneFlow, KITTI2012 and KITTI2015 datasets. The authors of the method have disclosed "GANet-deep" models trained on these datasets on their Github page[3].

Dataset	**SceneFlow**	**KITTI2012 / KITTI2015**
Training set size (stereo pairs)	35454	194 / 199
Hardware	8 GPUs (*)	8 GPUs (*)
Batch size	16 (**)	16 (**)
Image size (W x H)	576x240 random crops	576x240 random crops
Image preprocessing	Per channel image normalization (***)	Per channel image normalization (***)
Initial weights	Random	From training on SceneFlow
Optimizer	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)	Adam ($\beta_1 = 0.9$, $\beta_2 = 0.999$)
Learning rate	0.001	0.001 (first 300 ep.), 0.0001 (remaining ep.)
Epochs	10	640
	(*) P40 - 22GB (**) 8 for the disclosed pretrained models (***) subtract mean divide by std	(*) P40 - 22GB (**) 8 for the disclosed pretrained models (***) substract mean divide by std

Table 4: Training parameters on SceneFlow, KITTI2012 and KITTI2015

3 Results

The GANet method has achieved very good results on the KITTI2012 and KITTI2015 [10, 19]. The original model and other more recent variants based on GANet are placed high on the rankings of these benchmarks[4].

In the KITTI benchmarks, specific training on the concrete datasets was performed. But GANet also exhibits great generalization abilities and can perform well on other datasets without a specific training or fine tuning. Some result examples are presented by the authors of GANet on the Cityscapes [5] and the Middlebury [22] datasets on their Github page[5]. Figure 6 shows the result on one of the images of the Middlebury dataset computed with the demo associated to this article (see Section 4) that uses a model trained on SceneFlow.

The generalization ability of the method was also pointed out in [11] where a model trained on SceneFlow (comprised of close range images) was used on satellite images with encouraging results. Despite the current popularity of deep learning stereo matching methods, they are still not the

[3] https://github.com/feihuzhang/GANet [Accessed on June 2022].
[4] http://www.cvlibs.net/datasets/kitti/eval_stereo.php [Accessed on June 2022].
[5] https://github.com/feihuzhang/GANet [Accessed on June 2022].

preferred matching option in satellite stereo pipelines [6, 3, 21, 17]. Satellite images have specific characteristics that hinder the adaptation of well established methods used on close range images: a) the extremely small ratio between the depth range and the distance from the camera to the scene implies working with a camera model that deviates from the standard pinhole and deals with structures that occupy few pixels in the images; (b) the images for a certain location can only be acquired through several sweeps which may be days, weeks or even months apart, introducing variability in illumination, seasonal changes and man-made changes, among others. The variability poses important challenges for the matching of correspondent regions across the images. Despite the differences between the train and test sets, [11] shows that reconstruction results with GANet, used as the matching step in the S2P [6] satellite pipeline, were comparable to the results with the classic matching counterpart [9] currently in use in the pipeline. It is interesting to note that part of the internal structure of GANet mimics SGM [14] which has been extensively used as the main aggregation strategy in classic matching methods of satellite stereo pipelines.

4 Demo

The IPOL demo related to this article can be accessed at the web page of this article[6].

The demo uses the "GANet-deep" model trained on SceneFlow mentioned in Section 2.4.

To run the demo the users must first select a pair of images from the gallery, or upload their own images. The gallery (see Figure 5) has also the ground truth for the disparity, which can be compared with the result of an execution. In the case of uploaded images the ground truth is optional.

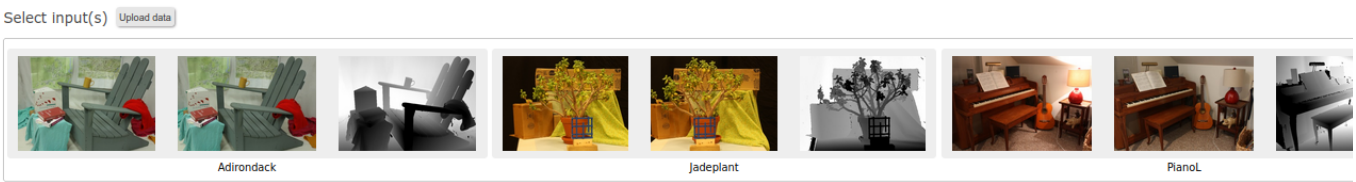


Figure 5: Gallery of available image pairs. The demo also allows to upload images.

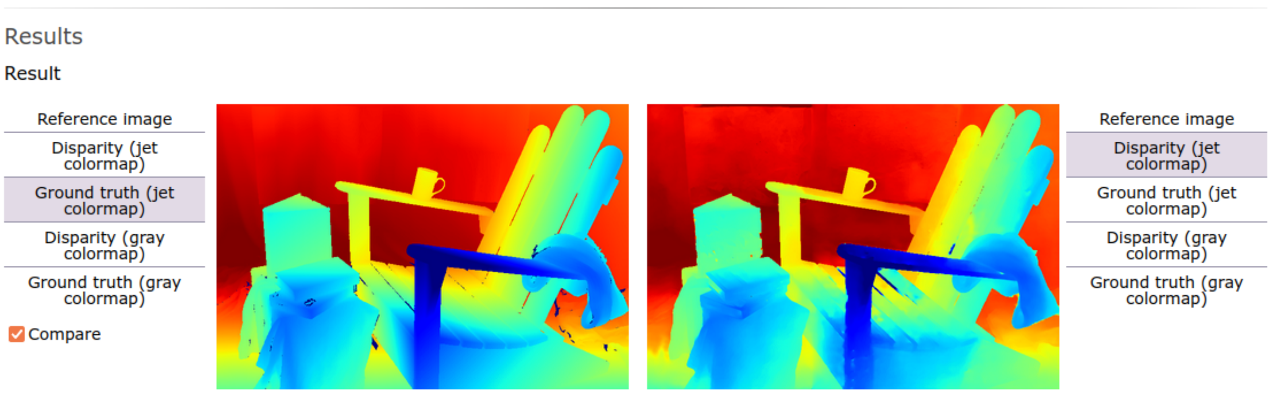


Figure 6: Results section and a side-to-side comparison of the computed disparity and the ground truth.

Once the input images are selected, they can be inspected. Next, the parameter must be selected and the Run button must be pressed. The max_disp parameter controls the number of disparity

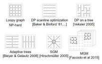
[6] https://doi.org/10.5201/ipol.2023.441

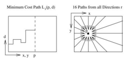
steps considered in the reconstruction. Smaller values of this parameter result in shorter running time but coarser results.

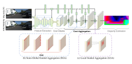
When the execution is finished, the computed disparity map can be inspected in the Results section by alternating the images (hovering over the buttons) or by a side-to-side comparison as shown in Figure 6.

Image Credits

All images by the author except:

 Reproduced from [8]

 Reproduced from [13]

 Reproduced from [25]

 Middlebury dataset [23] [7]

References

[1] A. Atapour-Abarghouei and T.P. Breckon, *A comparative review of plausible hole filling strategies in the context of scene depth image completion*, Computers & Graphics, 72 (2018), pp. 39–58. https://doi.org/10.1016/j.cag.2018.02.001.

[2] H.H. Baker and T.O. Binford, *Depth from edge and intensity based stereo*, in International Joint Conference on Artificial Intelligence (IJCAI) - Volume 2, San Francisco, CA, USA, 1981, Morgan Kaufmann Publishers Inc., p. 631–636.

[3] R.A. Beyer, O. Alexandrov, and S. McMichael, *The Ames Stereo Pipeline: NASA's Open Source Software for Deriving and Processing Terrain Data*, Earth and Space Science, 5 (2018), pp. 537–548. http://dx.doi.org/https://doi.org/10.1029/2018EA000409.

[4] M. Bleyer and M. Gelautz, *Simple but effective tree structures for dynamic programming-based stereo matching*, in International Conference on Computer Vision Theory and Applications (VISAPP), vol. 2, 2008, pp. 415–422. https://doi.org/20.500.12708/52347.

[5] 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*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3213–3223. http://dx.doi.org/10.1109/CVPR.2016.350.

[6] C. de Franchis, E. Meinhardt-Llopis, J. Michel, J-M. Morel, and G. Facciolo, *An automatic and modular stereo pipeline for pushbroom images*, ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences, II-3 (2014), pp. 49–56. http://dx.doi.org/10.5194/isprsannals-II-3-49-2014.

[7] A. Drory, C. Haubold, S. Avidan, and F.A. Hamprecht, *Semi-global matching: A principled derivation in terms of message passing*, in Pattern Recognition, Springer International Publishing, Cham, 2014, pp. 43–53. http://dx.doi.org/10.1007/978-3-319-11752-2_4.

[7] https://vision.middlebury.edu/stereo/

[8] G. Facciolo, *Stereovision for satellite images. Lecture notes of the course: Remote sensing data: from sensor to large-scale geospatial data exploitation*, February 2018.

[9] G. Facciolo, C. de Franchis, and E. Meinhardt, *MGM: A significantly more global matching for stereovision*, in British Machine Vision Conference, British Machine Vision Association, 2015, pp. 90.1–90.12. http://dx.doi.org/10.5244/C.29.90.

[10] A. Geiger, P. Lenz, and R. Urtasun, *Are we ready for autonomous driving? The KITTI vision benchmark suite*, in IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354–3361. http://dx.doi.org/10.1109/CVPR.2012.6248074.

[11] A. Gómez, G. Randall, G. Facciolo, and R. Grompone von Gioi, *An experimental comparison of multi-view stereo approaches on satellite images*, in IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2022, pp. 707–716. http://dx.doi.org/10.1109/WACV51458.2022.00078.

[12] R.A. Hamzah and H. Ibrahim, *Literature survey on stereo vision disparity map algorithms*, Journal of Sensors, (2016). https://doi.org/10.1155/2016/8742920.

[13] H. Hirschmüller, *Stereo Vision in Structured Environments by Consistent Semi-Global Matching*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2006, pp. 2386–2393. http://dx.doi.org/10.1109/CVPR.2006.294.

[14] H. Hirschmuller, *Stereo processing by semiglobal matching and mutual information*, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 30 (2008), pp. 328–341. http://dx.doi.org/10.1109/TPAMI.2007.1166.

[15] H. Ishikawa, *Exact optimization for Markov Random Fields with convex priors*, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 25 (2003), pp. 1333–1336. http://dx.doi.org/10.1109/TPAMI.2003.1233908.

[16] H. Laga, L.V. Jospin, F. Boussaid, and M. Bennamoun, *A survey on deep learning techniques for stereo-based depth estimation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, (2020). https://doi.org/10.1109/TPAMI.2020.3032602.

[17] M.J. Leotta, C. Long, B. Jacquet, M. Zins, D. Lipsa, J. Shan, B. Xu, Z. Li, X. Zhang, S-F. Chang, M. Purri, J. Xue, and K. Dana, *Urban Semantic 3D Reconstruction From Multiview Satellite Imagery*, in IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2019, pp. 1451–1460. http://dx.doi.org/10.1109/CVPRW.2019.00186.

[18] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox, *A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 4040–4048. http://dx.doi.org/10.1109/CVPR.2016.438.

[19] M. Menze, C. Heipke, and A. Geiger, *Joint 3D Estimation of Vehicles and Scene Flow*, in ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, vol. II-3/W5, 2015, pp. 427–434. http://dx.doi.org/10.5194/isprsannals-II-3-W5-427-2015.

[20] S. Roy and I.J. Cox, *A Maximum-Flow Formulation of the N-Camera Stereo Correspondence Problem*, in International Conference on Computer Vision (ICCV), vol. 5, 1998, p. 492. http://dx.doi.org/10.1109/ICCV.1998.710763.

[21] E. Rupnik, M. Daakir, and M.P. Deseilligny, *Micmac–a free, open-source solution for photogrammetry*, Open Geospatial Data, Software and Standards, 2 (2017), pp. 1–9. http://dx.doi.org/10.1186/s40965-017-0027-2.

[22] D. Scharstein, H. Hirschmüller, Y. Kitajima, K. Krathwohl, N. Nešić, X. Wang, and P. Westling, *High-resolution stereo datasets with subpixel-accurate ground truth*, in Pattern Recognition, Springer, 2014, pp. 31–42. http://dx.doi.org/10.1007/978-3-319-11752-2_3.

[23] D. Scharstein and R. Szeliski, *A taxonomy and evaluation of dense two-frame stereo correspondence algorithms*, International Journal of Computer Vision, 47 (2002), pp. 7–42. https://doi.org/10.1109/SMBV.2001.988771.

[24] O. Veksler, *Stereo Correspondence by Dynamic Programming on a Tree*, in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, 2005, pp. 384–390. http://dx.doi.org/10.1109/CVPR.2005.334.

[25] F. Zhang, V. Prisacariu, R. Yang, and P.H.S. Torr, *GA-Net: Guided Aggregation Net for End-To-End Stereo Matching*, in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, June 2019, IEEE, pp. 185–194. http://dx.doi.org/10.1109/CVPR.2019.00027.