LiteMVSNet: A Lightweight Multi-View Stereo Network with Cascade Depth Searching Scope and Recurrent Pseudo-3D CNNs

Li Yan, Dengsu Zhang, Hong Xie, Jin Shan
SGG4DRecon Group

Version: 0.01

Last update: April 12, 2019

Abstract

Various neural networks have shown that depth estimation using multi-view stereopsis can be well simulated by deep learning. The competition between running time and memory cost steps on stage. R-MVSNet, one of the state-of-the-art project, provides a gated recurrent unit (GRU) architecture to decompose 3D CNNs to 2D RNN for cost regularization, at the expense of inherent parallelism. The conflict between space and time for matching cost regularization is alliviated by our method: (1)We present an alternative network named LiteMVSNet that ourperforms R-MVSNet on the ETH3D benchmark, and requires less memory space. (2)We present a more effective depth regularization architecture with cascade depth searching scope. (3)Our networks own an efficient structure for feature extraction and cost regularization with pseudo-2D/3D CNNs. The proposed approach is compared with classic and learning-based algorithms.

1 Introduction

Multi-view Stereo (MVS) methods estimates a 3D model from a collection of overlapping 2D images with known camera intrinsic and extrinsic parameters, which can be estimated by structure from motion (SFM) pipline. Stereo matching algorithms universally takes four steps: cost computation, cost regularization, depth computation and depth refinement [21]. The matching cost can be calculated by different similarity measures[2][29], witch are always called features in machine learning. Traditional ways have tried every effort to illustrate implicit features in MVS pipline, such as texture[12][13], geometric structures[19][20][17], imaging characteristics[15][31], and sematic cues[26]. However, it seams inexhaustible to make explicit functional expressions of all these cues.

Fortunately, the CNN structure provides an elegant way to utilize these featues which are significant for 3D reconstruction. Learning-based MVS methods recently achieve state-of-the-art performance on 3D reconstruction benchmarks[9][11][22].

Nevertheless, many existing learning-based MVS[6][27][16] requires a large memory space for feature extraction and cost regularization. All of these projects use space-sweeping method[3]. The problem space grows cubically with the model resolution. Some methods are proposed for this defect: OctNet[18] and AO-CNN[25] apply the octree structure on 3D CNNs, as 3D data is usually sparse. R-MVSNet[28] introduces GRU to decompose 3D CNNs into recurrent 2D CNNs, so that R-MVSNet is capable for high resolution 3D reconstruction with unlimited depth-wise resolution. At the cost of the serial structure, R-MVSNet's running time grows linearly with the size of depth dimension. It is the trade-off between space and time that this paper concerns.

In this paper, we presents a lightweight multi-view stereo framework, named as LiteMVSNet. The framework is built upon MVSNet [27] and DeMoN[24] architectures, and introduces the cascade depth searching scope (CDSS) and recurrent pseudo-2D/3D CNNs (RP-CNNs) for depth inference. The depth searching scope is gradually shrinked to the truth, which helps to avoid unnecessary time cost and improves the efficiency of execution. The RP-CNNs are helpful for both decreasing the width of the model and increasing it's depth.

DTU[9], Tanks and Temples[11] and ETH3D datasets[22] are used for evaluation. We compare our method with the state-of-the-art projects: (COLMAP, DeepMVS, MVSNet, R-MVSNet, and DeepTAM). Our method has the ability for high resolution reconstruction, and is time saving for higher depth-wise resolution.

2 Related Work

FlowNet[4] and MatchNet[1], the pioneers to learn pair-wise matching, expand a brand new area in deep learning. Before long, FlowNet2[8], DispNet[14], and SGMNet[14] achieve to exceed non-learning method in early vision tasks (such as flow estimation, disparity computation and patch matching). DeMoN[24], mainly based on the idea of FlowNet2, clearly demonstats how to establish the correspondence for two-view geometry. However, expanding two-view to multi-view for learning-based method is a demanding work. It is not only a challenge in function modeling but also a dramatical increase in computing requirements.

Space-sweeping[3] (or plane-sweeping[5]) is the mainly technique employed in current MVS networks. There're three steps in space-sweeping: cost volume computation, cost regularization, and depth inference.

The mathing cost is defined by the similarity metrics. Almost all the learning-based architectures use 2D CNNs to extract features for similarity computation, but some, like DeepTAM[30], use traditional metrics for efficiency and astringency. SurfaceNet[10] introduces colored voxel cube (CVC) for cost computation with multi view stereopsis. However, it accumulates the cost in object space, which makes the method incapable for large scale scenes. MVSNet introduces a differentiable homography warping, so that cost volume can be computed in image space. In this way, the space can be only sweeped with planes. A more generalized nonlinear warping using flow can be found in the series of LMB's works[7][30]. Flow-warping is the key to our CDSS, which makes it possible to arbitrarily sweep the space.

There are mainly two ways to regularize the cost volume and infere the depth map: 2D CNNs and 3D

CNNs. Matching costs, which are calculated at different depth of a same pixel, can be feed into 2D CNNs' feature channle. This trick is used by DeMoN, DeepMVS and DeepTAM. In addition, some projects, like MVSNet and R-MVSNet, use 3D CNNs, which give the cost volume a more distinct geometry interpretation. The method with 2D CNNs discards the inherent 3D data structure, and forces the model to learn it. Despite the model's ability to express the geometry, additional memory costs are caused. Unlike 2D CNNs, 3D CNNs only make local links in depth dimension,

The skeleton structure of learned MVS has been developed by previous works. However, most of them ask for a large quantity of GPU memory. DeepMVS, which uses extremely high-dimensional features (VGGNet19[23]), can only handle 128px × 128px patches (the center of 64px × 64px used for tiling) with 4GB memory at once. The whole input image is splitted into patches, and then outputs are tiled to achieve full-resolution result. It takes around 4s for a patch on 1080Ti, and 4min for the whole image of 512px × 512px, which is unacceptable for practical application. Same shortcomings exist in other 2D-CNNs-based structures. 3D CNNs are much more time-saving but also memory-expensive. R-MVSNet introduces the GRU structure to sequentially regularizes the 2D cost maps along the depth direction. By this way, R-MVSNet reduces the memory consumption at the expense of running time. However, the GRU doesn't destroy the local link structure in depth dimension, so it is still more efficient both in space and time than 2D CNNs.

3 LiteMVSNet

In this section, we will describe the

3.1 Network Outline

3.2 Features

3.3 Cost Volume

3.4 Depth Map

3.5 Loss

4 Experiment

5 Related Work

5.1 Font Settings

I change the default article font computer modern to `newtx` series, and the default font size is set to `11pt`.

- `newtxtext` package for text font, similar to times new roman font.
- `newtxmath` package for math font, close to `times` and `mtpro2` packages.

- `newtxtt` package for typewriter font, with option `scale = 0.8`.

These packages operate perfectly but are inappropriate for big operators, for example `\sum` and `\prod`, thus, I change these operators back to computer modern font. Equation (1) shows the effects of these fonts:

$$(a+b)^n = \sum_{k=0}^{n} C_n^k a^{n-k} b^k \tag{1}$$

The `\linespread` (controls line spacing) is set to 1.3, and I use `microtype` to improve the font justification. `type1cm` package is used to remove the font shape and font size warning messages.

5.2 Custom Commands

I don't change any default command or environment, which means you can use all the basic LaTeX commands and environments as before. Besides, I define 3 commands

1. `\email{#1}`: create the hyperlink to email address.
2. `\figref{#1}`: same usage as `\ref{#1}`, but start with label text <**Figure n**>.
3. `\tabref{#1}`: same usage as `\ref{#1}`, but start with label text <**Table n**>.

5.3 List Environments

When you are using `itemize`, `enumerate`, or `description` environment, please add the `noitemsep` option to these environments. For example,

```
\begin{itemize}[noitemsep]
    \item Routing and resource discovery;
    \item Resilient and scalable networks;
    \item Distributed storage and search.
\end{itemize}
```

- Routing and resource discovery;
- Resilient and scalable computer networks;
- Distributed storage and search.

5.4 Table

I strongly recommend you to use the `booktabs` package in your paper. It adds three commands to make the table prettier, ie. `\toprule`, `\midrule` and `\bottomrule`. Here is an example.

5.5 Graphics

To include a graphic, you can use figure environment as usual. **Figure** 1 shows the effect. You can put all your images in the sub directories (`./image/`, `./img/`, `./figure/`, `./fig/`) of your current working directory.

```
\begin{figure}[!ht]
    \centering
    \includegraphics[width=0.6\textwidth]{mpg.png}
    \caption{The Relationship between MPG and Weight\label{fig:mpg}}
\end{figure}
```

Table 1: Regression Result Example

	(1)	(2)
	price	price
mpg	-238.9***	-49.51
	(53.08)	(86.16)
weight		1.747***
		(0.641)
constant	11,253***	1,946
	(1,171)	(3,597)
observations	74	74
R-squared	0.220	0.293

Standard errors in parentheses

*** p<0.01, ** p<0.05, * p<0.1

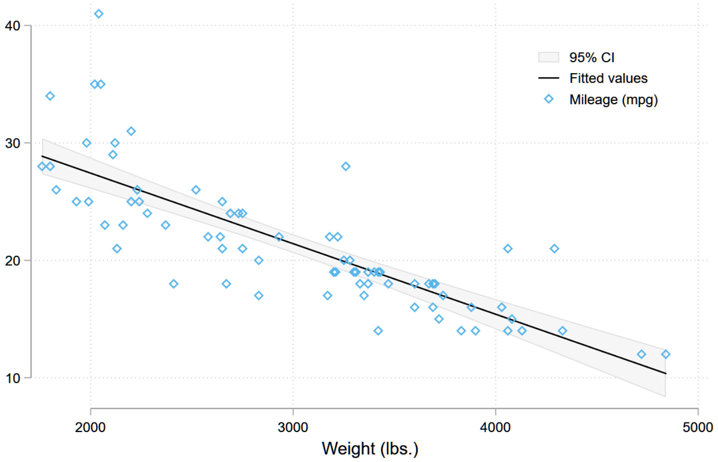

Figure 1: The Relationship between MPG and Weight

5.6 Bibliography

This template uses BibTeX to generate the bibliography, the default bibliography style is `aer`. [?] use data from a major peer-to-peer lending marketplace in China to study whether female and male investors evaluate loan performance differently. You can add bib items (from Google Scholar, Mendeley, EndNote, and etc.) to `wp_ref.bib` file, and cite the bibkey in the `tex` file.

References

[1] and T. Leung, Y. Jia, R. Sukthankar, and A. C. Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3279–3286, June 2015.

[2] Sylvie Chambon and Alain Crouzil. Similarity measures for image matching despite occlusions in stereo vision. *Pattern Recognition*, 44:2063–2075, 09 2011.

[3] R. T. Collins. A space-sweep approach to true multi-image matching. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 358–363, June 1996.

[4] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazirbas, V. Golkov, P. v. d. Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2758–2766, Dec 2015.

[5] D. Gallup, J. Frahm, P. Mordohai, Q. Yang, and M. Pollefeys. Real-time plane-sweeping stereo with multiple sweeping directions. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2007.

[6] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. *CoRR*, abs/1804.00650, 2018.

[7] E. Ilg, T. Saikia, M. Keuper, and T. Brox. Occlusions, motion and depth boundaries with a generic network for disparity, optical flow or scene flow estimation. In *European Conference on Computer Vision (ECCV)*, 2018.

[8] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. *CoRR*, abs/1612.01925, 2016.

[9] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413. IEEE, 2014.

[10] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. *CoRR*, abs/1708.01749, 2017.

[11] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4), 2017.

[12] Fabian Langguth, Kalyan Sunkavalli, Sunil Hadap, and Michael Goesele. Shading-Aware Multi-view Stereo. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, volume 9907, pages 469–485. Springer International Publishing, Cham, 2016.

[13] Z. Li, K. Wang, W. Zuo, D. Meng, and L. Zhang. Detail-preserving and content-aware variational multi-view stereo reconstruction. *IEEE Transactions on Image Processing*, 25(2):864–877, Feb 2016.

[14] 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 International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. arXiv:1512.02134.

[15] Pratul P. Srinivasan, Rahul Garg, Neal Wadhwa, Ren Ng, and Jonathan T. Barron. Aperture supervision for monocular depth estimation. 11 2017.

[16] Despoina Paschalidou, Ali Osman Ulusoy, Carolin Schmitt, Luc J. Van Gool, and Andreas Geiger. Raynet: Learning volumetric 3d reconstruction with ray potentials. *CoRR*, abs/1901.01535, 2019.

[17] C. Rhemann, A. Hosni, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 3017–3024, Washington, DC, USA, 2011. IEEE Computer Society.

[18] Gernot Riegler, Ali Osman Ulusoy, and Andreas Geiger. Octnet: Learning deep 3d representations at high resolutions. *CoRR*, abs/1611.05009, 2016.

[19] Andrea Romanoni and Matteo Matteucci. Tapa-mvs: Textureless-aware patchmatch multi-view stereo. *CoRR*, abs/1903.10929, 2019.

[20] Shuji SAKAI, Koichi ITO, Takafumi AOKI, Takafumi WATANABE, and Hiroki UNTEN. Phase-based window matching with geometric correction for multi-view stereo. *IEICE Transactions on Information and Systems*, E98.D:1818–1828, 10 2015.

[21] D. Scharstein, R. Szeliski, and R. Zabih. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. In *Proceedings IEEE Workshop on Stereo and Multi-Baseline Vision (SMBV 2001)*, pages 131–140, Dec 2001.

[22] Thomas Schöps, Johannes L. Schönberger, Silvano Galliani, Torsten Sattler, Konrad Schindler, Marc Pollefeys, and Andreas

Geiger. A multi-view stereo benchmark with high-resolution images and multi-camera videos. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[24] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. *CoRR*, abs/1612.02401, 2016.

[25] Peng-Shuai Wang, Chun-Yu Sun, Yang Liu, and Xin Tong. Adaptive O-CNN: A Patch-based Deep Representation of 3D Shapes. *ACM Transactions on Graphics (SIGGRAPH Asia)*, 37(6), 2018.

[26] Shibiao Xu, Feihu Zhang, Xiaofei He, Xukun Shen, and Xiaopeng Zhang. Pm-pm: Patchmatch with potts model for object segmentation and stereo matching. *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 24, 03 2015.

[27] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. *CoRR*, abs/1804.02505, 2018.

[28] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. *CoRR*, abs/1902.10556, 2019.

[29] Nelson Yen-Chung Chang, Yu-Cheng Tseng, and Tian Sheuan Chang. Analysis of color space and similarity measure impact on stereo block matching. 11 2008.

[30] H. Zhou, B. Ummenhofer, and T. Brox. Deeptam: Deep tracking and mapping. In *European Conference on Computer Vision (ECCV)*, 2018.

[31] Shaojie Zhuo and Terence Sim. Defocus map estimation from a single image. *Pattern Recogn.*, 44(9):1852–1858, September 2011.