

本科毕业设计论文

外文翻译

题目：基于卷积神经网络的双目视觉立体匹配

**作者姓名 王 灏**

**指导教师 宣琦副教授**

**专业班级 通信工程1301**

**学 院 信息工程学院**

**提交日期** 2017年2月28

通过训练卷积神经网络比较图像块的立体匹配

Jure Zbontar \*：jure.zbontar@fri.uni-lj.si

计算机与信息科学学院

卢布尔雅那大学

Vecna pot 113，SI-1001 Ljubljana，Slovenia

Yann LeCun：  yann@cs.nyu.edu

Courant数学科学研究所

纽约大学

715 Broadway，New York，NY 10003，USA

摘要:我们提出了一种从经过矫正的图像对中提取深度信息的方法。我们的方法集中在许多立体视觉算法的第一阶段：匹配代价计算。我们通过使用卷积神经网络进行小图像块的相似性度量来解决这个问题。 我们使用相似和不相似对的图像对构建二进制分类数据集来进行有监督的训练。

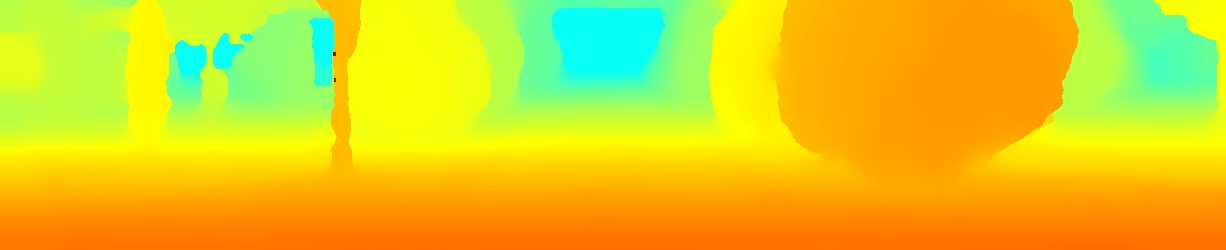
针对这个任务，我们检测两个网络架构：一个是速度调整， 另一个是准确性调整。 ？？

卷积神经网络的输出用于初始化立体匹配代价。一系列后处理步骤如下：基于交叉的代价聚合，半全局匹配，左右一致性检查，亚像素增强，一个中值滤波器和一个双边滤波器。我们用KITTI 2012，KITTI 2015和 Middlebury的立体视觉数据集来评估我们的方法，并发现它在所有的三大数据集中的表现均优于其他方法。

关键词：立体视觉，匹配代价，相似性学习，监督学习，卷积神经网络

1．介绍

考虑以下问题：给定由不同水平位置的相机拍摄的两个图像，我们希望计算左图像中每个像素的视差d。 视差指的是同一对象在左图像和右图像中的水平位置的差异——同一对象在左图像中的位置为（x，y），在右图像中的位置为（x-d，y）。 如果我们知道一个对象的视差，我们可以使用以下关系计算它的深度z：



左侧输入图像

右侧输入图像

输出视差图



1.7

m

90

m

20

m

**图1:** 输入是来自左侧和右侧相机的一对图像。 两个输入图像的差异主要在对象的水平位置上（其他的差异是由反射，遮挡和透视失真引起的）。注意靠近相机的对象比远离相机的对象有更大的视差。右图的输出是一个密集的视差图，暖色表示更大的视差值（和较小的深度值）

其中f是相机的焦距，B是相机中心之间的距离。图 1描述了输入和通过我们的方法处理的输出。

上述的立体匹配问题在许多领域中都是非常重要的，例如自动驾驶，机器人技术，中间视图生成和3D场景重建。 根据Scharstein和Szeliski （2002）的分类，一个典型的视觉算法包括四个步骤：匹配代价计算，代价聚合，优化，视差精化。 根据 Hirschmüller和Scharstein （2009）的分类，我们认为前两个步骤是计算匹配代价和后两个步骤为立体视觉算法。我们工作的重点是计算好匹配代价。

我们建议对图像块对训练卷积神经网络（ LeCun等人，1998年），其中真实的视差是已知的（例如，由雷达或结构光获得）。网络的输出用于初始化匹配代价。我们还需进行许多不是新颖的但是必要的后处理步骤以取得良好效果。匹配代价是由具有相似图像强度的邻近像素通过基于交叉的代价聚合的方式组合而成。平滑约束通过半全局匹配进行，同时左右一致性检查被用来检测和消除遮挡区域中的误差。我们进行亚像素增强并应用中值滤波器和双边滤波器以获得最终视差图。

本文的贡献是：

* 基于卷积神经网络的为计算立体匹配代价的两种架构的描述;
* 一种方法，伴随其源代码，在KITTI 2012，KITTI 2015和Middlebury立体视觉数据集中具有最低的错误率;
* 实验分析了数据集大小的重要性，与其他方法相比的错误率，以及不同超参数设置下精度和运行时间之间的权衡。

本文延伸了我们以前的工作（Zbontar和LeCun，2015年），包括一个新架构的描述，两个新数据集的结果，更低的错误率以及更彻底的实验。

2．相关工作

在引入大型立体数据集如KITTI和Middlebury之前，相对较少立体视觉算法使用地面实况信息来得到他们模型的参数;在这一节，我们回顾一下一些以前的做法。有关立体视觉算法的一般概述，请参阅 Scharstein和Szeliski（2002）。

Kong 和Tao (2004) 采用平方距离的总和来计算初始匹配代价。然后他们训练了一个模型来预测三个类别的概率分布：初始视差正确的，由于前景目标过大导致初始视差不正确的，并且由于其他原因导致初始视差不正确的。预测概率被用来调整初始匹配代价。Kong和Tao(2006)随后延伸了他们的工作，通过组合由计算归一化的不同的窗口大小和中心的互相关获得的预测。Peris等人（2012）用AD- Census（Mei等，2011）进行初始化匹配代价，并使用多类线性判别分析来得知从计算的匹配代价到最终视差的映射。

地面实况数据还用于得到概率图形模型的参数。Zhang和 Seitz（2007）使用一种替代优化算法来估算马尔科夫随机场超参数的最优值。Scharstein和Pal（2007）构建了一个新的30个立体对的数据集，并使用它来得到条件随机场的参数。Li 和Huttenlocher（2008）提出了一个带非参数代价函数的条件随机场模型，并使用结构化支持向量机来得到模型参数。

最近的工作（Haeusler等人，2013; Spyropoulos等人，2014）集中在估计计算的匹配代价的置信度。Haeusler等人（2013）使用了一种随机森林分类器来组合若干置信度度量方式。同样的，Spyropoulos等人（2014）训练了一个随机森林分类器来预测匹配代价的置信度，并且使用预测结果作为马尔科夫随机场中的软约束来减少立体算法的误差。

一个计算匹配代价的相关问题是得到局部图像描述符（Brown等人，2011; Trzcinski等人，2012; Simonyan等人，2014; Revaud等人，2015; Paulin等人，2015；Han等人，2015; Zagoruyko和Komodakis，2015）。这两个问题共享一个公共子任务：测量图像块之间的相似性。Brown等人（2011）提出了一个总体框架来得到图像描述符并使用鲍威尔的方法来选择良好的超参数。为得到局部图像描述符已经提出了几种解决方法，例如Boosting优化（Trzcinski等人，2012），凸优化（Simonyan等人，2014年），分层移动象限的相似性（Revaud等人，2015），卷积内核网络（Paulin等人，2015），和卷积神经网络 （Zagoruyko和Komodakis，2015；Han等人，2015）。Zagoruyko和Komodakis（ 2015）的工作，Han等人（2015），特别是，非常类似于我们自己，不同的主要是网络的架构;具体地，包括合并和子采样以考虑更大的块尺寸和更大的视点变化。

3．匹配代价

典型的立体视觉算法首先计算在每个位置**p**考虑所有视差d的匹配代价。一种计算匹配代价的简单方法是绝对差的总和：

(1)

其中，和是位置**p**在左右图像中的图像强度，是以P为中心的固定的矩形窗口内的位置集合。

我们使用粗体小写字母**p**和**q**表示图像位置。粗体小写**d**表示向量的视差d，即**d**=（**d**，0）。我们为超参数的名称使用typewriter字体。例如，我们将使用patch size字体来表示附近区的大小

等式（1）可以理解为测量匹配左图以位置**p**为中心的图像块与右图以位置**p–d**为中心的图像块的代价。

我们希望当两个图像块围绕相同的3D点时代价小，反之则代价大。

既然好的和坏的匹配的示例可以从公开可用数据集构建（例如，KITTI和Middlebury立体数据集），我们可以尝试通过监督学习方法来解决匹配问题。受到卷积神经网络在视觉问题中成功应用的启发，我们用它来评估两个小图像块匹配程度。

3.1构造数据集

我们使用来自KITTI或Middlebury立体视觉数据集的地面实况视差图来构建二进制分类数据集。在每个真实视差已知的图像位置处，我们提取一个消极的和一个积极的训练示例。这确保了数据集包含相等数量的正例和负例。一个积极的示例是指一对图像块，其中一个来自左边图像，一个来自右边图像，两者的中心像素是同一个3D点，而一个消极的示例是指一对不同于前者的图像块。以下部分详细描述数据集构建步骤。

用表示一对图像块，其中为左图中以位置**p**=（x，y）为中心的nn图像块，是右图中以位置**q**为中心的nn图像块，d表示位置**p**处的正确视差。一个消极的示例的获取是通过将右图像块的中心设置为

其中是从区间[dataset\_neg\_low, dataset\_neg\_high]或它的原点对应区间[−dataset\_neg\_high, −dataset\_neg\_low]中选出的。随机偏移确保了生成的图像块以不同与前者的3D点为中心。

一个积极示例的获取则是通过设置

其中是从区间[−dataset\_pos, dataset\_pos]中随机挑选的。上式中包含而不是将其设为0的原因与后续使用的立体视觉算法有关。特别是，我们发现当网络将低匹配代价分配给良好匹配和相近的匹配时基于交叉的代价聚合表现更好。在我们的实验中，超参数dataset\_pos永远不会大于一个像素。



**图2:** 这个快速架构是一个暹罗网络。 这两个子网络由多个卷积层和紧随其后的整流线性单元（缩​​写为 “ReLU”）组成。通过从每次输入的两个图像块中提取向量并计算它们之间的余弦相似性来获得相似度得分。 在上图和我们的具体实现中，余弦相似性计算可分为两个步骤：归一化和点积。因为每个位置仅需要执行一次归一化操作，所以这种方法减少了运行时间（详见3.3节）

3.2网络架构

为了比较图像块上的相似度，我们提出了两种网络架构。第一种网络架构比第二种运行速度更快，但产生的视差图不太准确。在这两种架构下，网络的输入是一对小图像块，输出是它们之间的相似性的度量。两个架构都包含一个可训练的用特征向量表示每个图像块的特征提取器。图像块之间的相似性通过特征向量上而不是原始图像强度值衡量。快速架构使用固定的相似性度量来比较两个特征 向量，而精确架构试图从特征向量中得到一个更好的相似性度量。

## 3.2.1 快速架构

第一种架构是暹罗网络，即两个共享权重子网加在头部（Bromley等人，1993）。子网络由多个带整流线性单元（除最后一层外）的卷积层组成。两个子网输出一个带有输入图像块属性的向量。得到的两个向量比较使用余弦相似性度量来产生网络的最终输出的。图2提供了该架构的概述。



**图3:** 精确架构从两个卷积特征提取器开始。提取的特征向量串联并通过多个全连接层比较。输入是两个图像块，输出是介于0和1之间的单个实数，我们称这个实数为输入图像相似性的度量。

通过使铰链损耗最小化来训练网络。损失是通过以同一图像位置为中心的含积极和消极两类的示例对来计算的。设是积极实例网络的输出，是消极实例网络的输出，并让余量m为正实数。该实例对的铰链损失定义为max（0，m+）。当积极实例的相似性大于消极实例至少余量m时，损失为零。在我们的实验中，我们设置余量为0.2。

该架构的超参数是每个子网中卷积层的数量（num\_conv\_layers）,卷积内核的大小（conv\_kernel\_size）,每层中的特征映射的数量（num\_conv\_feature\_maps），以及输入图像块的大小（input\_patch\_size）。

## 3.2.2 精确架构

第二种架构是从第一种架构中通过用一些全连接层替换余弦相似性度量得到的（见图3）。

这种架构的变化增加了运行时间，但降低了错误率。两个子网包括很多卷积层，在每层之后紧跟着整流线性单元。两个结果向量串联并通过多个紧跟整流线性单元全连接层前向传播。最后一个全连接层经过sigmoid非线性变换输出一个数字，这个数字就是两图像块的相似度得分。

我们使用二进制交叉熵损失函数来进行训练。令s表示一个训练示例的网络输出，t表示该训练示例的类别;如果示例属于积极示例的类别，则t = 1，如果示例属于消极示例的类别，则t = 0。该示例的二进制交叉熵损失被定义为。

两个架构使用不同的损失函数是基于实验性的证据。虽然我们宁愿对两个架构使用相同的损失函数，但实验表明在精确架构中，二进制交叉熵损失函数表现得比铰链损耗函数更好。另一方面，虽然快速架构的最后一步是余弦相似度计算，但交叉熵损失不直接适用。

精确架构的超参数是在每个子网络中卷积层的数量（num\_conv\_layers），每个层中的特征映射的数量（num\_conv\_feature\_maps），卷积内核的大小（conv\_kernel\_size），输入图像块的大小（input\_patch\_size），每个全连接层中的单元数（num\_fc\_units），以及全连接层数（num\_fc\_layers）。

3.3计算匹配代价

网络的输出用于初始化匹配代价：

其中，是当网络输入输入图像块和时的输出。负号将相似度分数转换为匹配代价。要计算整个匹配的成本代价张量，我们仅需要对每个图像位置和每个需考虑的视差执行前向传递。以下三个实现细节保证了运行时间可管理：

* 两个子网络的输出每个位置只需计算一次，并且不需要针对每个需考虑的差异重新计算。
* 通过传播全分辨率图像，可以为单次前向传递中的所有像素点计算两个子网络的输出，而不仅仅是小图像块。在整个w×h图像上执行单次前向传递比执行w\*h次小图像块的前向传递更块，因为许多中间结果可以重复使用。
* 全连接层在精确架构中的输出也可以在单次前向传递中计算。这是通过用1×1内核的卷积层替换每个全连接层实现的。我们还需要执行每个需考虑的视差的前向传递;对于KITTI数据集，最大视差d为228，对于Middlebury数据集，最大视差为400。因此，网络的全连接部分需要运行d次，这是精确架构的瓶颈。

为计算一对图像的匹配代价，我们每张图片在子网上运行1次，在全连接层运行d次，其中d是需考虑的最大视差。这个理解在设计网络的架构中是重要的。我们可能会选择两个图像在提交给网络之前级联的架构，但这意味着在运行时的大量代价，因为整个网络将需要运行d次。这种理解也导致了快速架构的发展，快速架构中唯一需要运行d次的层是特征向量的点积。

4．立体视觉算法

树叶的。

简洁数学

4.1基于交叉的代价聚合

为了得到最准

检查月牙形圆的边界的方向转变计算。因为位置的变化仅限于4或8个不同的转换（根据轮廓点

4.2半全局匹配

为了得到最准

4.3计算视差图

为了得到最准

5．实验

识别通过对输入图像中提取的HoCS特征进行邻近搜索实现。搜索数据库包括来自于23915

线性扫描数据库检索

蓝色点线图，而像形状识别取得了巨大的进步。

5.1 KITTI立体视觉数据集

蓝色点线图，而像形状识别取得了巨大的进步。

5.2 Middlebury立体视觉数据集

蓝色点线图，而像形状识别取得了巨大的进步。

5.3深度学习细节

蓝色点线图，而像形状识别取得了巨大的进步。

5.4数据集扩充

蓝色点线图，而像形状识别取得了巨大的进步。

5.5运行时间

蓝色点线图，而像形状识别取得了巨大的进步。

5.6匹配代价

蓝色点线图，而像形状识别取得了巨大的进步。

5.7立体视觉算法

蓝色点线图，而像形状识别取得了巨大的进步。

5.8数据集大小

蓝色点线图，而像形状识别取得了巨大的进步。

5.9迁移学习

蓝色点线图，而像形状识别取得了巨大的进步。

5.10超参数

蓝色点线图，而像形状识别取得了巨大的进步。

6．总结

为了允许公

被地方政

参考文献

1. G. Bradski. The OpenCV library. Dr. Dobb’s Journal of Software Tools, 2000.
2. Jane Bromley, James W Bentz, L´eon Bottou, Isabelle Guyon, Yann LeCun, Cliﬀ Moore,
3. Eduard S¨ackinger, and Roopak Shah. Signature veriﬁcation using a siamese time delay neural network. International Journal of Pattern Recognition and Artiﬁcial Intelligence, 7(04):669–688, 1993.
4. Matthew Brown, Gang Hua, and Simon Winder. Discriminative learning of local image descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(1): 43–57, 2011.
5. Thomas Brox and Jitendra Malik. Large displacement optical ﬂow: descriptor matching in variational motion estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(3):500–513, 2011.
6. Ayan Chakrabarti, Ying Xiong, Steven J. Gortler, and Todd Zickler. Low-level vision by consensus in a spatial hierarchy of regions. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
7. Zhuoyuan Chen, Xun Sun, Yinan Yu, Liang Wang, and Chang Huang. A deep visual correspondence embedding model for stereo matching costs. IEEE International Conference on Computer Vision (ICCV), 2015.
8. Sharan Chetlur, Cliﬀ Woolley, Philippe Vandermersch, Jonathan Cohen, John Tran, Bryan Catanzaro, and Evan Shelhamer. cuDNN: Eﬃcient primitives for deep learning. CoRR, abs/1410.0759, 2014.
9. Ronan Collobert, Koray Kavukcuoglu, and Cl´ement Farabet. Torch7: A matlab-like environment for machine learning. In BigLearn, NIPS Workshop, 2011.
10. Nils Einecke and Julian Eggert. A two-stage correlation method for stereoscopic depth estimation. In Digital Image Computing: International Conference on Techniques and Applications (DICTA), pages 227–234, 2010.
11. Andreas Geiger, Martin Roser, and Raquel Urtasun. Eﬃcient large-scale stereo matching. In Proceedings of the 10th Asian Conference on Computer Vision - Volume Part I, ACCV’10, pages 25–38. Springer-Verlag, Berlin, Heidelberg, 2011.
12. Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: the KITTI dataset. International Journal of Robotics Research (IJRR), 2013.
13. Fatma Guney and Andreas Geiger. Displets: Resolving stereo ambiguities using object knowledge. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
14. Ralf Haeusler, Rahul Nair, and Daniel Kondermann. Ensemble learning for confidence measures in stereo vision. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2013.
15. Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C Berg. MatchNet: Unifying feature and metric learning for patch-based matching. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
16. Heiko Hirschmu¨ller. Stereo processing by semiglobal matching and mutual information. IEEE Transactions on Pattern Analysis and Machine Intelligence, 30(2):328–341, 2008.
17. Heiko Hirschmu¨ller and Daniel Scharstein. Evaluation of cost functions for stereo matching. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2007.
18. Heiko Hirschmu¨ller and Daniel Scharstein. Evaluation of stereo matching costs on images with radiometric differences. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(9):1582–1599, 2009.
19. Michael Hornacek, Andrew Fitzgibbon, and Carsten Rother. SphereFlow: 6 DoF scene flow from RGB-D pairs. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014.
20. Dan Kong and Hai Tao. A method for learning matching errors for stereo computation. British Machine Vision Conference (BMVC), 2004.
21. Dan Kong and Hai Tao. Stereo matching via learning multiple experts behaviors. British Machine Vision Conference (BMVC), 2006.
22. Jana Kostkov´a and Radim S´ara. Stratified dense matching for stereopsis in complex scenes. British Machine Vision Conference (BMVC), 2003.
23. Jedrzej Kowalczuk, Eric T Psota, and Lance C Perez. Real-time stereo matching on CUDA using an iterative refinement method for adaptive support-weight correspondences. IEEE Transactions on Circuits and Systems for Video Technology, 23(1):94–104, 2013.
24. Yann LeCun, L´eon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11):2278–2324, 1998.
25. Yunpeng Li and Daniel P Huttenlocher. Learning for stereo vision using the structured support vector machine. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2008.
26. Xing Mei, Xun Sun, Mingcai Zhou, Haitao Wang, Xiaopeng Zhang, et al. On building an accurate stereo matching system on graphics hardware. IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pages 467–474, 2011.
27. Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
28. John Nickolls, Ian Buck, Michael Garland, and Kevin Skadron. Scalable parallel programming with CUDA. Queue, 6(2):40–53, 2008.
29. Mattis Paulin, Matthijs Douze, Zaid Harchaoui, Julien Mairal, Florent Perronin, and Cordelia Schmid. Local convolutional features with unsupervised training for image retrieval. In IEEE International Conference on Computer Vision (ICCV), pages 91–99, 2015.
30. Martin Peris, Atsuto Maki, Sara Martull, Yasuhiro Ohkawa, and Kazuhiro Fukui. Towards a simulation driven stereo vision system. In 21st International Conference on Pattern Recognition (ICPR), pages 1038–1042, 2012.
31. Eric T Psota, Jedrzej Kowalczuk, Mateusz Mittek, and Lance C Perez. Map disparity estimation using hidden markov trees. IEEE International Conference on Computer Vision (ICCV), 2015.
32. Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Deepmatching: Hierarchical deformable dense matching. ArXiv e-prints, 1(7):8, 2015.
33. Daniel Scharstein and Chris Pal. Learning conditional random fields for stereo. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2007.
34. Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. International Journal of Computer Vision, 47(1-3): 7–42, 2002.
35. Daniel Scharstein and Richard Szeliski. High-accuracy stereo depth maps using structured light. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June.2003.
36. Daniel Scharstein, Heiko Hirschmu¨ller, York Kitajima, Greg Krathwohl, Nera Neˇsi´c, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. German Conference on Pattern Recognition (GCPR), September 2014.
37. Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Learning local feature descriptors using convex optimisation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 36(8):1573–1585, 2014.
38. Sudipta N Sinha, Daniel Scharstein, and Richard Szeliski. Efficient high-resolution stereo matching using local plane sweeps. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014.
39. Aristotle Spyropoulos, Nikos Komodakis, and Philippos Mordohai. Learning to detect ground control points for improving the accuracy of stereo matching. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2014.
40. Deqing Sun, Stefan Roth, and Michael J Black. A quantitative analysis of current practices in optical flow estimation and the principles behind them. International Journal of Computer Vision, 106(2):115–137, 2014.
41. Tomasz Trzcinski, Mario Christoudias, Vincent Lepetit, and Pascal Fua. Learning image descriptors with the boosting-trick. In Advances in neural information processing systems, pages 269–277, 2012.
42. Christoph Vogel, Konrad Schindler, and Stefan Roth. Piecewise rigid scene flow. IEEE International Conference on Computer Vision (ICCV), 2013.
43. Christoph Vogel, Stefan Roth, and Konrad Schindler. View-consistent 3D scene flow estimation over multiple frames. European Conference on Computer Vision (ECCV), September 2014.
44. Christoph Vogel, Konrad Schindler, and Stefan Roth. 3D scene flow estimation with a piecewise rigid scene model. International Journal of Computer Vision, pages 1–28, 2015.
45. Koichiro Yamaguchi, David McAllester, and Raquel Urtasun. Efficient joint segmentation, occlusion labeling, stereo and flow estimation. European Conference on Computer Vision (ECCV), September 2014.
46. Ramin Zabih and John Woodfill. Non-parametric local transforms for computing visual correspondence. European Conference on Computer Vision (ECCV), 1994.
47. Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
48. Jure Zbontar and Yann LeCun. Computing the stereo matching cost with a convolutionalˇ neural network. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.
49. Chi Zhang, Zhiwei Li, Yanhua Cheng, Rui Cai, Hongyang Chao, and Yong Rui. Meshstereo: A global stereo model with mesh alignment regularization for view interpolation. IEEE International Conference on Computer Vision (ICCV), 2015.
50. Ke Zhang, Jiangbo Lu, and Gauthier Lafruit. Cross-based local stereo matching using orthogonal integral images. IEEE Transactions on Circuits and Systems for Video Technology, 19(7):1073–1079, 2009.
51. Li Zhang and Steven M Seitz. Estimating optimal parameters for MRF stereo from a single image pair. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(2): 331–342, 2007.