Journal of Network Intelligence
Taiwan Ubiquitous Information
 ISSN 2414-8105 (Online)
Volume **8**, Number **4**, November **2023**

A Review of Deep Learning-based Stereo Matching Algorithms

Kaifeng Wu

College of Electronic Engineering
Heilongjiang University
Harbin, China
wkf439431365@163.com

Xiaofei Wang*

College of Electronic Engineering
Heilongjiang University
Harbin, China
nk_wxf@hlju.edu.cn

*Corresponding author: Xiaofei Wang

Received March 7, 2023, revised May 26, 2023, accepted July 16, 2023.

Abstract. *Stereo matching is an inherent difficulty in stereo vision, which is widely used in many fields to recover image depth information by finding relevant homonymous point pairs through binocular images. The traditional stereo matching process is divided into four steps: cost computation, cost aggregation, parallax computation, and parallax optimization, but with the rapid development of neural networks, deep learning-based stereo matching methods are emerging rapidly, making a breakthrough in stereo matching possible, and deep learning-based methods have achieved excellent results with higher accuracy and faster speed than traditional matching methods. Learning-based matching networks have taken the top position in test benchmarks such as KITTI and Middlebury. Deep learning-based stereo matching methods are classified into non-end-to-end based and end-to-end based matching methods. In this paper, we review the work related to stereo matching and analyze and explain representative stereo matching networks, summarize the advantages of different types of matching networks and the directions for improvement, help researchers understand the work related to stereo matching, and look forward to further breakthroughs in stereo matching work in this field of research.*
Keywords: stereo matching; deep learning; neural network; parallax estimation; depth information

1. **Introduction.** Stereo vision is one of the hot problems in computer vision research and is an important branch of machine vision [1]. It uses the principle of parallax to estimate the depth information of the 3D scene in the corrected binocular image. It has extremely wide applications in robot vision [2], military applications, and medical imaging, aerial mapping, 3D reconstruction [3], autonomous driving [4], depth ranging [5], and industrial inspection [6]. The stereo vision process mainly contains six parts: image acquisition, camera calibration, image correction, feature extraction, stereo matching,and 3D reconstruction [7], among which the stereo matching process is the key step of stereo vision and also the difficult point of stereo vision, whose matching accuracy and efficiency will directly affect the final result of 3D reconstruction and play a great role in the construction of the whole stereo vision system.

The stereo matching process is the process of finding the correspondence between pixels in two images by transforming the two-dimensional search problem into a one-dimensional search problem along polar lines on the corrected left and right camera image pairs, performing parallax estimation by completing the matched pixel pairs, and then calculating the depth of the pixels using fB/d [8]. f is the camera focal length, B is the baseline distance between the two cameras, and d is the parallax estimation the resulting parallax. Traditional stereo matching methods are usually divided into four steps [9]: cost calculation, cost aggregation, parallax calculation, and parallax optimization. The matching cost calculation is used to initially measure the correlation between pixels to be matched in the parallax range, and the matching cost aggregation process aggregates the related pixel surrogate values with the target pixel, which can more accurately reflect the correlation between pixels. The parallax calculation step compares the surrogate values of each target pixel and the pixel to be matched to filter out the pixel pair with the highest correlation, and uses its parallax d as the optimal parallax. The parallax optimization step eliminates the incorrect parallax values that do not meet the conditions to further improve the matching accuracy. Traditional stereo matching methods can be divided into three categories: (1) local stereo matching algorithms [10,11], (2) global stereo matching algorithms [12], and (3) semi-global stereo matching methods [13,14]. The local stereo matching algorithm basically contains the four steps of the traditional stereo matching process by using the winner-take-all (WTA) method to calculate the parallax. The local stereo matching algorithm completes the matching process within the local search window by mainly using the grayscale information of the image itself and its neighborhood pixel point information when searching for matching points. The local matching algorithm can be divided into region based matching algorithm, feature based matching algorithm and phase based matching algorithm according to the selected matching primitives. The advantages are low time complexity of the algorithm, convenient, simple, flexible and versatile, fast operation speed, parallel computation and high efficiency of the algorithm. However, the performance is poor in the weak texture region and texture-free region. In the parallax discontinuity region, it is easy to have large deviation. The global stereo matching algorithm differs from the local stereo matching algorithm in that it does not include the cost aggregation step of display, and it uses the global constraint information of the image to construct a global energy function containing data terms and smoothing terms through the whole image pixel, and uses the method of minimizing the energy function to obtain the parallax map. The global stereo matching methods include confidence-propagation, graph-cuts, and dynamic planning methods. The global stereo matching algorithm has high accuracy and robustness, but the computational complexity is high and time consuming, and the computational efficiency is not high enough for parallel operation. The semi global matching(SGM) algorithm proposed by Hirschmuller [15] also uses the energy function idea, but it solves the NP difficulty problem in the form of multi-path optimization energy function, and the method achieves a good balance between maintaining the accuracy of the results and the computational complexity.

Recently, with the development of neural network [16,17,18,19] and deep learning technology [20,21,22,23], the research trend of stereo matching technology has gradually shifted from the study of traditional stereo matching algorithms to the study of matching algorithms dominated by deep learning. In general, stereo matching algorithms based on deep learning are mainly divided into two categories, namely, non-end-to-end matching algorithms and end-to-end matching algorithms. The pioneering work is the MC-CNN network proposed by Zbontar and LeCun [24], which introduced deep learning technology into stereo matching and successfully used CNN to calculate the cost of matching for the first time, replacing the manually designed cost calculation in the traditional method.

metrics, the MC-CNN network obtained satisfactory results in terms of speed and accuracy compared to the performance of traditional methods in weak texture regions and light imbalance regions. A similar non-end-to-end network is SGM-Net proposed by Seki and Pollefeys [25], which is a non-end-to-end network based on cost aggregation and solves the problem that the cost aggregation process requires artificially designed penalty parameters for different smoothing terms. These non-end-to-end networks have obtained good results, but there are still limitations. Some of the non-end-to-end networks still require manually designed parameters and functions, which are computationally complex and time-consuming and require more hardware resources [26], while the non-end-to-end networks also fail to solve the perceptual field limitation problem and lack relevance by not taking contextual information into account. Therefore, with the success of Mayer et al. in stereo matching work, end-to-end networks are gradually becoming applicable, and although deep learning techniques have greatly improved the traditional stereo matching work, the emergence of end-to-end network models has undoubtedly broken the barriers and led the stereo matching work on another new path.Unlike the non-end-to-end network which is an alternative to the traditional stereo matching step, the end-to-end stereo matching network takes the left and right views as input and obtains the mapping result from the trained neural network to directly output the parallax map. The end-to-end network models are also classified into two types of end-to-end networks consisting of 2D encoder-decoder [27] structures and end-to-end networks guided by regularization modules consisting of 3D convolution, according to their different architectures. Representative related works include the end-to-end regression network Disp-Net [28], first proposed by Mayer et al. using an encoder-decoder structure to compute correlated 3D costumers from left and right image features (encoding), while using CNN regression to obtain parallax map results (decoding), respectively. The next pioneering network is GC-Net [29], which processes a 4D costomer (height, width, parallax, number of feature channels) composed of stitched left and right image monolithic features extracted by two encoders with shared weights through a 3D CNN regularization module, while the network incorporates 4D costomer contextual information and is the first in the KITTI benchmark to The first end-to-end network that outperforms a manually designed end-to-end network. In addition to the aforementioned non-end-to-end and end-to-end networks, there exists a third type of classification, namely unsupervised stereo matching networks, due to their reliance on large amounts of labeled training data. Unsupervised learning-based [30] matching networks greatly simplify the training process of the network. Godard et al. [31] reconstructed images using polar line geometric constraints without using ground truth depth and generated parallax maps by training the network while constructing a new loss function back propagation to further train the network to improve robustness. Zhou et al. [32] iteratively updated the network parameters in an iterative manner to gradually converge a random network to a steady state, and the obtained results achieved no inferiority to the supervised learning matching methods in various stereo matching benchmarks. However, although these unsupervised learning methods yielded satisfactory results in the benchmarks, incorporating the monocular view approach into stereo matching is inherently difficult and lacks reliability compared to supervised learning networks trained with real data labels. In some mainstream benchmark tests today, such as KITTI 2012, KITTI 2015, Middlebury, etc., stereo matching methods based on deep learning basically occupy the front-end position and outperform traditional matching methods in terms of performance, becoming the mainstream of stereo matching research work. In order to help researchers sort out and further contribute to stereo matching work, this paper summarizes stereo matching work based on deep learning in recent years, mainly summarizing end-to-end matching methods and non-end-to-end matching methods supported by large

and rich real label data, and comparing the performance of these algorithms in different benchmark tests to help researchers of stereo matching work further grasp the advantages and disadvantages of various methods.

2. **Non-end-to-end Stereo Matching.** The non-end-to-end network is an alternative to one or more steps in the traditional four-step stereo matching process illustrated in Figure 1, so in the research and development of non-end-to-end stereo matching networks, researchers have designed stereo matching networks with different focuses, including and not limited to matching networks with a focus on matching cost, by designing convolutional neural networks with different learning features and metric functions instead of combining hand-designed features and similarity measures, and using them for cost calculation to improve the accuracy of stereo matching algorithms. Non-end-to-end networks also include matching networks based on cost aggregation. The traditional cost aggregation process considers the association and influence of the target pixels with the pixels in their neighboring regions and requires manual parameter setting, while the cost aggregation-based end-to-end networks use a learning approach to obtain the required parameters for the aggregation process and improve the aggregation effect. Remaining related works such as designing optimization networks using residual information or multi-stage strategies to perform parallax optimization on the resulting parallax after cost aggregation, a non-end-to-end approach based on parallax optimization, are also part of the researchers' attention. The following is a categorical summary and overview of some excellent or seminal work on end-to-end networks.

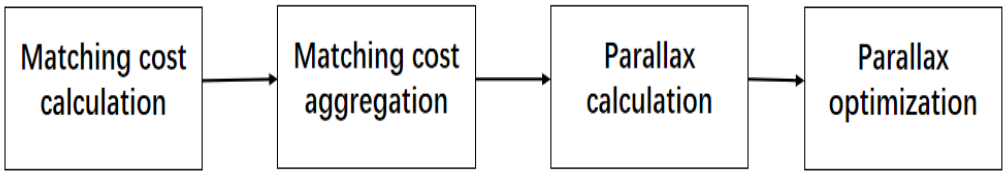
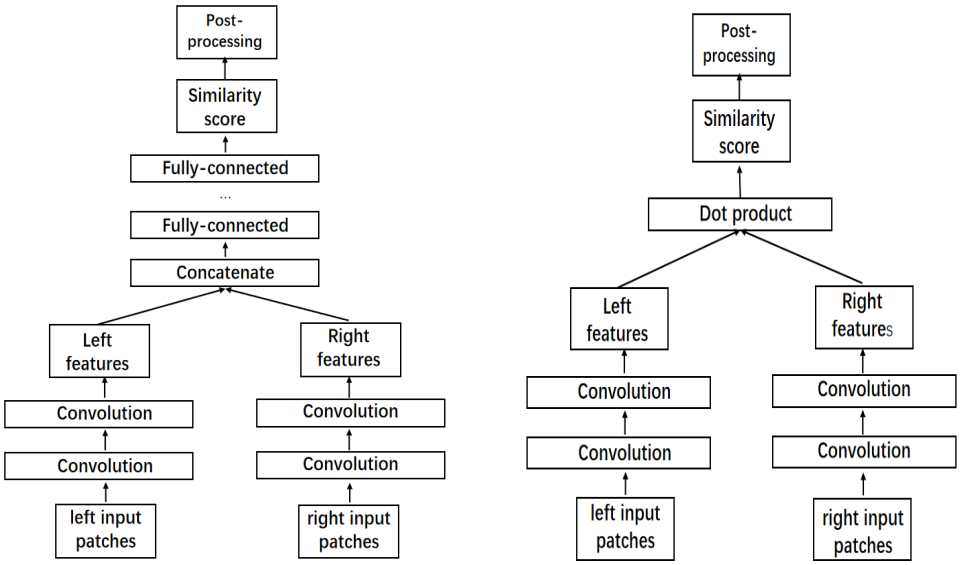


Figure 1. Traditional matching algorithm steps

2.1. **Cost Calculation.** The development of MC-CNN [24] network in this field is undoubtedly a pioneering work, where the trained CNN network can obtain the matching scores of the target image blocks in the left and right images and thus predict whether these two target image blocks match or not. Zbontar and LeCun used a twin network in the network framework of MC-CNN-art (Siamese Network), which is implemented by sharing weights, can evaluate the similarity of two inputs, as in Figure 2(a). In the MC-CNN-art network, the left and right images are used as input, and the window size is chosen as 9×9 to obtain the image blocks. The left and right image blocks are passed through a convolutional layer consisting of 32 kernels of size 5×5 for feature extraction, and the features are processed through two fully connected layers with 200 neurons each, and the left and right processed features are connected to generate a 400-dimensional feature vector, which is again passed through The similarity measure of the central pixels of the two image blocks is obtained by passing through multiple fully connected layers with 300 neurons each. After obtaining the cost metric, the network further aggregates the cost using traditional methods such as cross-arm aggregation crossover (CBCA) [33] and semi-global matching (SGM) along the scan line aggregation proposed by Mei et al. and finally obtains the parallax map after parallax calculation and optimization steps. The authors also proposed another MC-CNN-fst network, which is different from the MC-CNN-art

network that uses network learning to obtain the similarity metric, and this network uses the vector inner product as the similarity metric instead of the complex fully connected layer as in Figure 2(b), and the performance decreases as a result of the decrease in network complexity, which also makes this network framework increased compared to the MC-CNN-art framework error and a slight decrease in matching accuracy, but reduces the time cost and greatly enhances the computational speed. With the performance on



(a) MC-CNN-acrt (b) MC-CNN-fst

Figure 2. Siamese network structures:MC-CNN-art and MC-CNN-fst

the KITTI 2012, KITTI 2015, and Middlebury datasets, these methods have successfully demonstrated that CNNs extract image features and perform similarity measures more accurately and efficiently than the manually designed traditional methods, and thus a number of works have been inspired by the twin network architecture. Chen et al. [34] proposed Deep Embedding network, which uses convolutional kernels of different sizes to embed multi-scale features into two parallel networks, but the characteristics of the twin network (Siamese Network) architecture increase the time cost due to the need to pass through multiple fully connected layers when further processing the features. For example, when a twin network (Siamese Network) is assumed to have a matching cost time for forward inference of parallax , then an image of size with minimum parallax 0 and maximum parallax D requires a computation time of $M * N * (D + 1) * T$, when the inference time of the twin network is large, the time cost consumption reduces the efficiency of the overall framework. Therefore, the same cost calculation method as MC-CNN-fst architecture is used in the Deep Embedding network architecture, where the multi-scale features extracted by the twin network are dotted to obtain the final matching score. This further improves the time efficiency while solving the problem that fixed-size kernels do not work perfectly in different regions. Luo et al. [35] also used the twin network architecture to propose Content CNN, which uses the same vector dot product method as the above network to calculate the cost score to reduce the computational burden, while it uses a multi-classification model for parallax instead of a binary classification model for parallax, which improves the accuracy while providing security for the cost aggregation work that follows. Other scholars have proposed different network architectures focusing on more complex network structures, such as influenced by twin network architectures, Zagoruyko

and Komodakis proposed different classes of neural networks such as pseudo-twin [36], 2-channel, etc. Chen and Yuan [37] used Park and Lee add a 4P layer (Per-Pixel Pyramid Pooling) [38] so that the generated features have coarse to fine information to access a wider range of contexts. This expands the perceptual field without loss of resolution and detail.

2.2. **Cost Aggregation.** After obtaining the initial cost, traditional methods use cost aggregation to correlate the cost of a single pixel with the cost of its surrounding pixels (usually pixels on the aggregation path or neighborhood-related pixels) to further improve the matching cost correlation, and the commonly used aggregation methods are SGM along-scan aggregation with two penalty parameters $P1$ and $P2$ set manually to control the effect between different parallax differences when aggregating the cost along the path. The commonly used aggregation methods are SGM along-scan aggregation, where the effect of different parallax differences is controlled by manually setting two penalty parameters $P1$ and $P2$. Other commonly used aggregation methods are cross-cross-arm aggregation, which improves the accuracy by constructing closely associated cross-arm regions for the target pixels and aggregating only the surrogate values of the pixels in the regions during the cost aggregation process. However, these methods usually rely on a priori knowledge and have to manually set penalty parameters such as in the smoothing term, thus leading to the study of non-end-to-end networks based on cost aggregation. In the work of scholars Seki and Pollefeys SGM-Net [25] framework was proposed to distinguish positive and negative parallax transitions along the scan line based on different occlusion relations, which can give satisfactory results even in pathological regions. The framework automatically learns the penalty parameters by CNN, taking a 5×5 grayscale image block with its position parameters as input and the predicted results of SGM penalty parameters as output. The network introduces a new loss function to adjust the smoothing penalty for pixels with different confidence levels. The loss function consists of a path cost, which calculates the path cost by considering the difference between the parallax values of pixels to be aggregated along the scan line and their true parallax, and a neighborhood cost, which considers the transition differences between different neighboring pixels, such that the neighboring edge pixels should be penalized more during the aggregation process.The cost of the neighborhood cost is calculated. The network is thus trained to obtain penalty parameters to ensure that different pixels are subject to different smoothing terms, so that information can be propagated from reliable pixels to unreliable pixels along the scan line. Related work is also based on the combination of high confidence pixels and random forest algorithm by Spyropoulos et al. By using the pixels selected by the random forest classifier as ground control points (GCP) [39,40] and adding soft constraints to their matching, minimizing the MRF [41] energy is optimized to allow reliable pixels in the global framework to Schonberger et al. similarly proposed SGM-Forest [42] using a random forest classifier to select scanlines of different orientations for fusion based on the differences between pixels and select the optimal scanline path for each location pixel, replacing the simple summation combination of the original multi-directional scanline paths, thereby obtaining the confidence level.

2.3. **Parallax Optimization.** Parallax optimization is the last step of the matching process in traditional matching methods, which is used to eliminate unreliable parallax values after winner-take-all (WTA), commonly used methods such as left-right consistency check (LRCR) and uniqueness constraint, and after eliminating error points, interpolation, filtering and other methods are used to fill smooth parallax map holes and improve parallax map results. And the non-end-to-end network based on parallax optimization assists network optimization by introducing residual information and other methods to

achieve more reliable optimization results instead of traditional image filtering and other optimization means. Therefore, Jie et al. [43] proposed the LRCR network model, which can run parallax estimation in parallel with left-right consistency checking, and correct parallax values for unreliable regions by using iterative cyclic learning, and in each cycle, the correction process pays more attention to unreliable regions where more erroneous parallaxes exist due to the introduction of the soft attention mechanism of the LRCR framework . There are also scholars who use the idea of residual information correction in optimization networks for optimization, such as Batsos and Mordohai proposed RecRes-Net [44] to correct erroneous parallaxes by combining multi-scale residual information at multiple resolutions. The DRR [45] (Detect, Replace, Refine), which uses residual correction to improve the output parallax, is also outstanding, and the network framework proposed by Gidaris and Komodakis ensures the speed while maintaining the optimization effect. By repeating the process of detecting the wrong parallax, replacing the wrong parallax with a new one, and correcting the new parallax with the residual information, the DRR achieves an outperforming result in the KITTI 2015 test.

3. **End-to-end Stereo Matching.** However, these non-end-to-end methods still inevitably require some regularization functions to be designed manually to obtain the final results, and their network frameworks usually waste computational resources, e.g., the DRR in the parallax optimization network framework repeats its three processes continuously, which increases the computational burden, and Most of the post-processing processes of parallax optimization are accompanied by such problems. Moreover, non-end-to-end networks cannot correlate image context information well, and it is difficult to solve the problem of limited perceptual field. Therefore, the proposed end-to-end networks have gradually made an impact on the traditional matching methods and non-end-to-end matching methods, and the research on end-to-end matching methods has gradually become mainstream with the success of Mayer et al. Broadly speaking, end-to-end networks can be divided into two types of networks: 2D architecture networks and 3D architecture networks, which differ in their strategies for feature processing and ensemble encoding of cost bodies. 2D architecture uses Correlation operation to process features to build cost volume and uses 2D encoder-decoder to process cost bodies,as shown in Figure 3.3D architecture processes features to build cost bodies by concatenation operation processing of features to construct the cost body, and processing of the cost using a regularization module composed of 3D convolution. In the subsequent sections, the different network architectures are discussed according to their respective features and operating principles, which are highlighted and corroborated with excellent end-to-end algorithms to summarize them.

3.1. **2D Architecture.** Mayer's proposed parallax regression network Disp-Net [28] is a milestone in end-to-end parallax regression work, which consists of two frameworks, Disp-NetS and Disp-NetC, and they use a 2D encoder-decoder structure. The 2D encoder-decoder consists of a series of 2D CNN convolutional stacks, which are more often used in work such as semantic segmentation, optical flow, etc. Feature extraction is performed in the encoder part, and the last layer of the encoder generates in a low-scale parallax map based on a 1-in-64 feature map, while the resolution is extended to 1-in-32 by deconvolution of that scale feature map with the parallax map, and the decoder part The resolution is increased by up-convolution, and in order to compensate for the spatial information lost due to the downsampling process, the expanded parallax map is spliced with the corresponding scale features by jump connection, so that more semantic and detailed information can be used in the next deconvolution, and thus the contextual information

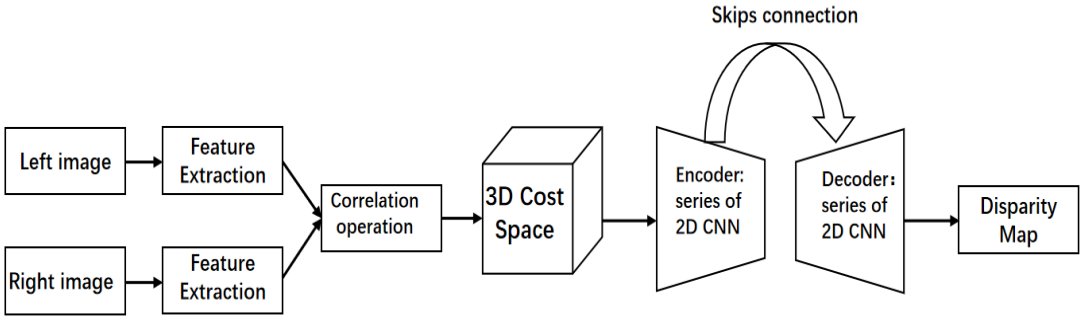


Figure 3. END-to-End network 2D architecture

is fused to expand the perceptual field and get a more accurate parallax map, which has a resolution of half of the input image resolution. The Disp-NetC architecture is influenced by FlowNetC proposed by Dosovitskiy et al. to solve the optical flow estimation problem [46], which differs from Disp-NetS in that it takes a stack of left and right RGB images as input and implicitly learns the correlation between the two image features, and uses a twin network with shared parameters to separately input the left and right images for feature extraction, and uses the the correlation module performs vector inner product of the extracted features to simulate the cost calculation process to generate the 3D cost space. The 2D encoder-decoder architecture is used in the subsequent part of the network to regress the final parallax map, and in the evaluation of real datasets at KITTI 2012 and KITTI 2015, Disp-NetC is nearly 1000 times faster than MC-CNN-art, which is less effective but sufficient to demonstrate the superiority of the end-to-end network. However, although Disp-Net has achieved good results, the inherent problems caused by weakly textured regions, monotonous repetitive regions, and occlusion are still difficult to be solved by Disp-Net, so many subsequent works are being carried out to solve the problems based on the Disp-Net framework.

The first one to achieve optimal results on the KITTI dataset is the CRL (Cascade Residual Learning) proposed by Pang et al. [47], which uses a multi-stage learning approach and is divided into two stages, as shown in Figure 4. In the first stage, DispFulNet acquires the parallax map, the inputs are left view I_L and right view I_R, and an additional upper convolution module is added on top of the DispNetC architecture, which is different from DispNetC in that the network outputs a full-resolution parallax map containing more detailed information. In the second stage, DispResNet implements parallax optimization, the input is the parallax$d_1$obtained in the first stage, the left and right views, the left view $\widetilde{I}_L$ synthesized according to the parallax on the right wrap, the error between the left view and the synthesized left view e_L. through the hourglass structure, combined with the first stage to generate a multi-scale residual signal, the final parallax map is output through the residual information for parallax optimization.

The success of the CRL architecture demonstrates that learning residuals can be more effective and efficient than learning parallax directly in the secondary network. The residual learning not only refines the parallax map better, but also facilitates the fine-tuning of the whole network and alleviates the overfitting problem. A similar idea is used in iResNet [48] proposed by Liang et al. The difference is that the framework allows iterative optimization of the parallax, allowing iterative application of the optimization module for tuning, and the information shared between the first and second stages of iResNet is more than CRL. The iResNet ranked first in the 2018 Robust Vision Challenge.

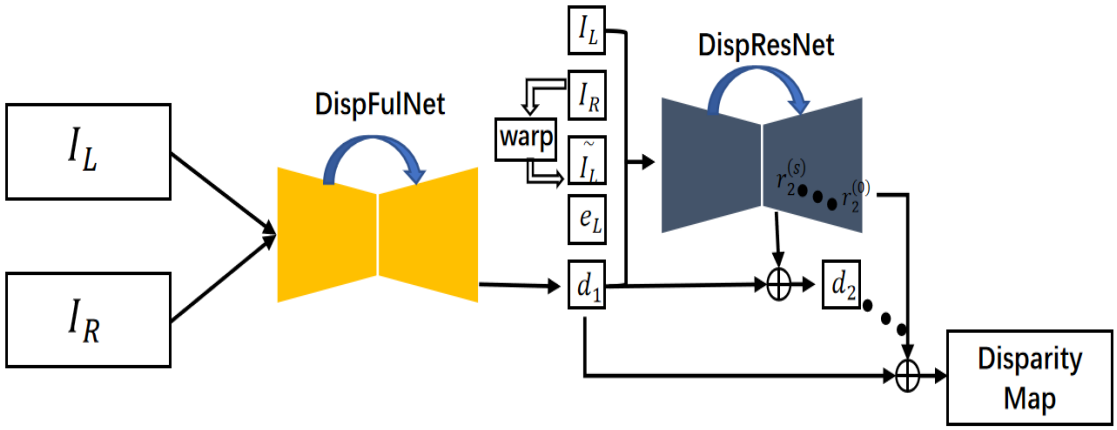


Figure 4. Multi-stage learning CRL network

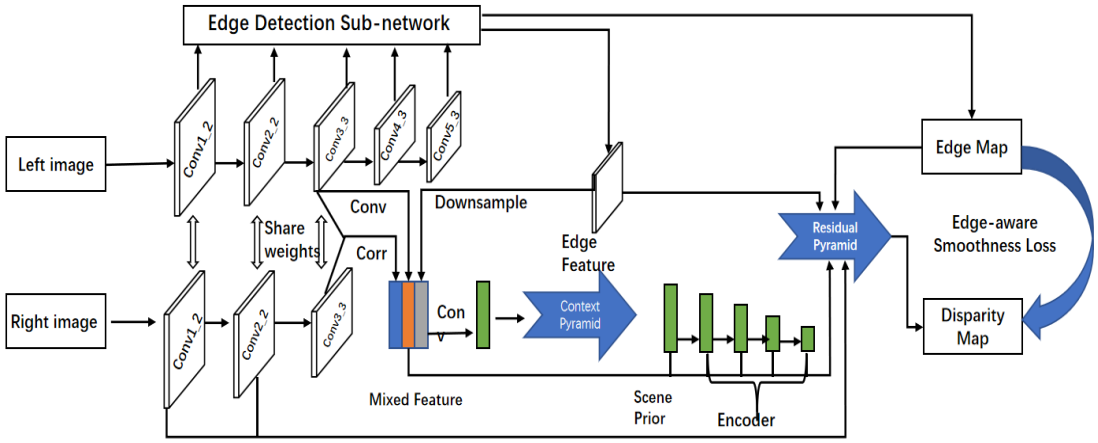


Figure 5. Optimized cascade network EdgeStereo

However, although parallax estimation by cascade structure has achieved satisfactory results in parallax optimization using the residual signal, learning the residual signal is actually fraught with difficulties because the initial parallax, which is usually very well predicted, converges the residual to zero and the complex network structure increases the computational burden. Therefore, Song et al. [49] proposed EdgeStereo to optimize the cascade structure by using residual pyramids in a single network to obtain parallax and optimize it. The EdgeStereo network consists of a parallax network and an edge sub-network, where the parallax network predicts the parallax map containing two modules, a pyramid module for encoding multi-scale contextual information of the pathological region and a residual pyramid module for refining the process of the residual pyramid module,as shown in Figure 5. The sub-network module obtains edge information with improved detail through feature embedding and edge-aware smoothing loss regularization. The main network part *conv*1_1 to *conv*3_3 extracts quarter resolution image features F_l and F_r carrying semantic information, obtains cost F_c through correlation layer, extracts reduced features F_r^l on feature F_l, connects F_c and F_r^l with edge features extracted by edge sub-network to generate hybrid features, context pyramid connects the outputs of the four branches of the context pyramid with the hybrid features into a scene prior to provide multi-scale contextual cues and low-level semantic information for parallax

estimation, and the full-resolution parallax map is obtained through Residual Pyramid refinement. And Cooperation of Edge Cues continuously supervises the parallax map prediction by edge information, so that the parallax variation in edge regions is larger and the inspection variation in non-edge regions is smaller. Yang et al. [50] proposed the SegSereo network model, which introduces semantic information into the matching work and introduces semantic maximum loss (softmax loss) greatly improves the parallax accuracy and achieves optimal results in the KITTI test benchmark.

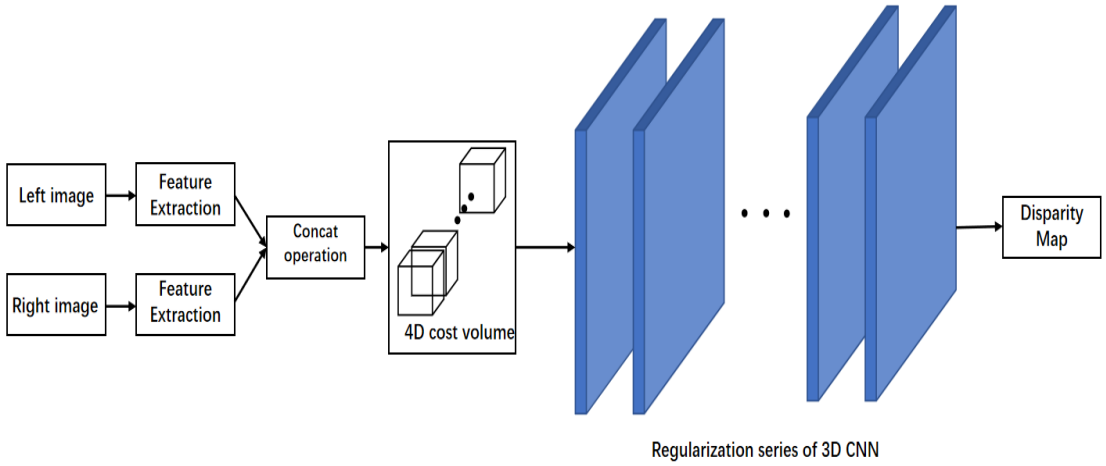


Figure 6. End-to-End network 3D architecture

3.2. **3D Architecture.** Unlike 2D architecture end-to-end networks that are derivatives of traditional neural networks, 3D architecture end-to-end networks are networks that emerged specifically for the development of stereo matching work,as shown in Figure 6. They are different from networks and variants such as the 2D architecture DispNet, which are more focused on designing a network dominated by 3D convolutional regularization, with the advantage that the generated 4D costomer retains more details, geometric features of the image with contextual information, and the disadvantage that the computational complexity is too high, requiring more memory usage and sacrificing the running time. The GC-Net network proposed by Kendall et al., one of the first approaches to propose Cost Volume, is also the first end-to-end network that outperforms traditional matching methods in the KITTI test benchmark. The first step of GC-Net extracts left and right image features through 2D convolutional layers, downsamples the input image and accesses the Backbone of ResNet to output a feature map with half resolution. In the second step, using the feature construction Cost Volume, the left feature map and the D (maximum parallax) right feature map generated with a pixel-by-pixel shift along the parallax direction are concatenated one by one, thus aligning the left and right map feature points, traversing all possible parallaxes, and forming a 4D tensor of $\frac{1}{2}W \times \frac{1}{2}H \times \frac{1}{2}D \times C$ for each stereo image pair, with W and H being the impact width and height, D being the maximum parallax, and C being the number of feature channels. The third step learns the regularization function that can combine the context and optimize the parallax, using multi-scale 3D convolution and deconvolution to form a decoder-encoder for upsampling and downsampling, and a separate deconvolution module to upsample the cost body back to the original image size and process the cost body to get a cost body tensor of $W \times H \times D \times 1$ with the same size as the original image. In the fourth step, the parallax map is regressed and the soft argmax is applied to the cost bodies. soft argmax

is fully differentiable and the probability values of the matching cost bodies are used as weights to weight and sum each parallax to obtain the parallax map,as shown in Figure 7.

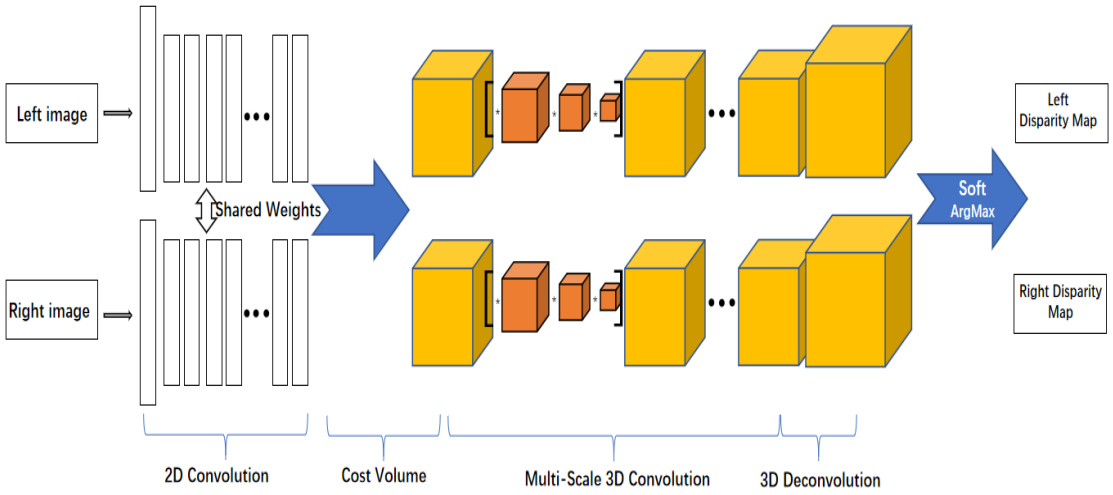


Figure 7. End-to-End 3D architecture network GC-Net

A large part of the excellent effect of GC-Net network originates from the fact that its constructed Cost Volume contains four dimensions, which means that there is more image stereo geometry information applied to the network framework. Influenced by GC-Net network, Chang and Chen [51] added more information to the network learning to pursue the accuracy improvement, thus proposing PSM-Net.The main innovation of this network is the introduction of the spatial pyramid pooling module SPP (spatial pyramid pooling), which can better utilize the contextual information compared to GC-Net, and the SPP module uses adaptive average pooling to compress the features to a 4-scale average pooling and The feature dimensionality is reduced by a 1×1 convolutional layer, and then the low-dimensional feature maps are sampled and concatenated to aggregate multi-scale information to build Cost Volume to make full use of global information. Next, the network proposes stacked hourglass networks, which are essentially stacked 3D encoder-decoder structures, to regularize 4D Cost Volume to predict parallax. Based on PSM-Net, Yang et al. [52] proposed GWC-Net, which proposes a group-wise strategy, grouping multichannel feature maps according to channels to improve parallax accuracy by fully considering the correlation of feature channels while retaining the advantages of the original Cost Volume construction method. However, adding more information into the network framework improves the accuracy, but also increases the computational burden and sacrifices the running time. This is why many networks are working in the other direction to reduce network complexity. Therefore, there are also many networks that work in the other direction to reduce the complexity of the network and pursue the improvement of the speed of the transport loss. Related to make to solve this problem is the GA-Net proposed by Zhang et al. [53], the main work of this network is the optimization made to the part from obtaining cost colume to softmax regression parallax, by proposing two 2D convolutional layers with fewer parameters instead of the traditional stacked 3D convolution with higher parameters to achieve nearly the same effect with fewer parameters. One of the two convolutional layers is the SGA (semi-global aggregation layer), which is a modification of the traditional semi-global matching SGM and is a differentiable approximation of the SGM method, which aggregates the cost of multiple

different directions and is the network framework to get accurate parallax estimation in the occlusion region, weak texture region, and other non-regions. Secondly, the locally guided aggregation layer (LGA), which is mainly used to solve the edge blurring problem, deals with thinner structures and object edges by aggregating local costs to refine parallax. Lu et al. [54] proposed SCV-Net (Sparse Cost Volume Net) in the process of generating from features cost colume from features, which significantly reduces memory usage to increase speed. Also using the pyramid model is AnyNet proposed by Wang et al. [55]. This network uses a coarse-to-fine strategy to extract feature maps at three scales and obtain parallax results by deploying the pyramid model. Yang et al. [56] also uses the pyramid strategy to propose Hierarchical Stereo Matching (HSM) network, which solves the problem of high-resolution [57,58] stereo matching by extracting different resolution features and calculating the cost according to the resolution. Other works achieved better results by designing function-specific networks in combination with existing networks. Liu et al. designed dynamic self-assembly optimization strategies applied to cost distribution and parallax map, respectively, and proposed Lac+GANet [59] network combining network architectures GwcNet and GA-Net to significantly improve the module performance. Xu et al. [60] proposed ACVNet, constructing a new cost volume whose generated attention concatenation volume (ACV) suppresses redundant information to increase relevance and uses a more lightweight aggregation network while ensuring accuracy. Cheng et al. [61] proposed a hierarchical NAS framework end-to-end search network LEAStereo, which jointly optimizes the entire network framework and ranks high in various benchmark tests.

In summary, end-to-end networks have been increasingly effective in stereo matching work, and these end-to-end networks with 2D and 3 architectures aim to solve the accuracy problem in the maladaptive region. 2D architectures continuously optimize the network mainly with learning ideas such as multi-stage and multi-tasking in order to pursue a better combination of contexts to improve network reliability. 3D architectures emerge for stereo matching work The end-to-end network of 3D architecture emerges for stereo matching work, with the main research of optimizing cost colume or reducing the number of 3D convolutions to achieve the pursuit of parallax accuracy or matching speed.A summary of the learning-based stereo matching algorithm classification is shown in Figure 8.

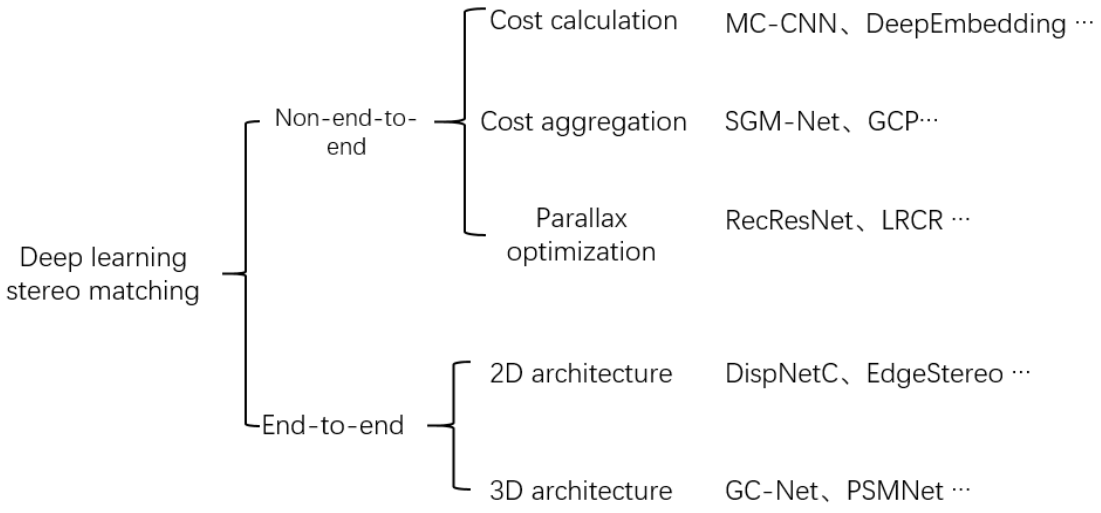
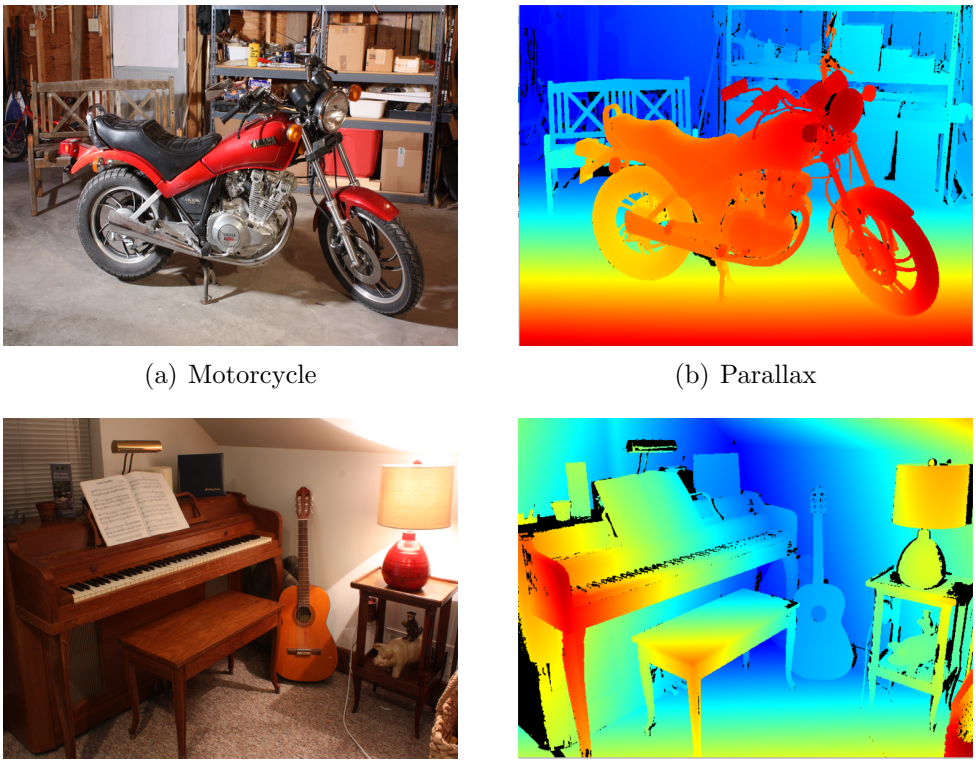


Figure 8. Deep learning for stereo matching algorithm classification and algorithm enumeration

4. Results.

4.1. **Dataset. KITTI** KITTI test [62] benchmark contains two benchmarks, KITTI 2012 and KITTI 2015. KITTI 2012 contains 194 training images and 195 test images, which is the first stereo matching dataset containing outdoor static scenes that rank each matching algorithm according to specified pixel errors in different criteria, including Non-occluded areas (Out-Noc), Out-All areas (Out-All), Average disparity / end-point error in non-occluded areas (Avg-Noc), Average disparity / end-point KITTI 2015 contains 200 training scenes and 200 test scenes, unlike the benchmark KITTI 2012, which contains dynamic scenes for which a pixel is correctly estimated if the parallax error or end-point error is$< 3px$ or $< 5\%$.

Middlebury Middlebury [63] test benchmark contains several different datasets ranging from 2001 to 2021. The benchmark uses the Middlebury 2014 dataset as a public processing standard, which consists of 33 subpixel-level indoor still scenes, including 13 training pairs, 10 additional pairs, and 10 test pairs. Full resolution F, half resolution H and quarter resolution Q of these images are officially provided for researchers to use. The official tool MiddEval3 is provided to help researchers compare the data with the real parallax GT maps, and the experimental results are uniformly submitted and ranked on the website. the Middlebury 2014 dataset provides parallax in PFM format, so the parallax maps required for comparison must also be in PFM floating point format.Some of the data are shown in Figure 9.



(a) Motorcycle (b) Parallax

(c) Piano (d) Parallax

Figure 9. Middlebury data set motorcycle with piano and parallax chart

4.2. **Evaluation Indicators.** The main evaluation annotations for stereo matching work are the root mean square error (RMS) and bad point rate B proposed by Scharstein and Szeliski in 2002 [64], which are the evaluation annotations for most of the mainstream

datasets, while the mean absolute error A is also considered as a measure of matching accuracy, calculated as follows, respectively.

$$R = (\frac{1}{N}\sum_{(x,y)} |d_C(x,y) - d_T(x,y)|^2)^{\frac{1}{2}} \tag{1}$$

$$B = (\frac{1}{N}\sum_{(x,y)} |d_C(x,y) - d_T(x,y)| > \delta_d) \tag{2}$$

$$A = (\frac{1}{N}\sum_{(x,y)} |d_C(x,y) - d_T(x,y)|) \tag{3}$$

where $d_C(x,y)$ is the parallax predicted by the matching method, $d_T(x,y)$ is the true parallax, δ_d is the set parallax threshold, and N is the image size.

Table 1. Stereo matching algorithm classification

Classification	**Methods**	**Advantages**	**Disadvantages**	**Related Work Improvement**
Traditional stereo matching	SGM,AD-Census, etc	Simple,no real parallax required,adaptable,low resource consumption	Slow speed, low accuracy, manual design parameters required	Improvements to the traditional four steps, etc.
Non-End-to-End Networks	MC-CNN SGM-Net, etc.	Higher accuracy than traditional methods, improved network on traditional steps, better parallax map results	Requires manual design parameters,high computational burden, does not take full advantage of contextual information, and limited perceptual field	multi-stage learning. Multi-network architecture learning, multi-task learning, etc.
End-to-End Networking	Disp-Net GC-Net,etc	Leverages contextual information to expand the field of sensing with high accuracy compared to non-end-to-end networks	High network complexity and computational burden, large memory consumption and high runtime	Optimize cost colume or reduce the number of 3D convolutions, etc. to improve accuracy or speed up

4.3. **Experimental Comparison and Analysis.** As shown in Table 4, Out-Noc is the percentage of error pixels in the non-occluded region, and Out-All is the percentage of error pixels in the global region. Table 1 shows the experimental results of each classical matching method selected by classification for different error thresholds in KITTI 2012. It can be seen that among the non-end-to-end matching methods, the results achieved by the matching methods based on cost calculation, generation aggregation, and parallax optimization are not very different, and the matching networks since the pioneering work such as MC-CNN and SGM-Net have significantly improved in accuracy compared with the traditional matching algorithm SGM, however, the non-end-to-end networks led by these two networks consume a large amount of computational resources and the overall algorithm runs at a significantly reduced speed. The end-to-end matching network also achieves significant improvement over the classical network for both 2D and 3D architectures. The end-to-end network has higher accuracy and faster overall comparison than the non-end-to-end network due to the full contextual expansion of the sensory field. The

Table 2. KITTI 2012 Algorithm Comparison

Method	Family	2 pixels		3 pixels		4 pixels		5 pixelss		Runtime (s)	Environment
		Out-Noc	All	Out-Noc	All	Out-Noc	All	Out-Noc	All		
SGM	Traditional	8.66	10.16	5.76	7.00	4.38	5.41	3.56	4.41	3.7	1 core @ 3.0 Ghz (C/C++)
MC-CNN-art	Consideration calculation	3.90	5.45	2.43	3.63	1.90	2.85	1.64	2.93	67	Nvidia GTX Titan
DeepEmbedding	Consideration calculation	5.05	6.47	3.10	4.24	2.32	3.25	1.92	2.68	3	Nvidia GTX Titan (CUDA, Caffffe)
Content-CNN	Consideration calculation	4.98	6.51	3.07	4.29	2.39	3.36	2.03	2.82	0.7	Nvidia Titan X (CUDA)
SGM-Net	Cost aggregation	3.60	4.92	2.37	3.09	1.97	2.52	1.72	2.17	67	Nvidia (R) Titan X (Torch7)
RecRes Net	Parallax optimization	3.37	4.38	2.21	2.94	1.73	2.31	1.45	1.92	0.3	GPU @ NVIDIA TITAN X (Tensorflow)
Disp-NetC	2D architecture	7.38	8.11	4.11	4.65	2.77	3.20	2.05	2.39	0.06	Nvidia Titan X
iResNet	2D architecture	2.69	3.34	1.71	2.16	1.30	1.63	1.06	1.32	0.12	Nvidia Titan
Edge Stereo	2D architecture	2.32	2.88	1.46	1.83	1.07	1.34	0.83	1.04	0.32	Nvidia GTX 1080Ti (Caffffe)
SegStereo	2D architecture	2.66	3.19	1.68	2.03	1.25	1.52	1.00	1.21	0.6	Caffe
GC-Net	3D architecture	2.71	3.46	1.77	2.30	1.36	1.77	1.12	1.46	0.9	Nvidia Titan
PSMNet	3D architecture	2.44	3.01	1.49	1.89	1.12	1.42	0.90	1.15	0.41	Nvidia Titan Xp (CUDA)

comparison of 2D and 3D architectures shows that the more complex network architecture of the 3D architecture has better accuracy than the 2D architecture, but also slightly slower operation speed.

As shown in Table 3, D1-bg, D1-fg, and D1-all represent the background region, foreground region, and all regions, respectively,where the error threshold is limited to 3 pixels, and this dataset is selected from KITTI 2015 with Middlebury 2014, as seen in the table more non-end-to-end networks are joined with end-to-end networks and the accuracy has improved, and throughout the benchmark test, deep-based Throughout the benchmark tests, stereo matching methods based on deep learning have dominated the head rankings, and their matching methods are often inspired by these classical networks such as GANet+ADL, the current matching algorithm ranked first in KITTI 2015, which is a combination of GANet and other algorithms. From the comparison of KITTI and Middlebury test benchmark experimental results, the overall speed of the end-to-end network is significantly better than traditional methods and non-end-to-end matching methods. Algorithm accuracy and speed are hardly compatible, for example, SGM-Forest has the highest accuracy with 7.37 bad point rate compared to other algorithms in the Table 4 in Middlebury 2014 test benchmark, but consumes 88.5s which is nearly 50 times slower than MC-CNN-fst. And MC-CNN-fst also obtains nearly 90 times faster than MC-CNN-acrt at the expense of accuracy. It is increasingly difficult for current traditional matching algorithms to achieve better matching methods than deep learning, and it is even difficult to squeeze into the list. Deep learning-based stereo matching algorithms have gradually outperformed traditional matching methods in terms of matching accuracy, but research on more robust stereo matching algorithms is still necessary.

Table 3. KITTI 2015 Algorithm Comparison

Family	Method	All-pixels			Non-Occluded pixels			Runtime (s)	Environment
		D1-bg	D1-fg	D1-all	D1-bg	D1-fg	D1-all		
Non-End-to-EndNetworks	MC-CNN-art	2.89	8.88	3.89	2.48	7.64	3.33	67	Nvidia GTX Titan X(CUDA,Lua/Torch7)
	Connent-CNN	3.73	8.58	4.42	3.32	7.44	4.00	1	Nvidia GTX Titan X (Torch)
	SGM-Net	2.66	8.64	3.66	2.23	7.44	3.09	67	Titan X
	SGM-Forest	3.11	10.74	4.38	2.79	9.70	3.93	6	1 core@3.0 Ghz (Python/C/C++)
	LRCR	2.55	5.42	3.03	2.23	4.19	2.55	49.2	Nvidia GTX Titan X
	RecRes Net	2.46	6.30	3.10	2.23	5.37	2.75	0.3	GPU @ NVIDIA TITAN X (Tensorflow)
	DRR	2.58	6.04	3.16	2.34	4.87	2.76	0.4	Nvidia GTX Titan X
End-to-End Networks	DispNetC	4.32	4.41	4.34	4.11	3.72	4.05	0.06	Nvidia GTX Titan X(Caffe)
	CRL	2.48	3.59	2.67	2.32	3.12	2.45	0.47	Nvidia GTX 1080
	iResNet	2.25	3.40	2.44	2.07	2.76	2.19	0.12	Nvidia Titan X(Caffe)
	Edge Stereo	1.84	3.30	2.08	1.69	2.94	1.89	0.32	Nvidia GTX 1080Ti (Caffe)
	Seg Stereo	1.88	4.07	2.25	1.76	3.70	2.08	0.6	Caffe
	GC-Net	2.21	6.16	2.87	2.02	5.58	2.61	0.90	Nvidia GTX Titan X
	PSMNet	1.86	4.62	2.32	1.71	4.31	2.14	0.41	Nvidia Titan Xp(CUDA)
	GWC-Net	1.74	3.93	2.11	1.61	3.49	1.92	0.32	Nvidia Titan Xp (–)
	HSM	1.80	3.85	2.14	1.63	3.40	1.92	0.14	Titan X Pascal
	GA-Net-15	1.48	3.46	1.81	1.55	3.82	1.93	1.8	GPU(Pytorch)
	SCV-Net	2.22	4.53	2.61	2.04	4.28	2.41	0.36	Nvidia GTX 1080Ti
	LEA Stereo	1.40	2.91	1.65	1.29	2.65	1.51	0.30	GPU @ 2.5 Ghz (Python)
	LaC+GANet	1.44	2.83	1.67	1.26	2.64	1.49	1.8	GPU @ 2.5 Ghz (Python)
	ACVNet	1.37	3.07	1.65	1.26	2.84	1.52	0.2	NVIDIA RTX 3090 (PyTorch)

Table 4. Middlebury 2014 Algorithm Comparison

Method	Family	Res	Bad2.0	Avgerr	Time
SGM-Forest	Non-End-to-End	*H*	7.37	2.84	88.5
MC-CNN-acrt	Non-End-to-End	*H*	8.08	3.82	150
MC-CNN-fst	Non-End-to-End	*H*	9.47	4.37	1.69
SGM	Traditional	*H*	18.4	5.32	9.90
EdgeStereo	End-to-End	*F*	18.7	2.68	0.35
iResNet	End-to-End	*H*	22.9	3.31	0.34

5. **Conclusion.** This paper summarizes the work related to deep learning-based stereo matching, classifying and summarizing from traditional matching methods to learning-based network methods, and comparing deep learning-based net stereo matching network methods by each test benchmark has outperformed traditional matching methods in terms of accuracy. The non-end-to-end network structure of the deep learning-based stereo matching method is simple and generalizable, but does not combine contextual information. The end-to-end network based approach combines contextual information network is more complex and more accurate, but consumes more computational resources and is more expensive. Therefore, the deep learning based matching method needs to be coordinated in terms of accuracy and running time to ensure the accuracy while minimizing the time, and because of the need for a large number of training images to support the portability and generalization capability is weak, the problems arising from inherently pathological regions areas such as weak texture, repeated texture and other regions are still a major hurdle to overcome. Therefore, it is necessary to design a robust, high accuracy, fast, cross-domain portable stereo matching method.

REFERENCES.

[1] H. Lee and Y. Shin, "Real-time Stereo Matching Network with High Accuracy," *IEEE International Conference on Image Processing(ICIP)*, pp. 4280-4284, 2019.

[2] S. Ji, C. Luo, and J Liu, "A Review of Deep Learning-Based Methods For Dense Matching of Stereo Images," *Journal of Wuhan University*, vol. 46, no. 2, pp. 1671-8860, 2021.

[3] N. Wang, X. Hu, F. Zhu, and J. Tang, "Single-view 3D Reconstruction Algorithm Based on View-aware," *Journal of Electronics & Information Technology*, vol. 42, no. 12, pp. 3053-3060, 2020.

[4] L. Shen, R. Zhang, Y. Zhu, and Y. Wu, "High-precision and Real-time Localization Algorithm for Automatic Driving Vehicles," *Journal of Electronics & Information Technology*, vol. 42, no. 1, pp. 28-35, 2020.

[5] S.-S. Shivakumar, T. Nguyen, I.-D. Miller, S.-W. Chen, V. Kumar, and C. Taylor, "DFuseNet: Deep Fusion of RGB and Sparse Depth Information for Image Guided Dense Depth Completion," *IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 13-20, 2019.

[6] K. Zhou, X. Meng, and B. Cheng, "Review of Stereo Matching Algorithms Based on Deep Learning," *Computational Intelligence and Neuroscience*, Avaiable: https://doi.org/10.1155/2020/8562323, 2020

[7] C. Wang, T. Shi, L.-L. Tang, and Y.-C. Chen, "Design of Neural Network Model for Lightweight 3D Point Cloud Classification," *Journal of Network Intelligence* , vol. 5, no. 3, pp. 122-128, 2020.

[8] J. Pang, W. Sun, J.-S. Ren, and Q. Yan, "Cascade Residual Learning:A Two-Stage Convolutional Neural Net-work for Stereo Matching," *IEEE International Conference on Computer Vision Workshops*, pp. 878-886, 2017.

[9] C. Wang, X. Wang, J. Zhang, L. Zhang, X. Bai, X. Ning, J. Zhou, and E. Hancock, "Uncertainty Estimation for Stereo Matching Based on Evidential Deep Learning," *Pattern Recognition*, vol. 124, pp. 108498, 2022.

[10] Z.-F. Wang, and Z.-G. Zheng, "2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1-8, 2008.

[11] K. Zhang, J. Lu, and G. Lafruit, "Cross-based Local Stereo Matching Using Orthogonal Integral Images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 1073-1079, 2009.

[12] Y. Chen, L.-L. Yang, Z.-P. Wang, "Literature Survey on Stereo Vision Matching Algorithms," *Journal of Graphics*, vol. 41, no. 5, pp. 702-708, 2020.

[13] K. Ambrosh, "Accurate Hardware-based Stereo Vision Compute," *Computer Vision and Image Understanding*, vol. 114, pp. 1303-1316, 2010.

[14] R. Zabin, and J. Woodfill, "Non-parametric Local Transforms for Computing Visual Correspondence," *Computer Vision-ECCV'94: Third European Conference on Computer Vision, Stockholm, Sweden, May 2-6, 1994. Proceedings. Springer Science & Business Media* , vol. 2, p. 151, 1994.

[15] H. Hirschmuller, "Stereo Processing by Semiglobal Matching and Mutual Information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 328-341, 2008.

[16] F. Zhang, T.-Y. Wu, Y. Wang, R. Xiong, G. Ding, P. Mei, and L. Liu, "Application of Quantum Genetic Optimization of LVQ Neural Network in Smart City Traffic Network Prediction," *IEEE Access*, vol. 8, pp. 104555-104564, 2020.

[17] M.-E. Wu, J.-H. Syu, and C.-M. Chen, "Kelly-based Options Trading Strategies on Settlement Date Via Supervised Learning Algorithms," *Computational Economics*, vol. 59, no. 4, pp. 1627-1644, 2022.

[18] Y.-P. Feng, and Z.-M. Lu, "A New Bone Direction Prediction Method Based on Spatial-temporal Graph Convolutional Network," *Journal of Network Intelligence*, vol. 7, no. 4, pp. 835-847, 2022.

[19] F. Zhang, T.-Y. Wu, J.-S. Pan, G. Ding, and Z. Li, "Human Motion Recognition Based on SVM in VR Art Media Interaction Environment," *Human-centric Computing and Information Sciences*, vol. 9, pp. 1-15, 2019.

[20] C.-Y. Yin, H.-H. Zhi, and H.-B. Li, "Survey of Binocular Stereo-matching Methods Based on Deep Learning," *Computer Engineering*, vol. 48, no.10, pp. 1-12, 2022.

[21] F. Zhang, T.-Y. Wu, and G. Zheng, "Video Salient Region Detection Model Based on Wavelet Transform and Feature Comparison," *EURASIP Journal on Image and Video Processing*, Avaiable: https://doi.org/10.1186/s13640-019-0455-2, 2019.

[22] K. Wang, F. Li, C.-M. Chen, M.-M. Hassan, J. Long, and N. Kumar, "Interpreting Adversarial Examples and Robustness for Deep Learning-based Auto-driving Systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 7, pp. 9755-9764, 2021.

[23] Y. Ma, Y. Peng, and T.-Y. Wu, "Transfer Learning Model for False Positive Reduction in Lymph Node Detection Via Sparse Coding and Deep Learning," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2121-2133, 2022.

[24] J. Zbontar, and Y. LeCun, "Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches," *Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2287-2318, 2016.

[25] A. Seki, and M. Polleyeys, "SGM-Nets:Semi-global Matching with Neural Networks," *Conference on Computer Vision and Pattern Recognition*, pp. 6640-6649, 2017.

[26] Z. Yin, T. Darrell, and F. Yu, "Hierarchical Discrete Distribution Decomposition for Match Density Estimation," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6044-6053, 2019.

[27] J.-P. Yao, K.-Y. Cheng, S.-X. Fan, and X.-J. Li, "A Joint Aspect-based Sentiment Analysis Method Based on the Encoder-decoder Architecture," *Journal of Network Intelligence*, vol. 6, no. 2, pp. 276-288, 2021.

[28] N. Mayer, E. Ilg, P. Hausser, P. Fischer, and D. Cremers, "A Large Dataset to Train Convolutional Networks for Disparity, Optical Flflow, and Scene Flflow Estimation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4040-4048, 2016.

[29] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry, "End-to-end Learning of Geometry and Context for Deep Stereo Regression," *IEEE International Conference on Computer Vision*, pp. 66-75, 2017.

[30] J.-F. Lu, J.-B. Ni, L. Li, T. Luo, and C.-C. Chang, "A Coverless Information Hiding Method Based on Constructing a Complete Grouped Basis with Unsupervised Learning," *Journal of Network Intelligence*, vol. 6, no. 1, pp. 29-39, 2021.

[31] C. Godard, O.-M. Aodha, and G.-J. Brostow, "Unsupervised Monocular Depth Estimation with Left-right Consistency," *The 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6602-6611, 2017.

[32] C. Zhou, H. Zhang, X. Shen, and J. Jia, "Unsupervised Learning of Stereo Matching," *IEEE International Conference on Computer Vision*, pp. 1567-1575, 2017.

[33] X. Mei, X. Sun, M. Zhou, S. Jiao, H. Wang, and X. Zhang, "On Building an Accurate Stereo Matching System on Graphics Hardware," *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 467-474, 2011.

[34] Z. Chen, X. Sun, L. Wang, Y. Yu, and C. Huang, "A Deep Visual Correspondence Embedding Model for Stereo Matching Costs," *IEEE International Conference on Computer Vision*, pp. 972-980, 2015.

[35] W. Luo, A.-G. Schwing, and R. Urtasun, "Efficient Deep Learning for Stereo Matching," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5695-5703, 2016.

[36] S. Zagoruyko, and N. Komodakis, "Learning to Compare Image Patches Via Convolutional Neural Networks," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4353-4361, 2015.

[37] J. Chen, and C. Yuan, "Convolutional Neural Network Using Multi-scale Information for Stereo Matching Cost Computation," *2016 IEEE International Conference on Image Processing (ICIP)*, pp. 3424-3428, 2016.

[38] H. Park and K. M. Lee, "Look Wider to Match Image Patches with Convolutional Neural Networks," *IEEE Signal Processing Letters*, vol. 24, no. 12, pp. 1788-1792, 2017.

[39] A. Spyropoulos, N. Komodakis, and P. Mordohai, "Learning to Detect Ground Control Points for Improving the Accuracy of Stereo Matching," *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1621-1628, 2014.

[40] A. Spyropoulos, and P. Mordohai, "Correctness Prediction, Accuracy Improvement and Generalization of Stereo Matching Using Supervised Learning," *International Journal of Computer Vision volume*, vol. 118, no. 3, pp. 300-318, 2016.

[41] N. Komodakis, G. Tziritas, and N. Paragios, "Fast, Approximately Optimal Solutions for Single and Dynamic MRFs," *2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis*, pp. 1-8, 2007.

[42] J.-L. Schonberger, S.-N. Sinha, and M. Pollefeys, "Learning to Fuse Proposals from Multiple Scanline Optimizations in Semi-global Matching," *European Conference on Computer Vision (ECCV)*, pp. 739-755, 2018.

[43] Z. Jie, P. Wang, Y. Ling, B. Zhao, Y. Wei, J. Feng, and W. Liu, "Left-right Comparative Recurrent Model for Stereo Matching," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3838-3846, 2018.

[44] K. Batsos, and P. Mordohai, "RecResNet:A Recurrent Residual CNN Architecture for Disparity Map Enhancement," *2018 International Conference on 3D Vision (3DV). IEEE*, pp. 238-247, 2018.

[45] S. Gidaris, and N. Komodakis, "Detect, Replace, Refifine: Deep Structured Prediction for Pixel Wise Labeling," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7187-7196, 2017.

[46] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hauuser, C. Hazirbas, V. Golkov, P.-V.-D. Smagt, D. Cremers, and T. Brox, "FlowNet:Learning Optical Flow with Convolutional Networks," *IEEE International Conference on Computer Vision.*, pp. 2758-2766, 2015.

[47] J. Pang, W. Sun, J.-S.-J. Ren, C. Yang, and Q. Yan, "Cascade Residual Learning: A Two-stage Convolutional Neural Network for Stereo Matching," *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*, pp. 887-895, 2017.

[48] Z.-F. Liang, Y.-L. Feng, Y.-L. Guo, S.-P. Huang, and J.-H. Lai, "Deep Learning for Real-Time Image Enhancement: From Smartphone to DSLR Cameras," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2811-2820, 2018.

[49] X. Song, X. Zhao, H. Hu, and L. Fang, "EdgeStereo:A Context Integrated Residual Pyramid Network for Stereo Matching ," *Computer Vision-ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2-6, 2018, Revised Selected Papers, Part V 14. Springer International Publishing* , pp. 20-35, 2019.

[50] G. Yang, H. Zhao, J. Shi, Z. Deng, and J. Jia, "Segstereo: Exploiting Semantic Information for Disparity Estimation," *European Conference on Computer Vision (ECCV)*, pp. 636-651, 2018.

[51] J.-R. Chang, and Y.-S. Chen, "Pyramid Stereo Matching Network," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5410-5418, 2018.

[52] G. Yang, J. Manela, M. Happold, and D. Ramanan, "Hierarchical Deep Stereo Matching on High-resolution Images," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5515-5524, 2019.

[53] F. Zhang, V. Prisacariu, R. Yang, and P.-H. Torr, "GA-Net: Guided Aggregation Net for End-to-end Stereo Matching," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 185-194, 2019.

[54] C. Lu, H. Uchiyama, D. Thomas, A. Shimada, and R.-I. Taniguchi, "Sparse Cost Volume for Efficient Stereo Matching," *Remote Sensing*, vol. 10, no. 11, pp. 1844, 2018.

[55] Y. Wang, Z. Lai, G. Huang, B.-H. Wang, L. Van Der Maaten, M. Campbell, and K.-Q. Weinberger, "Anytime Stereo Image Depth Estimation on Mobile Devices," *2019 International Conference on Robotics and Automation (ICRA)*, pp. 5893-5900, 2019.

[56] H. Jang, S. Kim, and S. Kim, "High-Resolution Stereo Matching Using Multiple Deep Networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 5042-5055, 2020.

[57] W. Lv, W. Lv, J. Li , "High-resolution Satellite Multi-class Cloud Detection Based on Improved AlexNet," *Journal of Network Intelligence*, vol. 6, no. 2, pp. 189-205, 2021.

[58] H.-X. Du, H.-B. Ma, and Z. Fan, "High-resolution Human Pose Estimation Method Based on Efficient Convolution," *Journal of Network Intelligence*, vol. 7, no. 4, pp. 909-920, 2022.

[59] B. Liu, H. Yu, and Y. Long, "Local Similarity Pattern and Cost Self-reassembling for Deep Stereo Matching Networks," *AAAI Conference on Artificial Intelligence*, vol. 36, no. 2, pp. 1647-1655, 2022.

[60] G. Xu, J. Cheng, P. Guo, and X. Yang, "Attention Concatenation Volume for Accurate and Efficient Stereo Matching," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12981-12990, 2022.

[61] X. Cheng, Y. Zhong, M. Harandi, Y. Dai, X. Chang, H. Li, and Z. Ge, "Hierarchical Neural Architecture Search for Deep Stereo Matching," *Advances in Neural Information Processing Systems*, vol. 33, pp. 22158-22169, 2020.

[62] M.-S. Hamid, N.-F. Abd Manap, R.-A. Hamzah, and A.-F. Kadmin, "Stereo Matching Algorithm Based on Deep Learning: A survey," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 5, pp. 1663-1673, 2022.

[63] M. Poggi, F. Tosi, K. Batsos, P. Mordohai, and S. Mattoccia, "On the Synergies Between Machine Learning and Binocular Stereo for Depth Estimation from Images: A Survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 9, pp. 5314-5334, 2021.

[64] D. Scharstein, and R. Szeliski, "A Taxonomy and Evaluation of Dense Two-frame Stereo Correspondence Algorithms," *International Journal of Computer Vision*, vol. 47, no. 1, pp. 7-42, 2002.