

ARTICLE INFO

Keywords:

Self-Supervised Depth Estimation

Stereo Depth Estimation

Stereo Matching

Cross-Attention

ABSTRACT

Depth Estimation plays a crucial role in recent applications in robotics, autonomous vehicles, and augmented reality. These scenarios commonly operate under constraints imposed by computational power. Stereo image pairs offer an effective solution for depth estimation since it only needs to estimate the disparity of pixels in image pairs to determine the depth in a known rectified system. Due to the difficulty in acquiring reliable ground-truth depth data across diverse scenarios, self-supervised techniques emerge as a solution, particularly when large unlabeled datasets are available. We propose a novel self-supervised convolutional approach that outperforms existing state-of-the-art Convolutional Neural Networks (CNNs) and Vision Transformers (ViTs) while balancing computational cost. The proposed CCNeXt architecture employs a modern CNN feature extractor with a novel windowed epipolar cross-attention module in the encoder, complemented by a comprehensive redesign of the depth estimation decoder. Our experiments demonstrate that CCNeXt achieves competitive metrics on the KITTI Eigen Split test data while being 10.18× faster than the current best model and achieves state-of-the-art results in all metrics in the KITTI Eigen Split Improved Ground Truth and Driving Stereo datasets when compared to recently proposed techniques. To ensure complete reproducibility, our project is accessible at <https://github.com/alelopes/CCNeXt>.

1. Introduction

Depth information is critical for many computer vision and image analysis applications. It has been applied to tasks such as synthetic object insertion in computer graphics (Luo et al., 2020), robot-assisted surgery (Stoyanov et al., 2010), autonomous driving (Levinson et al., 2011), and smart glasses (Wang et al., 2023). These applications typically encounter constraints related to computing resources.

Depth can be estimated using a monocular camera (Bhat et al., 2023), stereo camera (Fang et al., 2023), or a multi-camera system (Wei et al., 2023). Monocular depth estimation techniques involve learning meter or relative depth information that can be extracted from individual images, a process that necessitates supervised training (Guizilini et al., 2020) or estimating additional parameters such as camera parameters for self-supervised techniques using monocular videos (Guizilini et al., 2022; Godard et al., 2019). In contrast, stereo systems can solve depth estimation by calculating the disparity of pixels in a pair of cameras. For rectified images, knowing the intrinsic and extrinsic camera parameters is sufficient to transform depth estimation into a stereo matching problem. Self-supervised methods can also be applied to stereo systems, where the primary objective is to establish correspondences between left and right images without relying on ground-truth data. Since the camera system is known, the main advantage of a stereo system is that it obviates the necessity of estimating the inter-frame camera poses. This is particularly advantageous in complex scenarios involving object motion (Godard et al., 2019).

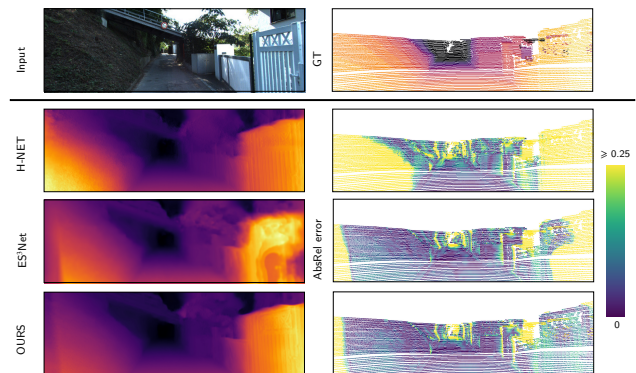


Figure 1: Above the line: Input image and depth map ground truth (GT). Below the line: Depth estimations from H-Net (Huang et al., 2022), ES³Net (Fang et al., 2023) and our method on the left. The AbsRel metric error is shown on a map on the right for all estimations when compared to the ground truth. For the AbsRel error map, the darker/bluer the point is, the lower the error.

CNNs have been successfully applied to depth estimation and stereo matching (Wang et al., 2021; Huang et al., 2022; Godard et al., 2017) tasks. Nevertheless, there has been a notable shift towards the adoption of ViT-based architectures in Computer Vision. Although some works seek to provide explanations for differences between ViTs and CNNs or the effectiveness of ViT techniques (Park and Kim, 2022; Xie et al., 2023), many ViTs proposals compare their architectures with basic CNNs baselines, such as the ResNet, as recently reported in (Liu et al., 2022; Smith et al., 2023). Furthermore, certain ViT studies fail to include the runtime of the proposed methods when comparing them to

*Corresponding author

✉ a115968@dac.unicamp.br (A. Lopes)

ORCID(s): 0000-0002-3873-2835 (A. Lopes); 0000-0001-7824-5217 (R. Souza); 0000-0003-0125-630X (H. Pedrini)

CNN strategies and fail to address the trade-off between metric performance and runtime.

This trade-off is particularly evident in the field of self-supervised stereo depth, where prior methods typically face two fundamental limitations. First, *computational burden*: Transformer-based approaches that attend across entire epipolar lines (e.g., ChiTransformer (Su and Ji, 2022)) achieve state-of-the-art metrics but at the expense of runtime and memory, restricting their use in real-time or embedded applications. Second, *metric performance*: lightweight CNN-based designs (e.g., Monodepth2 (Godard et al., 2019) and ES³Net (Fang et al., 2023)) offer good latency but tend to underperform in accuracy, especially in low-texture or occluded regions. Thus, the field continues to face the challenge of achieving both efficiency and competitive accuracy.

To address the need to achieve state-of-the-art metrics while not increasing the model execution time, we introduce the **Cross-attentional Convolutional ConvNeXt (CCNeXt)** architecture, a novel CNN-based architecture for self-supervised stereo depth estimation. CCNeXt is explicitly designed to balance metric performance and computational efficiency, surpassing both recent CNN and ViT architectures on KITTI and DrivingStereo (Figure 1) while running significantly faster than the state-of-the-art ChiTransformer (Su and Ji, 2022).

For the encoder, we changed the standard ResNet-based encoder (Godard et al., 2019; Huang et al., 2022) to the smaller version of ConvNeXt (Liu et al., 2022) feature extractor with the addition of a bottleneck block in the first convolutional block (stem cell), allowing a better feature representation in the larger skip connection dimension. Our model is trained using a shared-weight network between left-right image pairs, coupled with a novel windowed epipolar cross-attention mechanism, which allows the flow of feature representations between right-left feature pairs. We remodel the dual view training for the decoder to reduce execution time and propose the Individual Contextual Expansive Path (ICEP) to enhance the higher dimension outputs. Our model achieves state-of-the-art results in the KITTI dataset (Geiger et al., 2012) for Absolute Relative difference (AbsRel), Squared Relative difference (SqRel), and accuracy with a maximum relative error of 25% ($\delta < 1.25$) depth metrics compared to the state-of-the-art method ChiTransformer while being more than ten times faster, and achieves state-of-the-art results for all depth metrics compared to monocular, monocular-stereo, and stereo self-supervised techniques on the KITTI improved Ground Truth (IGT) dataset (Uhrig et al., 2017), and DrivingStereo datasets (Yang et al., 2019). In addition, we conduct a statistical analysis of the results, extending the customary practice of only comparing the average metric results of different models.

In summary, unlike prior self-supervised stereo methods that either sacrifice runtime for accuracy or vice versa, CCNeXt introduces a modern ConvNeXt backbone, windowed epipolar cross-attention, and a lightweight decoder that together achieve state-of-the-art accuracy while reducing execution time by an order of magnitude.

The main contributions of our work are:

- We introduce a novel encoder-decoder architecture based on ConvNeXt for self-supervised stereo depth estimation, enhancing feature extraction capabilities through the incorporation of a windowed epipolar cross-attention and a composed stem cell for the encoder, and implemented a novel strategy called ICEP for the decoder. This approach enhances low and high-resolution output quality and metrics, as demonstrated by ablation studies.
- Our approach achieves state-of-the-art results for three out of seven depth estimation metrics on KITTI while being 10.18× faster than the current best model and achieves state-of-the-art results for all metrics on the KITTI IGT and DrivingStereo datasets.
- We performed a statistical analysis to compare the outcomes of our model with the literature results. We also compared published solutions with available training procedures on the DrivingStereo dataset, enabling us to discern the strengths and limitations of the KITTI dataset for depth estimation.

2. Related Work

Depth estimation methods can be classified based on the number of images used (monocular, stereo, or multi-view) and the type of supervision (supervised or self-supervised). While these approaches share a common goal, they often achieve remarkable results under specific configurations, such as supervised monocular or self-supervised stereo depth estimation.

This distinction is particularly clear in monocular methods, where supervised and self-supervised techniques diverge significantly. For instance, Fu *et al.* (Fu et al., 2018) proposed a method using ordinal regression to estimate continuous depth through linear combinations of outputs, which was further refined with adaptive and iterative combinations (Shao et al., 2024; Li et al., 2024). In contrast, self-supervised techniques typically lack similarities in loss functions, architectures, or strategies with supervised methods. These techniques often estimate depth and camera motion between consecutive frames. Zhou *et al.* (Zhou et al., 2017) proposed a pipeline that predicts depth and pose using dedicated networks for each task. Subsequent improvements included multi-scale sampling and auto-masking loss to improve visual artifacts and object motion (Godard et al., 2019) and scale-aware training for better generalization (Guizilini et al., 2020). Recently, Guizilini *et al.* (Guizilini et al., 2022) proposed a Transformer-based approach that uses a cross-attention matching mechanism instead of predicting camera pose for the input frames.

Supervised and self-supervised stereo vision approaches also differ in loss, architecture, and training strategies. Supervised stereo depth and stereo matching techniques commonly employ a cost volume based on CNN-extracted

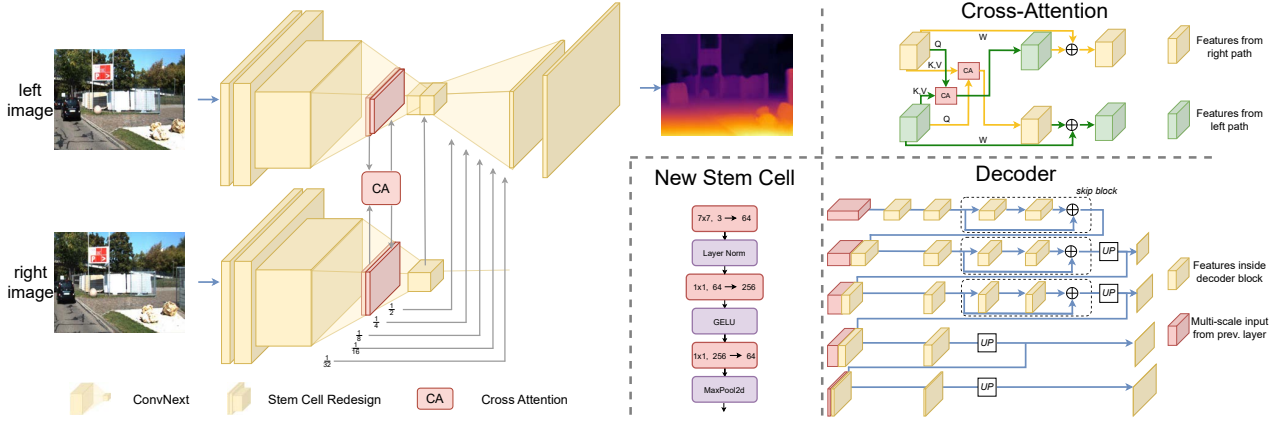


Figure 2: Overview of our framework. We proposed a shared-weight architecture with skip connections and windowed epipolar cross-attention on the Encoder (Left). Details on cross-attention (Top right), Decoder architectures (Bottom right), and new stem cell proposal (Center bottom) are also displayed.

features from left and right images to create a matching prediction (Xu et al., 2023a; Chang and Chen, 2018; Duggal et al., 2019; Lipson et al., 2021; Xu et al., 2022; Kusu-pati et al., 2020). These are regularized to produce better representations since this simple cost volume is noise-contaminated (Laga et al., 2020). There are many techniques to regularize, where some works use Convolutional Gated Recurrent Units (ConvGRUs) to update the disparity maps using local cost values (Xu et al., 2023a; Lipson et al., 2021), or using 2D convolutions (Duggal et al., 2019). Transformer-based architectures were also proposed (Guo et al., 2022; Xu et al., 2023b; Li et al., 2021b), where global context information is extracted using a combination of cross-attention and axial attention (Guo et al., 2022), and a unified model was proposed for rectified stereo matching, unrectified stereo depth estimation, and optical flow (Xu et al., 2023b). Therefore, these architectures are typically designed to enhance fine-grained prediction details under a supervised loss function, often by incorporating complex decoders with various cost volume compositions. However, as we demonstrate in Section 5.1, they tend to yield poor representations in self-supervised settings.

For self-supervised stereo methods, Godard et al. (2017) introduced left-right consistency, generating depth maps for both left and right images using a combination of consistency loss, disparity smoothness loss, and photometric reprojection error loss. Besides being applied for monocular depth estimation, Monodepth2 (Godard et al., 2019) also presented setups for self-supervised stereo depth estimation and a monocular-stereo setup, mixing left-right information with video frames.

More recently, H-Net (Huang et al., 2022) has proposed finding epipolar features by matching candidates using optimal transport as part of an attention mechanism during the encoder and decoder stages, allowing features from the left and right paths to be shared in a siamese encoder-decoder network. Su and Ji (2022) proposed Chi-Transformer, a Transformer-based approach that applies

multiple self-attention and cross-attention layers for left-view and right-view features extracted using a CNN and fuses information back in a decoder using convolutions. ChiTransformer presents different strategies, including a gated positional cross-attention mechanism that limits the attentional area to match the epipolar geometry. Recently, ES³Net (Fang et al., 2023) used the RealtimeStereo (Chang et al., 2020) as a backbone architecture, adapting it to a self-supervised training using two left-right flipped inputs to predict left and right disparity maps. Together, these approaches illustrate the broader trade-off discussed in the Introduction: CNN-based designs are generally efficient but lag behind in accuracy, whereas Transformer-based methods achieve strong performance at the expense of runtime and memory.

ChiTransformer currently holds state-of-the-art performance for self-supervised depth estimation across various metrics. However, its runtime is compromised by the presence of multiple attention modules. To overcome this bottleneck, we propose CCNeXt, which integrates a modern CNN backbone with a novel windowed epipolar cross-attention mechanism and a lightweight decoder. This design achieves state-of-the-art accuracy while significantly reducing execution time.

Beyond stereo depth, self-supervised and weakly supervised learning have progressed rapidly across domains beyond stereo depth. Examples include federated contrastive learning with feature distillation for human activity recognition (Xiao and Tong, 2025) and contrastive token and label activation for remote-sensing semantic segmentation (Hu et al., 2024). While these works share our overarching objective of learning strong representations with limited supervision, CCNeXt differs in two fundamental ways. First, our supervision signal is geometry-driven: stereo photometric reprojection under known intrinsics and baseline, in contrast to instance- or feature-level contrastive objectives and augmentation strategies without geometric grounding. Second, our architecture explicitly encodes epipolar awareness through

windowed cross-attention constrained by a physically valid disparity range, which is unique to stereo geometry. Thus, CCNeXt is complementary: methods such as contrastive pretraining or token selection could be integrated into our encoder, but our core novelty lies in geometry-constrained attention that simultaneously enhances efficiency and matching reliability.

3. Methodology

In this section, we describe our CCNeXt architecture, detailing our encoder composed of the backbone choice, stem cell substitution, windowed cross-attention mechanism, and our proposed decoder architecture with changed convolutional layer positions and skip blocks. The complete architecture is shown in Figure 2.

3.1. Encoder

ConvNeXt. We initially upgraded the ResNet encoder backbone architecture used by Monodepth2, ChiTransformer, and H-Net strategies. We chose ConvNeXt for its equivalent performance to Transformer-based architectures such as Swin Transformer (Liu et al., 2021). Besides using ResNet-based architectures, Monodepth2 and H-Net use a lighter ResNet18 version than the ResNet50 version adopted by ChiTransformer. Thus, we chose a ConvNeXt version that is slightly faster and has approximately 10% fewer parameters than ResNet18.

Stem Cell. For our ConvNeXt architecture, we propose the usage of additional convolutional layers, substituting the stem cell of the original ConvNeXt to extract deeper features. This allows our architecture to have more complex features before skip connecting this information to the decoder. The first skip connection is the closest one to the highest feature map dimensions. Therefore, it directly affects the quality of the final prediction. The proposed Stem Cell structure follows the structure of the Inverted Bottleneck that is applied in each block of the ConvNeXt (Liu et al., 2022). Details of the implementation, including convolutional layer parameters (kernel sizes, input and output channels) and the ordering of additional layers such as normalization and activation, are provided in Figure 2.

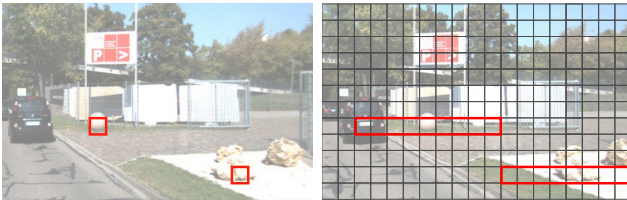


Figure 3: Windowed Epipolar Cross-Attention. We proposed a windowed cross-attention mechanism that limits the attention to valid epipolar candidates. Points of interest in an image (Left) can only be matched with a limited group of pixels w_{pl} on the other stereo image.

Windowed Cross-Attention. The proposed windowed cross-attention mechanism (Figure 3) is inspired by Huang

et al. (2022) in its EG-MEA approach of H-Net, MVSTER (Wang et al., 2022b), and Swin Transformer (Liu et al., 2021). H-Net is inspired by Wang et al. (2018) self-attention, which adapts the Transformer-based mechanism to CNNs using 1×1 convolutions instead of using the projection matrices for key, query and value. The principal particularity of our proposed cross-attention mechanism is that it is a windowed epipolar attentional mechanism; therefore, only features within a limited range along the same row of the left-view X_l and right-view X_r features are evaluated as matching candidates for Key and Query components. This restriction greatly reduces the likelihood of false positives between visually similar structures that cannot correspond geometrically. For instance, this is particularly beneficial when multiple traffic lights appear at the same height but on opposite sides of a road.

To further constrain the attention, we exploit the stereo geometry to define a maximum search range. Given a rectified stereo pair $I^l, I^r \in \mathbb{R}^{H \times W}$ with focal length f_x and baseline B , the largest disparity at encoder scale s (downsampling factor 2^s) can be estimated from a minimum plausible scene depth $D_{\min}$ as in Equation 1.

$$\Delta_{\max}(s) = \frac{Bf_x}{D_{\min} 2^s}. \quad (1)$$

Instead of attending across the entire image row, each pixel only considers candidates within $\pm \Delta_{\max}(s)$ around its location. This yields a window of width $w_s = 2\Delta_{\max}(s) + 1$, which is typically much smaller than the full row. As a result, the attention cost per query is reduced from $O(W/2^s)$ in full-row attention to $O(w_s)$ in our formulation. This windowing not only lowers the computational complexity but also prevents spurious correspondences at geometrically implausible disparities (e.g., multiple traffic lights aligned at the same height but far apart in the scene).

In Equation 2, we define the cross-attentional mechanism, where $Q_r = q(X_r)$, $K_l = k(X_l)$, and $V_l = v(X_l)$ are defined in $Q_r \in \mathbb{R}^{c_{in} \times w \times h}$, $K_l \in \mathbb{R}^{c_{in} \times w \times h}$, and $V_l \in \mathbb{R}^{c_{in} \times w \times h}$. The q , k , and v are 1×1 convolutions that map input of dimension $(c \times h \times w)$ to $(c_{in} \times h \times w)$.

$$\text{Att}_l(Q_r, K_l, V_l) = \text{softmax}(Q_r K_l^T) V_l \quad (2)$$

The Att_l softmax term maps every $w^r \times w_{pl}^l$, making it possible to find the correspondent w_i^r position for each w_i^l element in each windowed row of the feature map. Subsequently, we apply an additional convolutional layer W_l as the last linear layer of the original self-attention mechanism. This output is combined with the right-view input features X_r as available in Equation 3.

$$Y_r = W_l(\text{Att}_l) + X_r \quad (3)$$

We did not use multi-head attention due to its computational cost increase, so our attention can be interpreted as the multi-head attention layer proposed by Vaswani et al. (2017) with the number of heads equal to one. This architecture is

also mirrored for the attentional outputs of Y_l , where now we have Q_l , K_r , and V_r , instead of Q_r , K_l , and V_l .

3.2. Decoder

For the decoder, we introduced the ICEP architecture (Figure 2) with added convolutional blocks with skip connections instead of having cross-attention modules as performed in H-Net. This modification enables our architecture to predict a single-view disparity estimation using a pair of inputs, as there is no interdependence between the right and left depth predictions in the decoder. Consequently, this design choice results in a significant reduction of execution time and memory usage, nearly halving both. Similarly to H-Net, before starting the decoder stage, we concatenate features from both left and right paths to form a $(2C \times H \times W)$ volume that is then reduced to a $(C \times H \times W)$ feature map by applying a 3×3 convolutional layer. This volume is similar to the cost volume used in the stereo matching approaches. However, we do not concatenate different dimension features (D_n) to form a single $(2CD_n \times H \times W)$ feature map. Consequently, we form D_n features of $(C_i \times H_i \times W_i)$, where i varies depending on the input feature map size.

Intra Skip Blocks. We propose the addition of convolutional blocks, each with their own skip connections, within the decoder path (Figure 2). This strategy is intended to enhance the representation, preserve the backpropagation of gradient in low-resolution features, and break the left-right decoders' dependency during inference, which reduces the execution time of the model. The skip blocks apply two convolutions with the same channels as the previous layers in each i -th layer. We did not use the intra skip blocks in the higher resolution layers of the decoder since it would increase the computational cost of our decoder. In the lower resolution blocks, this increment is derisory.

ICEP. The final ICEP decoder modifies the Monodepth-style architecture (Godard et al., 2017) with three key changes: (i) a symmetric high-resolution head that prevents the highest-scale features from supervising multiple output scales simultaneously; (ii) delayed fusion of the top-level skip (x_0), so that y_0 alone benefits from the finest encoder details, improving boundary sharpness; (iii) intra-decoder skip blocks (two 3×3 convolutions with residual connection) at lower-resolution stages, which preserve gradient flow and suppress phantom responses in textureless regions. These design changes allow for more accurate depth prediction across all scales.

In Monodepth, skip connection concatenations start from the beginning of the decoder, meaning the last two blocks share almost the same features except for an upsampling and extra convolutions. As a result, high-resolution features (x_0) influence both y_1 and y_0 , which disturbs fine-grained detail learning. In ICEP, we postpone the x_0 fusion so that only y_0 uses the finest-scale features, producing consistent and more detailed outputs (see Figure 4).

For upsampling, we adopt nearest-neighbor interpolation followed by a 3×3 convolution to avoid checkerboard

artifacts, and concatenate encoder skips with decoder activations at each scale before the intra-skip block. This design preserves high-frequency details while keeping computation modest.

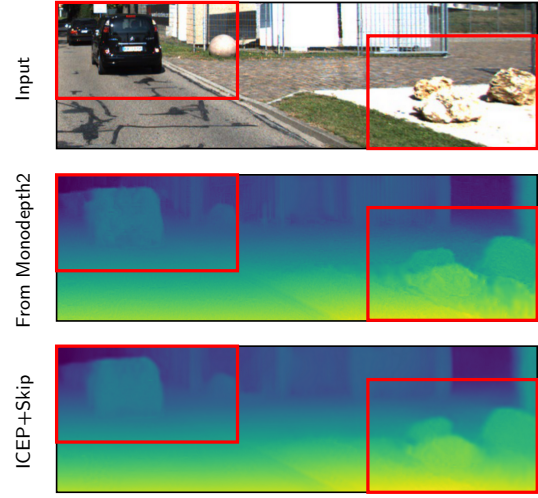


Figure 4: On **top**: Input image from the KITTI dataset. On **middle**: Disparity estimation using the same encoder and the decoder of Monodepth2. On **bottom**: Same encoder as in the middle with the ICEP decoder with Skip blocks.

3.3. Training Procedures

Both left and right views are trained, producing disp_r and disp_l outputs for each view. Similar to (Zhong et al., 2017; Huang et al., 2022; Su and Ji, 2022; Godard et al., 2017, 2019; Wang et al., 2021), we formulate the problem as a photometric reprojection error, adding an edge-aware smoothness loss, as proposed by Monodepth (Godard et al., 2017). Although we predict the disparities as the output of our network, we do not warp the disparity information directly on the input image I_l to generate the other view image I_j^{rep} . We first convert our disparity map into a dense depth map, and then we perform the camera projection from one view to the other using the intrinsic parameters and extrinsic parameters of the camera system. For the extrinsic parameters, we only add the translation caused by the distance of the cameras in the stereo system, which is the baseline distance. The same procedure is performed in most self-supervised depth estimation or stereo matching techniques (Wang et al., 2021; Huang et al., 2022; Godard et al., 2019; Su and Ji, 2022).

The photometric reprojection error ($\mathcal{L}_{\text{pe}}$) is formed by the combination of the SSIM metric (Wang et al., 2004) and the L1 loss, as available in Equation 4, where $\alpha = 0.85$. We also apply the auto-masking strategy proposed by Godard et al. (Godard et al., 2017), filtering out pixels where the change between left and right original views (I^l and I^r) is smaller than the difference of the predicted reprojection and

the original image (I^l and I^l_{pred}).

$$\mathcal{L}_{\text{pe}}(I^l, I^l_{\text{proj}}) = \frac{\alpha}{2} \left(1 - \text{SSIM}(I^l, I^l_{\text{proj}}) \right) + (1 - \alpha) \|I^l - I^l_{\text{proj}}\| \quad (4)$$

The edge-aware smoothness loss term is composed of the mean-normalized disparity $d^*_t = d_t / \bar{d}_t$, and is available in Equation 5.

$$\mathcal{L}_{\text{sm}}(I^{\text{proj}}, d^*) = \frac{1}{N} \sum_{i,j} \left| \partial_x d^*_{ij} \right| e^{-\|\partial_x I^{\text{proj}}_{ij}\|} + \left| \partial_y d^*_{ij} \right| e^{-\|\partial_y I^{\text{proj}}_{ij}\|} \quad (5)$$

The final loss function comprises the right and left smoothness and reprojection errors, as in Equation 6, where m is the total number of scales. Although having both depths, we did not use left-right disparity consistency loss since it did not improve results in any evaluated scenario. For the smoothness loss term, we adopt the same weighting factor $\gamma = 1 \times 10^{-3}$ as reported in Monodepth2. We also experimented with values an order of magnitude lower and higher, but observed no improvement across any evaluated scenario.

$$\mathcal{L}_{\text{total}} = \frac{1}{2m} \sum_{i=0}^{m-1} \mathcal{L}_{\text{pe}}(I^l, I^l_{\text{proj}}) + \gamma \mathcal{L}_{\text{sm}}(I^l_{\text{proj}}, d^*_l) + \mathcal{L}_{\text{pe}}(I^r, I^r_{\text{proj}}) + \gamma \mathcal{L}_{\text{sm}}(I^r_{\text{proj}}, d^*_r) \quad (6)$$

The relationship of depth estimation and stereo matching in a rectified pair of images is defined as $\text{depth} = B f_x / \text{disp}$, where B is the baseline distance between the cameras (0.54 meters for both used datasets) and f_x is the focal length of the camera system in the x direction.

Therefore, using a known camera system, we can define a minimum and maximum depth value and determine a minimum and maximum disparity value that our system will generate.

4. Experiments

This section describes the experiments we conducted in our work, emphasizing implementation details and datasets used. In addition to metric average comparison, we perform a statistical analysis of the experiments since the metric differences are shrinking, making it impractical to compare research papers' metrics solely on the examination of the average of prediction metrics. This may occur due to the extensive usage of a single dataset (KITTI (Geiger et al., 2013)), its inability to evaluate different weather and illumination conditions (Wang et al., 2021), and its low percentage of pixels in distances higher than 40 meters (Yang et al., 2019). Some experiments performed here did not include all the published papers, mainly because weights, models, and training procedures are not available for PFN (Pilzer et al., 2019), Li et al. (Li et al., 2021a), PVStereo (Wang

et al., 2021), and SsSMnet (Zhong et al., 2017). ChiTransformer (Su and Ji, 2022) provides both code and weights, but they only present them for a distinct architecture that downsizes the original network, thereby harming the performance metrics when utilizing this shared model. H-Net (Huang et al., 2022) training procedures and weights are also unavailable. However, the authors shared the model and weights over request, enabling it to be statistically compared. Nevertheless, the lack of training procedures does not allow it to be expanded to different datasets or dimensions.

For this reason, we made available the training code, model, and weights of our presented model, allowing the reproducibility of the results presented in this paper.

4.1. Implementation Details

Our model was implemented using PyTorch. The model is trained with an input resolution of 1280×384 pixels for the KITTI experiments and 288×640 pixels for the Driving-Stereo dataset. We performed experiments using batch size 8, 30 epochs, and using the Adam optimizer (Kingma and Ba, 2014) for both datasets. We set a learning rate of $1e-4$, and after 15 epochs, we decayed its value to $1e-5$. We trained our models with four scales, therefore $m = 4$ in Equation 6. We used the ConvNext 'Pico' version for the backbone with a pre-trained version on the ImageNet (Russakovsky et al., 2015) dataset. For a matter of comparison, this architecture presents 8.57M parameters, while the ResNet18 version used in H-Net, Monodepth2, and Monodepth has 11.17M parameters, and the ResNet50 used by ChiTransformer, PFN, and STTR (Li et al., 2021b) has 21.28M parameters. The complete CCNeXt architecture has 18.36M parameters. Our architecture incorporates three window-cross attention layers, inserted after the first three stages of the ConvNeXt encoder to enhance feature interaction across views.

4.2. KITTI Dataset

The KITTI dataset (Geiger et al., 2013) comprises 61 scenes with a total number of 42,406 image pairs available, usually called 'KITTI Raw'. Following the literature for monocular and stereo depth estimation (Su and Ji, 2022; Godard et al., 2017, 2019; Huang et al., 2022; Klingner et al., 2020), we use the procedures introduced by Eigen et al. (Eigen et al., 2014), typically called the Eigen Split. This proposal separates 697 images extracted from 29 scenes for evaluation and trains/validates the remaining 32 scenes, which results in 22,600 images for training and 888 for validation. For evaluation, we also used the Garg crop (Garg et al., 2016), which mainly removes uninteresting regions, e.g., sky regions. We also employ the KITTI Eigen Split on the Depth Completion dataset (Uhrig et al., 2017), commonly referred to as the KITTI Improved Ground Truth (IGT) dataset with the Garg crop for evaluation. This official dataset from KITTI includes a semi-dense depth ground truth generated by the authors, with outliers removed from the laser scan in the raw data. We utilized the IGT to evaluate the model, obtaining their ground truth data from the same 697 images as the Eigen Split, whenever available. This yields a testing set comprising 652 images.

Table 1

Results on KITTI Eigen Split and IGT. Comparative results with self-supervised methods for KITTI Eigen Split dataset and with different modalities methods for the IGT dataset. In the *Train* column: **M**: monocular self-supervised, **MS**: monocular-stereo self-supervised, and **S**: stereo self-supervised. All methods were trained using the Eigen Split, except for the ES³Net, which used the KITTI Raw. The methods are also classified in terms of input image sizes. Low Resolution (LR) indicates techniques trained using 640×192 , and Full Resolution (FR) ones were trained using dimensions larger than 800×240 . Best results are in **bold**. †Reproducibility codes and weights not available for the IGT. ‡Shared weights do not reproduce the results from the paper for the KITTI Eigen Split, and the paper did not present results for IGT.

	Method	Train	Input	lower is better				higher is better		
				AbsRel	SqRel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
KITTI Eigen Split	MonoDepth2 (Godard et al., 2019)	S	FR	0.107	0.849	4.764	0.201	0.874	0.953	0.977
	PFN (Pilzer et al., 2019) †	S	LR	0.102	0.802	4.657	0.196	0.882	0.953	0.977
	Li et al. (Li et al., 2021a) †	S	FR	0.096	0.744	4.500	0.188	0.890	0.960	0.979
	H-Net (Huang et al., 2022)	S	LR	0.076	0.607	4.025	0.166	0.918	0.966	0.982
	ChiTransformer (Su and Ji, 2022) †	S	FR	0.073	0.634	3.105	0.118	0.924	0.989	0.997
	ES ³ Net (Fang et al., 2023)	S	FR	0.074	0.660	4.042	0.169	0.920	0.962	0.980
	CCNeXt (ours)	S	FR	0.070	0.581	3.883	0.171	0.926	0.963	0.979
IGT (Uhrig et al., 2017)	PackNet-SfM (Guizilini et al., 2020)	M	FR	0.071	0.359	2.723	0.092	0.959	0.992	0.998
	DepthFormer (Guizilini et al., 2022)	M	FR	0.055	0.265	2.723	0.092	0.959	0.992	0.998
	iDisc (Piccinelli et al., 2023)	M	FR	0.059	0.216	2.573	0.089	0.965	0.995	0.999
	DIFFNet+SfM-TTR (Izquierdo and Civera, 2023)	M	FR	0.056	0.273	2.600	0.093	0.969	0.992	0.997
	MonoDepth2 (Godard et al., 2019)	MS	FR	0.080	0.466	3.681	0.127	0.926	0.985	0.995
	SQLdepth (Wang et al., 2024)	MS	FR	0.052	0.223	2.550	0.084	0.971	0.995	0.998
	MonoDepth2 (Godard et al., 2019)	S	LR	0.085	0.537	3.868	0.139	0.912	0.979	0.993
	H-Net (Huang et al., 2022)	S	LR	0.059	0.208	2.370	0.087	0.977	0.996	0.998
	ChiTransformer (Su and Ji, 2022) †	S	FR	0.072	0.271	2.768	0.103	0.950	0.993	0.998
	ES ³ Net (Fang et al., 2023)	S	FR	0.039	0.173	2.062	0.069	0.988	0.996	0.998
	CCNeXt (ours)	S	FR	0.031	0.090	1.671	0.055	0.992	0.998	0.999

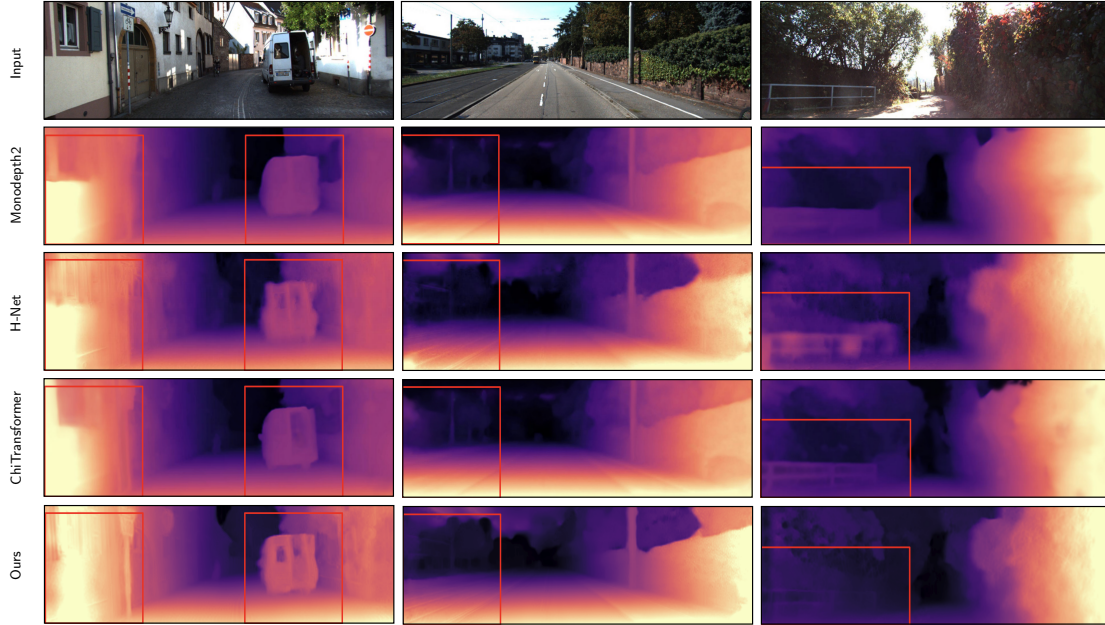


Figure 5: Qualitative results on the KITTI Eigen Split test set. Our model produces finer details, particularly for objects at close to mid-range distances. This can be seen in the left column, where the traffic sign, windows, and doors are clearly preserved; in the middle column, where the house, tree, and post are better delineated; and in the right column, where the iron rail is reconstructed without incorrectly filling its gaps, unlike competing methods. Comparison images from other approaches were obtained from the respective authors' papers.

All techniques except ES³NET used for comparison in our work were trained with the Eigen Split training set. In contrast, ES³NET was trained on the entire KITTI Raw dataset, which incorporates both the training and testing data from the Eigen Split.

4.3. DrivingStereo Dataset

We also use the DrivingStereo dataset (Yang et al., 2019), which contains 174,437 stereo pairs for training collected over 38 video sequences and 7,751 stereo pairs collected in 4 video sequences for testing. DrivingStereo

contains more diverse scenarios than KITTI, including urban, suburban, highway, elevated, and country roads, with sunny, rainy, cloudy, foggy, and dusky weather. In addition, it provides a 2,000-image subset specifically divided by weather conditions (sunny, cloudy, foggy, and rainy), which we use for further evaluation.

4.4. Estimating Depth Procedures

We followed the procedures proposed in Monodepth2 (Godard et al., 2019) for the conversions of depth and disparity for the KITTI dataset training. We convert the network’s sigmoid output σ into depth D by defining $D = 1/(a\sigma + b)$. Since $\{\sigma \in \mathbb{R} : 0 < \sigma < 1\}$, the a and b terms account for the conversion of the $(0, 1)$ range to a maximum and minimum disparity values based on a minimum and maximum depth of 0.1 and 100 units. Hence, since B and f_x are constants in the depth-disp conversion formula, the equivalence of both depth functions gives us that $\text{disp} = Bf_x(a\sigma + b)$. Therefore, the real disparity in pixels can be extracted by accounting for the predicted rescaled disparity extracted from $(a\sigma + b)$ multiplied by a Bf_x term.

During training, Monodepth2 also changed the translation in the x direction of the extrinsic parameter to 0.1 instead of 0.54. Therefore, we must multiply the final depth predictions by 5.4 to generate real depth results. The minimum and maximum depth predicted by our KITTI network is between 0.54 and 540 meters.

For the DrivingStereo dataset experiments, we also used the equation $\text{depth} = Bf_x/\text{disp}$, and since the baseline distance is the same as KITTI ($B = 5.4$), we have a maximum and minimum depth of 0.54 to 540 meters. We also performed experiments with lower intervals in the KITTI dataset, but they did not improve the results.

5. Results and Discussions

5.1. KITTI Results

Comparison with Depth Estimation Techniques: metric results for the Eigen Split on both KITTI and IGT datasets are available in Table 1, and a qualitative comparison is shown in Figure 5. For the IGT dataset, CCNeXt achieves state-of-the-art results in all metrics, surpassing all other reported monocular, monocular-stereo, and stereo self-supervised models. For the original Eigen Split, our approach achieves state-of-art results in the AbsRel, SqRel, and $\delta < 1.25$ metrics. ChiTransformer achieves state-of-the-art results for the other metrics, but it is important to highlight that this technique utilizes a ResNet-50 in the encoder combined with six Transformer-based self-attention mechanisms and six rectification blocks. Each rectification block contains two extra self-attention layers and one cross-attention layer. Thus, our approach is 10.18 $\times$ faster than ChiTransformer, and achieves state-of-the-art metrics in the KITTI dataset, as displayed in Table 2. To measure time, we ran each model 200 times using identical images from the testing set with batch size 1 and calculated the average and

Table 2

Runtime was measured for state-of-the-art self-supervised stereo techniques. The ChiTransformer architecture was reproduced based on the description provided in the original paper. Although we could not replicate the exact training procedure, the reproduced model based on their official repository is sufficient for a fair runtime estimation. All measurements were conducted with an input resolution of 1280 $\times$ 384 for CCNeXt and 1216 $\times$ 352 for ChiTransformer, using an Intel Xeon Gold 6148 CPU.

Method	ChiTransformer (Su and Ji, 2022)	CCNeXt (ours)
Runtime(s)	29.82 $\pm$ 1.00	2.93 $\pm$ 0.03

Table 3

Median Value Metric Comparison of Available Techniques on KITTI Eigen Split. $\dagger p < 0.05$ for Wilcoxon test using Bonferroni correction.

Method	AbsRel	SqRel	RMSE	$\delta < 1.25$
H-Net (Huang et al., 2022)	0.072	0.489	3.830	0.927
ES ³ Net (Fang et al., 2023)	0.067	0.491	3.841	0.928
CCNeXt (ours)	0.064$\dagger$	0.445$\dagger$	3.695$\dagger$	0.933$\dagger$

standard deviation for each execution. In instances where the code was unavailable, we replicated the architecture implementation based on information extracted from the papers to extract runtime. In Table 1, ES³Net’s results were reported in this paper but ES³Net was trained using the whole KITTI Raw, which comprises Eigen’s training and testing data as part of the training dataset. This potentially favors their metrics by breaking the equality of comparison with other papers.

Besides the highlighted differences in some metric results, *e.g.*, in the AbsRel metric, it is possible to see many equal or similar results with differences lower than 5%, which is not enough to imply a statistical gap between the outcomes of different methods. Furthermore, it is pertinent to note that while the average is frequently employed for method comparison, the distributions of the results presented here are non-parametric, as evidenced by a significance level of $p < 0.05$ in the Shapiro-Wilk test. Consequently, utilizing the median in this context provides a more appropriate and robust representation. As seen in Table 3, for $\delta < 1.25^2$ and $\delta < 1.25^3$, CCNeXt exhibits no discernible statistical difference when juxtaposed with other state-of-the-art methods. However, when analyzing AbsRel, SqRel, RMSE, and $\delta < 1.25$, CCNeXt demonstrates superior performance, evidenced by statistical significance ($p < 0.05$ in the Wilcoxon test, with the application of the Bonferroni correction to account for multiple comparisons). Consequently, in comparison with H-Net and ES³Net, our approach demonstrates superiority in four out of seven metrics while achieving similar performance in two metrics ($\delta < 1.25^2$ and $\delta < 1.25^3$) and exhibiting inferiority in just one (RMSE log). While we intended to perform statistical analysis using our technique alongside the top three methods from the literature, there were no reproducible codes or weights available to replicate the results of some papers. As

Table 4

Unsupervised Stereo Matching vs CCNeXt. Results on the KITTI Eigen Split comparing the IGEV under self-supervised setting and CCNeXt. All stereo matching models were trained under self-supervised settings by us, since original papers only present results for supervised settings. Best results are **bolded**.

Method	Train	Input	lower is better				higher is better		
			AbsRel	SqRel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
PMN (Chang and Chen, 2018) (Self-Supervised Training)	S	LR	1.659	45.910	20.786	0.986	0.155	0.301	0.440
IGEV (Xu et al., 2023a) (Self-Supervised Training)	S	LR	0.187	5.372	7.251	0.331	0.889	0.926	0.946
CCNeXt (ours)	S	LR	0.075	0.639	4.140	0.175	0.920	0.962	0.979

Table 5

Depth Metric Results on the DrivingStereo. Comparative results with ES³Net using depth estimation metrics. The best results are in **bold**.

Method	lower is better			higher is better
	AbsRel	SqRel	RMSE	$\delta < 1.25$
Monodepth2 (Godard et al., 2019)	0.169	3.669	10.283	0.805
ES ³ Net (Fang et al., 2023)	0.208	10.917	11.206	0.885
CCNeXt (ours)	0.067	1.270	5.457	0.9619

a result, only the H-Net and ES³Net were employed in this phase.

Comparison with Stereo Matching Techniques: We performed experiments using supervised stereo-matching techniques since they had already been adapted to predicting disparity (consequently depth) in a stereo camera set. We conducted experiments using the IGEV (Xu et al., 2023a) and PMN (Chang and Chen, 2018) models applying ES³Net (Fang et al., 2023) training procedures. The results are available in Table 4. To ensure a fair comparison, we report the performance of our model trained at a similarly low resolution, consistent with the settings used for PMN and IGEV. PMN and IGEV did not achieve competitive results in this setting compared to our results. This is likely due to the fact that these architectures are typically optimized to enhance detailed predictions under a supervised loss function, often through complex decoders with various cost volume compositions.

5.2. DrivingStereo Results

Due to the reduced number of evaluation images and low variable scenarios of the KITTI dataset, we perform experiments on a more complex dataset. For that reason, we chose the DrivingStereo dataset. We intended to compare all self-supervised depth estimation published techniques for this evaluation, but unfortunately, we only have training code that allows us to adapt the techniques for a new dataset for ES³Net and older techniques such as Monodepth2.

We report depth metrics in Table 5 and present qualitative results in Figure 6. Comparing the performance of CCNeXt with ES³Net and Monodepth2 on the DrivingStereo dataset, there is a reduction of 67.79% and 60.36% for AbsRel, 88.37% and 65.38% for SqRel, and 51.30% and 46.93% for RMSE, respectively. These reductions are significantly larger than the respective reductions of 5.4% and 34.58% for AbsRel, 11.97% and 31.57% for SqRel,

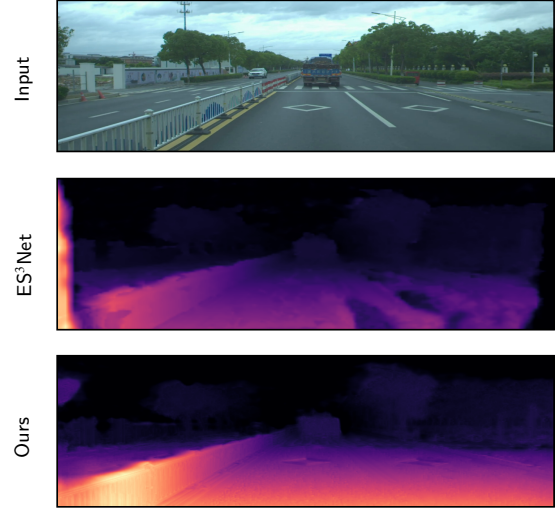


Figure 6: On **top**: Left-view input image from the DrivingStereo dataset. On **middle**: Disparity estimation of ES³Net. On **bottom**: Results of our model (CCNeXt).

Table 6

DrivingStereo Results. EPE and D1 error metrics reported on the DrivingStereo dataset. In the *Train* column: **D**: supervised stereo-matching and **S**: stereo pair depth self-supervised.

Method	Train	EPE	D1 error
SegStereo (Yang et al., 2018)	D	1.13	5.31%
EdgeStereo (Song et al., 2020)	D	1.12	5.35%
GuideNet (Yang et al., 2019)	D	1.36	7.33%
ES ³ Net (Fang et al., 2023)	S	4.406	16.86%
CCNeXt (ours)	S	1.45	6.5%

3.93% and 18.49% for RMSE on the KITTI dataset. Thus, it is noteworthy to assert that evaluating depth on the DrivingStereo dataset appears to accentuate the disparities between the two models. This occurs due to the increased complexity of the DrivingStereo dataset compared to KITTI.

We also report EPE and D1 error metrics in Table 6, since the DrivingStereo dataset presents disparity ground truth results in the original paper (Yang et al., 2019). Comparing the performance of CCNeXt with ES³Net, there is a 67.09% and 61.45% reduction in the EPE and D1 error metrics, respectively. We also included a comparison with other supervised reported models, which shows that the CCNeXt also achieves close metric errors compared to supervised

Table 7

Depth Metric Results on the DrivingStereo Weather Split. Comparative results for the four different weather splits using MonoDepth2 and ES³Net. The best results are in **bold**.

Domain	Method	<i>lower is better</i>				<i>higher is better</i>		
		Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
Cloudy	MonoDepth2 (Godard et al., 2019)	0.412	14.953	15.892	0.411	0.540	0.867	0.936
	ES ³ Net (Fang et al., 2023)	0.187	10.766	9.429	0.289	0.906	0.940	0.957
	CCNeXt (ours)	0.052	0.668	4.388	0.105	0.971	0.989	0.995
Foggy	MonoDepth2 (Godard et al., 2019)	0.345	11.141	16.850	0.361	0.567	0.892	0.950
	ES ³ Net (Fang et al., 2023)	0.254	13.811	13.203	0.355	0.854	0.913	0.938
	CCNeXt (ours)	0.061	0.587	5.409	0.115	0.956	0.987	0.994
Sunny	MonoDepth2 (Godard et al., 2019)	0.407	14.395	15.509	0.419	0.555	0.855	0.928
	ES ³ Net (Fang et al., 2023)	0.148	6.590	8.104	0.273	0.902	0.942	0.960
	CCNeXt (ours)	0.054	0.569	4.279	0.112	0.963	0.987	0.994
Rainy	MonoDepth2 (Godard et al., 2019)	0.281	5.306	13.860	0.306	0.599	0.896	0.959
	ES ³ Net (Fang et al., 2023)	0.570	36.889	25.986	0.756	0.668	0.745	0.788
	CCNeXt (ours)	0.118	3.398	7.623	0.197	0.927	0.966	0.978

models, even though our method solely relies on the stereo pair during training. EdgeStereo (Song et al., 2020) presents 22.75% and 17.69% reduction compared to CCNeXt, but it is a supervised strategy.

Additionally, We evaluate our method on the weather subset, comparing against ES³Net and MonoDepth2 and results are available in Table 7. Our model performs consistently across Cloudy, Foggy, and Sunny conditions, despite not being explicitly trained for weather robustness. It achieves the best results in 27 of 28 metrics, with performance dropping mainly in Rainy, followed by Foggy condition. ES³Net shows a similar trend, with larger differences between Sunny and Cloudy. Although MonoDepth2 outperforms ES³Net on the default DrivingStereo test set, it performs worse on this weather split, except in Rainy, where it achieves its best results. Its ranking across conditions (Rainy > Foggy > Sunny > Cloudy) contrasts with the more stable behavior of CCNeXt and ES³Net.

For weather splits, CCNeXt reduces Abs Rel, Sq Rel, and RMSE by 69.86%, 77.14%, and 46.97% when compared to the second best solution, close to the reductions on the full test set (60.36%, 65.38%, and 46.93%, respectively). The Rainy results highlight the need for strategies that explicitly address weather generalization. Our training relied only on basic augmentations (brightness, hue, contrast), similar to MonoDepth2, and incorporating rain-specific augmentations (e.g., raindrop effects) could already improve performance.

5.3. Computational Efficiency (FLOP Analysis)

In addition to runtime comparisons between our model and ChiTransformer, we conduct a detailed analysis of computational complexity in terms of floating-point operations (FLOPs). FLOPs are computed using `fvcore FlopCountAnalysis` with an input resolution of 1280×384 for CCNeXt and 1216×352 for ChiTransformer, both with a batch size of 1, under a consistent counting convention across models. The analysis includes convolutional, linear, and attention layers, while excluding element-wise operations such as activations and normalizations, following

Table 8

Comparison of FLOPs (GigaFLOPs) Measured Using `fvcore FlopCountAnalysis`. Full architectures and encoders are computed at 1280×384 for CCNeXt and at 1216×352 for ChiTransformer, batch size 1. Per-layer cross-attention FLOPs are measured on (1, 64, 144, 320), matching our first CA stage.

Model / component	FLOPs (G)
<i>Full Architecture</i>	
ChiTransformer (Su and Ji, 2022)	665
CCNeXt (ours)	65.942
<i>Encoder Only</i>	
ChiTransformer (Su and Ji, 2022)	68.168
CCNeXt (ours)	40.264
<i>Per-layer Cross-Attention (on (1, 64, 144, 320))</i>	
Full-row CA	3.563
Windowed Epipolar CA (ours)	1.046

standard practice. Table 8 summarizes the total FLOPs for both the full architectures and their respective encoder components.

When isolating the encoder stage, ChiTransformer exhibits a computational cost of 68.168 GFLOPs, while our CCNeXt encoder requires only 40.264 GFLOPs. This difference becomes even more pronounced when comparing the entire architectures: ChiTransformer demands 665 GFLOPs, whereas CCNeXt operates at just 65.942 GFLOPs, an order of magnitude lower. These results highlight that the extensive use of self-attention and cross-attention layers in ChiTransformer significantly increases its computational burden. In contrast, our architecture maintains a lightweight decoder, which accounts for only 39% of the total FLOPs.

A key contributor to this efficiency is our Windowed Cross-Attention mechanism. To quantify its impact, we compare it against standard full cross-attention. For an input of shape (1, 64, 144, 320), corresponding to the dimensions entering the first cross-attention layer in our network, the full cross-attention module incurs 3.563 GFLOPs, whereas our windowed variant requires only 1.046 GFLOPs, representing a 70.6% reduction in computational cost. Given

Table 9

Ablation Studies. Evaluation of the ConvNeXt encoder, the impact of windowed cross-attention (W-CA), and the contributions of the ICEP module with skip blocks on the KITTI Eigen Split dataset. Rows labeled “Monodepth2 decoder” correspond to the decoder from Godard et al. (2019). For comparison, we also include two Transformer-based encoders alongside ConvNeXt.

Method	lower is better			higher is better
	AbsRel	SqRel	RMSE	$\delta < 1.25$
Incremental Ablation Steps				
ResNet encoder + Monodepth2 decoder	0.086	0.631	4.192	0.914
ConvNeXt encoder (no W-CA) + Monodepth2 decoder	0.077	0.639	4.187	0.922
CCNeXt encoder (with W-CA) + Monodepth2 decoder	0.072	0.654	4.176	0.925
CCNeXt encoder (with W-CA) + CCNeXt decoder	0.070	0.581	3.883	0.926
Additional Encoder Experiments				
Swin Transformer Encoder	0.075	0.617	4.077	0.924
PVT Encoder	0.072	0.583	3.995	0.923
ConvNeXt Encoder	0.070	0.581	3.883	0.926

that multiple cross-attention layers are used throughout our encoder, this saving accumulates across the network, making CCNeXt a favorable option when balancing accuracy with computational efficiency.

6. Ablation Studies

This section presents ablation studies to analyze the contribution of individual stages to our final results (Table 9). We first evaluate changes incrementally, and then compare alternative encoder backbones.

Encoder. When comparing the ResNet-based encoder with the ConvNeXt without the cross-attention layer, it becomes clear that ConvNeXt consistently provides more reliable feature representations for depth prediction. The improvement is most noticeable in the AbsRel metric, where ConvNeXt reduces the relative error, but its benefits also extend to more stable predictions across other metrics. This suggests that the architectural refinements in ConvNeXt, such as larger receptive fields and improved convolutional blocks, contribute to better capturing mid- and long-range dependencies that are important for dense regression tasks like depth estimation. Moreover, once the windowed cross-attention is incorporated, the results further improve, particularly by lowering the RMSE and boosting $\delta < 1.25$. This indicates that the attention mechanism enables the encoder to refine contextual relationships across spatial regions, which are crucial for scenes containing multiple objects at varying scales and distances.

Decoder. The decoder design also plays a critical role in the final performance. Compared to the baseline Monodepth2 decoder, our proposed decoder combined with skip connections produces consistent improvements across nearly all metrics. The most significant impact appears in the SqRel and RMSE metrics, where the inclusion of Skip blocks reduces large squared errors. This effect can be attributed to the Skip connections mitigating phantom regions and enforcing better spatial consistency (Figure 4). By preserving low-level details from earlier layers and mixing them with high-level semantic features, the decoder ensures

smoother depth transitions, particularly at object boundaries. Additionally, the ICEP module enhances the information exchange between encoder and decoder stages, enabling the network to better reconstruct fine-grained details such as thin structures (e.g., poles, railings, and traffic signs) while avoiding the over-smoothing often observed in standard decoders.

Qualitative results for these variants are shown in Figure 7. Relative to the ResNet baseline with the Monodepth2 decoder, the CCNeXt architecture produces smoother and more coherent depth maps, with fewer regions of large error. It also captures relative depth ordering more accurately (e.g., car vs. traffic sign, traffic sign vs. wall), leading to more consistent scene geometry.

Additional Encoder Experiments. To further validate the choice of ConvNeXt as our default encoder, we extended our analysis to two representative Transformer-based alternatives: the Swin Transformer (Liu et al., 2021) and Pyramid Vision Transformer v2 (PVTv2) (Wang et al., 2022a). Both models were selected in configurations with computational costs close to our ConvNeXt variant, allowing for a fair comparison in terms of accuracy–efficiency trade-off. The tested Swin Transformer (Tiny version, complete encoder has 105 GFLOPs) and PVTv2 (PVTv2-B1, complete encoder has 47.052 GFLOPs) were compared against our ConvNeXt (complete encoder has 43.567 GFLOPs), where the Transformer-based baselines were further enhanced with Windowed Cross-Attention blocks between encoder stages to ensure a consistent comparison.

Performance remains below ConvNeXt across all major metrics, particularly in AbsRel and RMSE. This suggests that, in the context of the stereo self-supervised depth estimation, local window-based self-attention is less effective than ConvNeXt’s convolutional designs for capturing dense spatial correlations. Furthermore, the significantly higher FLOPs of Swin highlight an unfavorable accuracy–efficiency balance, making it less practical for real-world deployment.

In contrast, the PVT encoder yields results that are close to ConvNeXt, with minor degradation in precision-based metrics such as $\delta < 1.25$. The hierarchical design of PVT appears to retain competitive representational power while remaining computationally reasonable. However, ConvNeXt still offers the best overall trade-off: it achieves the lowest AbsRel and SqRel, while maintaining lower complexity than both Transformer alternatives.

These findings underscore that, despite the recent popularity of Transformer-based encoders, well-designed convolutional architectures such as ConvNeXt remain highly competitive for dense regression problems, particularly when computational efficiency is a priority. In particular, ConvNeXt balances accuracy, efficiency, and stability across diverse depth estimation metrics, making it a strong choice for scenarios where computational budgets are constrained. Nonetheless, PVT emerges as a promising candidate for scaling to larger and more diverse datasets, where the global

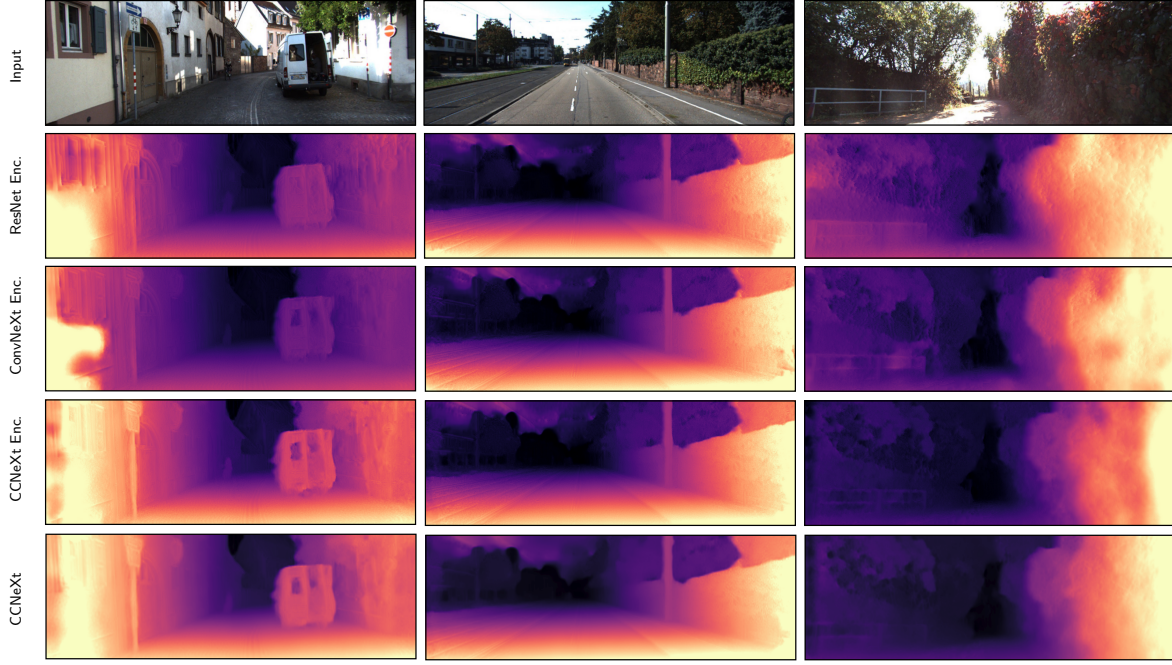


Figure 7: Qualitative results of the ablation models on the KITTI Eigen Split test set. CCNeXt exhibits better consistency over region predictions than the other methods. It also produces less noisy regions.

reasoning capacity of Transformers may provide further benefits.

7. Limitations and Future Directions

Our model has three main limitations: performing inference with (i) different datasets, (ii) multi-dataset training, and (iii) non-rectified systems. Since we are performing the backward projection from 2D to 3D and then the perspective projection from 3D to 2D, we depend on the camera system’s internal parameters to infer depth and create the reprojection image. Thus, performing inference in unseen datasets could be problematic, especially using the predicted rescaled disparity extracted from $(a\sigma + b)$.

This is also the same limitation encountered in training using multiple datasets simultaneously. We could solve the rescaling disparity limitation if we directly used equation $\text{depth} = Bf_x/\text{disp}$ as performed in the DrivingStereo setting. Another possible solution for both limitations would be to predict the pixel disparity with the model and directly warp one image view to generate the other using disparity information. In this case, it would be necessary to define minimum and maximum pixel disparity values as the same for all datasets.

Finally, we need a pair of rectified images to do the depth and disparity matching. If all epipolar lines are parallel to the horizontal axis and corresponding points are in the same vertical coordinate, we satisfy the properties of a rectified image (Loop and Zhang, 1999). To fix unrectified image pairs, online rectification methods have been extensively researched (Wang et al., 2023; Kumar et al., 2018; Rehder et al., 2017; Lafiosca and Ceccaroni, 2022), but they were not

extensively explored in our work. We conducted qualitative experiments with Direct Stereo Rectification (Lafiosca and Ceccaroni, 2022) on the raw KITTI data, which contains unrectified stereo pairs. An example is shown in Figure 8. The prediction reveals coarse errors introduced by inaccuracies in the online rectification step. These errors are particularly visible when comparing the orientation of vertical structures (e.g., poles in the scene center) across the two views, where imperfect alignment propagates into disparity estimation mistakes.

Beyond these specific limitations, we also highlight several developmental tendencies and open challenges for the field. First, integrating contrastive or self-distillation objectives with geometry-constrained attention may further enhance representation quality without increasing supervision requirements. Second, while CCNeXt demonstrates strong efficiency, future research should explore even lighter architectures to enable deployment on embedded platforms with strict latency and power budgets. Finally, robustness under adverse environmental factors such as extreme weather, nighttime, or sensor noise remains a critical challenge, where larger-scale datasets and adaptive learning strategies will be essential.

8. Conclusions

This paper introduced CCNeXt, a novel convolutional network architecture designed for self-supervised stereo depth estimation. We enhanced the encoder module by incorporating a contemporary convolutional feature extractor coupled with a novel windowed epipolar cross-attention module. Additionally, we redesigned the widely employed

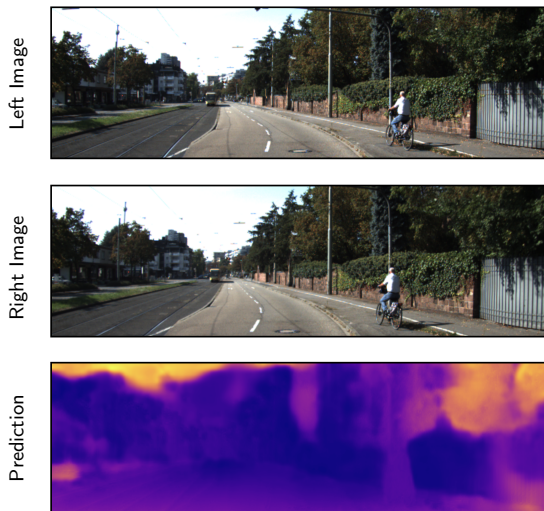


Figure 8: On **top**: Left-view input image from the KITTI dataset rectified using Direct Stereo Rectification. On **middle**: Right-view input image from the KITTI dataset rectified using Direct Stereo Rectification. On **bottom**: Results of our model (CCNeXt) on this input pair.

Monodepth2 decoder to facilitate improved backpropagation and feature representations, resulting in enhanced estimations. Experiments demonstrate that CCNeXt not only achieves state-of-the-art results on KITTI but also establishes new benchmarks on DrivingStereo, outperforming prior methods in both the general split and the challenging weather-specific splits. This highlights the robustness of our approach under diverse conditions while being 10.18× faster than the current best model on the KITTI dataset.

9. Acknowledgments

The authors are thankful to the National Council for Scientific and Technological Development (grant #304836/2022-2) for its financial support.

References

Bhat, S.F., Birkel, R., Wofk, D., Wonka, P., Müller, M., 2023. ZoeDepth: Zero-Shot Transfer by Combining Relative and Metric Depth. arXiv preprint arXiv:2302.12288.

Chang, J.R., Chang, P.C., Chen, Y.S., 2020. Attention-Aware Feature Aggregation for Real-Time Stereo Matching on Edge Devices, in: Asian Conference on Computer Vision, pp. 1–16.

Chang, J.R., Chen, Y.S., 2018. Pyramid Stereo Matching Network, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5410–5418.

Duggal, S., Wang, S., Ma, W.C., Hu, R., Urtasun, R., 2019. DeepPruner: Learning Efficient Stereo Matching via Differentiable Patchmatch, in: IEEE/CVF International Conference on Computer Vision, pp. 4384–4393.

Eigen, D., Puhrsch, C., Fergus, R., 2014. Depth Map Prediction from a Single Image using a Multi-Scale Deep network. Advances in Neural Information Processing Systems 27.

Fang, I., Wen, H.C., Hsu, C.L., Jen, P.C., Chen, P.Y., Chen, Y.S., 2023. ES3Net: Accurate and Efficient Edge-Based Self-Supervised Stereo Matching Network, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4471–4480.

Fu, H., Gong, M., Wang, C., Batmanghelich, K., Tao, D., 2018. Deep Ordinal Regression Network for Monocular Depth Estimation, in: IEEE/CVF International Conference on Computer Vision, pp. 2002–2011.

Garg, R., Bg, V.K., Carneiro, G., Reid, I., 2016. Unsupervised CNN for Single View Depth Estimation: Geometry to the Rescue, in: European Conference on Computer Vision, Springer, pp. 740–756.

Geiger, A., Lenz, P., Stiller, C., Urtasun, R., 2013. Vision meets Robotics: The KITTI Dataset. International Journal of Robotics Research (IJRR).

Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3354–3361.

Godard, C., Mac Aodha, O., Brostow, G.J., 2017. Unsupervised Monocular Depth Estimation with Left-Right Consistency, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 270–279.

Godard, C., Mac Aodha, O., Firman, M., Brostow, G.J., 2019. Digging into self-supervised monocular depth estimation, in: IEEE International Conference on Computer Vision, pp. 3828–3838.

Guizilini, V., Ambrus, R., Pillai, S., Raventos, A., Gaidon, A., 2020. 3D Packing for Self-Supervised Monocular Depth Estimation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2485–2494.

Guizilini, V., Ambrus, R., Chen, D., Zakharov, S., Gaidon, A., 2022. Multi-Frame Self-Supervised Depth with Transformers, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 160–170.

Guo, W., Li, Z., Yang, Y., Wang, Z., Taylor, R.H., Unberath, M., Yuille, A., Li, Y., 2022. Context-enhanced stereo Transformer, in: European Conference on Computer Vision, Springer, pp. 263–279.

Hu, Z., Gao, J., Yuan, Y., Li, X., 2024. Contrastive Tokens and Label Activation for Remote Sensing Weakly Supervised Semantic Segmentation. IEEE Transactions on Geoscience and Remote Sensing 62, 1–11.

Huang, B., Zheng, J.Q., Giannarou, S., Elson, D.S., 2022. H-Net: Unsupervised Attention-based Stereo Depth Estimation Leveraging Epipolar Geometry, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 4460–4467.

Izquierdo, S., Civera, J., 2023. SfM-TTR: Using Structure from Motion for Test-Time Refinement of Single-View Depth Networks, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21466–21476.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Klingner, M., Termöhlen, J.A., Mikolajczyk, J., Fingscheidt, T., 2020. Self-Supervised Monocular Depth Estimation: Solving the Dynamic Object Problem by Semantic Guidance, in: 16th European Conference on Computer Vision, Springer, Glasgow, UK, pp. 582–600.

Kumar, A., Gururaj, M., Seshadrinathan, K., Narayanswamy, R., 2018. Multi-Capture Dynamic Calibration of Multi-Camera Systems, in: IEEE/CVF International Conference on Computer Vision Workshops, pp. 1843–1851.

Kusupati, U., Cheng, S., Chen, R., Su, H., 2020. Normal Assisted Stereo Depth Estimation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2189–2199.

Lafiosca, P., Ceccaroni, M., 2022. Rectifying Homographies for Stereo Vision: Analytical Solution for Minimal Distortion, in: Intelligent Computing, pp. 484–503.

Laga, H., Jospin, L.V., Boussaid, F., Bennamoun, M., 2020. A Survey on Deep Learning Techniques for Stereo-Based Depth Estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence 44, 1738–1764.

Levinson, J., Askeland, J., Becker, J., Dolson, J., Held, D., Kammel, S., Kolter, J.Z., Langer, D., Pink, O., Pratt, V., Sokolsky, M., Stanek, G., Stavens, D., Teichman, A., Werling, M., Thrun, S., 2011. Towards Fully Autonomous Driving: Systems and Algorithms, in: IEEE Intelligent Vehicles Symposium, pp. 163–168.

Li, C., Li, G., Li, T.H., 2021a. Separating Left and Right Images in Unsupervised Monocular Depth Estimation, in: IEEE International Conference on Power Electronics, Computer Applications (ICPECA), IEEE, pp. 65–69.

- Li, Z., Liu, X., Drenkow, N., Ding, A., Creighton, F.X., Taylor, R.H., Unberath, M., 2021b. Revisiting Stereo Depth Estimation from a Sequence-to-Sequence Perspective with Transformers, in: IEEE/CVF International Conference on Computer Vision, pp. 6197–6206.
- Li, Z., Wang, X., Liu, X., Jiang, J., 2024. Binsformer: Revisiting Adaptive Bins for Monocular Depth Estimation. *IEEE Transactions on Image Processing*.
- Lipson, L., Teed, Z., Deng, J., 2021. Raft-Stereo: Multilevel Recurrent Field Transforms for Stereo Matching, in: International Conference on 3D Vision (3DV), IEEE. pp. 218–227.
- Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B., 2021. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows, in: IEEE/CVF International Conference on Computer Vision, pp. 10012–10022.
- Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S., 2022. A ConvNet for the 2020s, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11976–11986.
- Loop, C., Zhang, Z., 1999. Computing Rectifying Homographies for Stereo Vision, in: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, IEEE. pp. 125–131.
- Luo, X., Huang, J.B., Szeliski, R., Matzen, K., Kopf, J., 2020. Consistent Video Depth Estimation. *ACM Transactions on Graphics*, 71–1.
- Park, N., Kim, S., 2022. How Do Vision Transformers Work? *arXiv preprint arXiv:2202.06709*, 1–26.
- Piccinelli, L., Sakaridis, C., Yu, F., 2023. iDisc: Internal Discretization for Monocular Depth Estimation, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21477–21487.
- Pilzer, A., Lathuilière, S., Xu, D., Puscas, M.M., Ricci, E., Sebe, N., 2019. Progressive Fusion for Unsupervised Binocular Depth Estimation using Cycled Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 2380–2395.
- Rehder, E., Kinzig, C., Bender, P., Lauer, M., 2017. Online Stereo Camera Calibration from Scratch, in: IEEE Intelligent Vehicles Symposium, IEEE. pp. 1694–1699.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115, 211–252.
- Shao, S., Pei, Z., Wu, X., Liu, Z., Chen, W., Li, Z., 2024. Iebins: Iterative Elastic Bins for Monocular Depth Estimation. *Advances in Neural Information Processing Systems* 36.
- Smith, S.L., Brock, A., Berrada, L., De, S., 2023. ConvNets Match Vision Transformers at Scale. *arXiv preprint arXiv:2310.16764*.
- Song, X., Zhao, X., Fang, L., Hu, H., Yu, Y., 2020. EdgeStereo: An Effective Multi-Task Learning Network for Stereo Matching and Edge Detection. *International Journal of Computer Vision* 128, 910–930.
- Stoyanov, D., Scarzanella, M.V., Pratt, P., Yang, G.Z., 2010. Real-Time Stereo Reconstruction in Robotically Assisted Minimally Invasive Surgery, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 275–282.
- Su, Q., Ji, S., 2022. ChiTransformer: Towards Reliable Stereo from Cues, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1939–1949.
- Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A., 2017. Sparsity Invariant CNNs, in: International Conference on 3D Vision, IEEE. pp. 11–20.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention is All You Need. *Advances in Neural Information Processing Systems* 30.
- Wang, H., Fan, R., Cai, P., Liu, M., 2021. PVStereo: Pyramid Voting Module for End-to-End Self-Supervised Stereo Matching. *IEEE Robotics and Automation Letters* 6, 4353–4360.
- Wang, J., Scharstein, D., Bapat, A., Blackburn-Matzen, K., Yu, M., Lehman, J., Alsison, S., Wang, Y., Tsai, S., Frahm, J.M., 2023. A Practical Stereo Depth System for Smart Glasses, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21498–21507.
- Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L., 2022a. PVTv2: Improved Baselines with Pyramid Vision Transformer. *Computational Visual Media* 8, 1–10.
- Wang, X., Girshick, R., Gupta, A., He, K., 2018. Non-Local Neural Networks, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 7794–7803.
- Wang, X., Zhu, Z., Huang, G., Qin, F., Ye, Y., He, Y., Chi, X., Wang, X., 2022b. Mvster: Epipolar Transformer for Efficient Multi-View Stereo, in: European Conference on Computer Vision, Springer. pp. 573–591.
- Wang, Y., Liang, Y., Xu, H., Jiao, S., Yu, H., 2024. SQLdepth: Generalizable Self-Supervised Fine-Structured Monocular Depth Estimation, in: AAAI Conference on Artificial Intelligence, pp. 5713–5721.
- Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image Quality Assessment: from Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing* 13, 600–612.
- Wei, Y., Zhao, L., Zheng, W., Zhu, Z., Rao, Y., Huang, G., Lu, J., Zhou, J., 2023. Surrounddepth: Entangling Surrounding Views for Self-Supervised Multi-Camera Depth Estimation, in: Conference on Robot Learning, PMLR. pp. 539–549.
- Xiao, Z., Tong, H., 2025. Federated Contrastive Learning with Feature-Based Distillation for Human Activity Recognition, IEEE.
- Xie, Z., Geng, Z., Hu, J., Zhang, Z., Hu, H., Cao, Y., 2023. Revealing the Dark Secrets of Masked Image Modeling, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14475–14485.
- Xu, G., Cheng, J., Guo, P., Yang, X., 2022. Attention Concatenation Volume for Accurate and Efficient Stereo Matching, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12981–12990.
- Xu, G., Wang, X., Ding, X., Yang, X., 2023a. Iterative Geometry Encoding Volume for Stereo Matching, in: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21919–21928.
- Xu, H., Zhang, J., Cai, J., Rezaatofghi, H., Yu, F., Tao, D., Geiger, A., 2023b. Unifying Flow, Stereo and Depth Estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Yang, G., Song, X., Huang, C., Deng, Z., Shi, J., Zhou, B., 2019. DrivingStereo: A Large-Scale Dataset for Stereo Matching in Autonomous Driving Scenarios, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 899–908.
- Yang, G., Zhao, H., Shi, J., Deng, Z., Jia, J., 2018. SegStereo: Exploiting Semantic Information for Disparity Estimation, in: European Conference on Computer Vision, pp. 636–651.
- Zhong, Y., Dai, Y., Li, H., 2017. Self-Supervised Learning for Stereo Matching with Self-Improving Ability. *arXiv preprint arXiv:1709.00930*, 1–21.
- Zhou, T., Brown, M., Snavely, N., Lowe, D.G., 2017. Unsupervised Learning of Depth and Ego-Motion from Video, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1851–1858.