

# SteReFo: Efficient Image Refocusing with Stereo Vision

Benjamin Busam<sup>\*,1,2</sup>    Matthieu Hog<sup>\*,1</sup>    Steven McDonagh<sup>1</sup>    <sup>\*</sup>equal contribution  
b.busam@tum.de    matthieu.hog@huawei.com    steven.mcdonagh@huawei.com  
Gregory Slabaugh<sup>1</sup>    <sup>1</sup>Huawei Noah's Ark Lab    <sup>2</sup>Technical University of Munich  
gregory.slabaugh@huawei.com

## Abstract

Whether to attract viewer attention to a particular object, give the impression of depth or simply reproduce human-like scene perception, shallow depth of field images are used extensively by professional and amateur photographers alike. To this end, high quality optical systems are used in DSLR cameras to focus on a specific depth plane while producing visually pleasing bokeh.

We propose a physically motivated pipeline to mimic this effect from all-in-focus stereo images, typically retrieved by mobile cameras. It is capable to change the focal plane a posteriori at 76 FPS on KITTI [13] images to enable real-time applications. As our portmanteau suggests, *SteReFo* interrelates stereo-based depth estimation and refocusing efficiently. In contrast to other approaches, our pipeline is simultaneously fully differentiable, physically motivated, and agnostic to scene content. It also enables computational video focus tracking for moving objects in addition to refocusing of static images. We evaluate our approach on publicly available datasets [13, 33, 9] and quantify the quality of architectural changes.

## 1. Introduction

**Motivation.** Around the turn of the millennium, Japanese photographers coined the term *bokeh* for the soft, circular out-of-focus highlights produced by near circular apertures [41]. To this day, bokeh is a sign of high quality photographs acquired using professional equipment, closely linked to the depth of field of the optical system in use [36]. Historically, producing such photos has been exclusively possible with high-end DSLRs. Synthesizing the effect of such high-end hardware finds application in particular in consumer mobile devices where the goal is to mimic the physical effects of high-quality lenses in silico [19]. Due to the inherent narrow aperture of cost-efficient optical systems commonly used in mobile phones, the acquired image is all-in-focus. This property hampers the natural image background defocus often desired in many types of scene

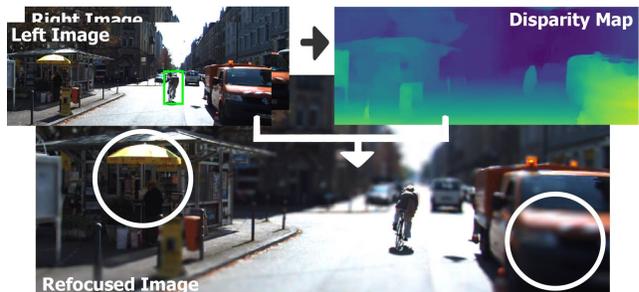


Figure 1: *SteReFo* on video sequence of [13]. A disparity map is computed from binocular images while a 2D tracker provides a bounding box (green) to look up the focus depth on the object of interest (the cyclist). With the retrieved depth, the proposed differentiable refocusing stage (white arrow) is utilized to refocus the input frame. The refocused image is in focus in areas that are equal in depth to the cyclist (left) while closer (right) and more distant regions are blurred.

capture, such as portrait images.

To address this problem a trend has emerged, where shallow depth of field images are computationally synthesized from all-in-focus images [50], usually by leveraging a depth estimation. In the rest of the paper we refer to this task as *refocusing*.

**Drawbacks of recent approaches.** The portrait mode of recent smartphones uses depth estimation from monocular [51] or dual-pixel [50] cameras. To circumvent depth estimation errors, previous approaches rely heavily on segmentation of a single salient object, making them limited to scenes with a unique, predominant region of interest. Moreover, this restriction limits the applicability of the underlying refocusing pipeline in other use cases such as object-agnostic image and video refocusing.

**Contributions and Outline.** We present a general approach that utilizes stereo vision to refocus images and videos (cf. Fig. 1). Our pipeline, entitled *SteReFo*, leverages the state-of-the-art in efficient stereo depth estimation to obtain a high-quality disparity map and uses a fast, dif-

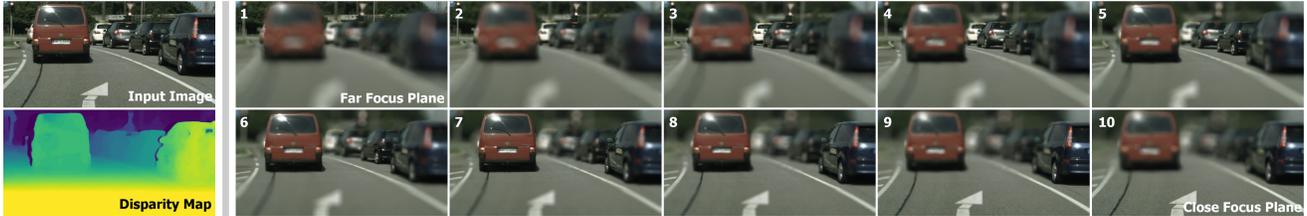


Figure 2: Computational refocusing of an image from [9]. On the left, one input image (from the stereo pair) together with the intermediate disparity map is illustrated while the right part depicts a continuous sequential refocusing on depth planes from far (1) to close (10). Note the smooth transition of the refocus plane, not feasible with segmentation approaches. Also, note the physically motivated radial bokeh effect on the right traffic light in (7), (8) similar to the effects produced by high-end DSLR equipment. Stereo imagery enables in particular high depth precision for sharp boundaries which can be observed e.g. on the right side of the red car.

differentiable layered refocusing algorithm to perform the refocusing (Fig. 3 shows the overall pipeline). A total runtime of 0.14 sec (0.11 sec for depth and 0.03 sec for refocusing) makes it computationally tractable for portable devices. Moreover, our method is agnostic to objects present in the scene and the user retains full control of both blur intensity and focal plane (cf. Fig. 2). We also conduct a study to assess the optimal way to combine depth information with the proposed layered refocusing algorithm. Unlike previous work, we quantify the refocus quality of our methods by means of a perceptual metric. More specifically, our contributions are:

1. An efficient pipeline for **refocusing from stereo images** at interactive frame-rates with a **differentiable formulation of refocusing** for modular use in neural networks.
2. The proposal and study of novel **architectures to combine stereo vision and refocusing** for physically motivated bokeh.
3. Both **qualitative** and **quantitative** analysis of our approach on **synthetic** and **real** images from SceneFlow [33], KITTI [13] and CityScapes [9].
4. A combination of 2D tracking and depth-based refocusing to enable **computational focus tracking in videos** with tractable computational complexity.

To the best of our knowledge, *SteReFo* is the first method that is jointly trainable for stereo depth and refocusing, made possible by the efficient design of our differentiable refocusing. Our model makes effective refocusing attainable, yet the approach does not require semantic priors and is not limited in blur intensity. We show that it is possible to mimic the manual refocusing effects found in video acquisition systems by autonomous parameter adjustment.

## 2. Related Literature

Large bodies of work exist in the domains of both vision-based depth estimation and computational refocusing. We

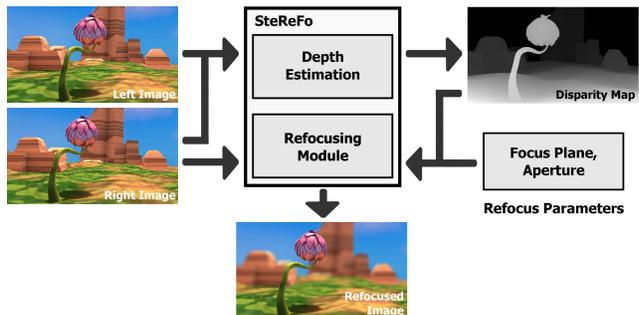


Figure 3: The refocusing pipeline. A stereo pair of all-in-focus images is processed by the depth estimation module which outputs a disparity map. The disparity map together with one input image and the refocus parameters are the input for our efficient refocusing pipeline which leverages the proposed layered depth of field to virtually set a focus plane to refocus the input image.

briefly review work most relevant to ours, putting our contributions into context.

### 2.1. Depth Estimation

Depth estimation from imagery is a well studied problem with a long history to perform estimation from image pairs [43, 30, 47], from temporal image sequences in classical structure from motion (SfM) [12, 22] and simultaneous localization and mapping (SLAM) [18, 35, 11] and reasoning about overlapping images with varying viewpoint [2, 26]. In addition, the task of single image depth estimation has shown recent progress using contemporary learning based methods [29, 14, 16, 32].

**Monocular vision.** Deep learning based monocular depth estimation employ CNNs to directly infer depth from a static monocular image. They are either trained fully supervised (with a synthetic dataset or ground truth from different sensors) [10, 29] or leverage multiple cameras at training time to use photo-consistency for supervision [14, 40].

However, these approaches are in general tailored for a specific use case and suffer from domain shift errors. To address this drawback, stereo matching [16] or multi-view stereo [32] can be used as a proxy. While these recent approaches estimate reliable depth values, the depth often suffers from over-smoothing [15] which manifests as “flying pixels” in the free space found across depth discontinuities. Accurately and faithfully reproducing such boundaries is, however, critically important for subsequent refocusing quality. Therefore we focus our approach on a binocular stereo cue.

**Multi-view prediction.** For high-accuracy depth maps that preserve precise object boundaries, multiple views are still necessary [44] and binocular stereo is supported by large synthetic [33] as well as real datasets (cf. KITTI [13], CityScapes [9]). Leveraging this data, StereoNet [25] uses a hierarchical disparity prediction network with a deep visual feature backbone which is capable of running at 60 FPS on a consumer GPU. Its successor [56] extends the work with self-supervision to the domain of active sensing while maintaining the core efficiency. We build upon their work to leverage this computational advantage.

More recently, Tonioni et al. [48] have proposed a way to perform continuous online domain adaptation for disparity estimation with real-time applicability.

**Other modalities and applications.** Fusion of different visual cues can boost accuracy of individual tasks. Leveraging temporal stereo sequences for unsupervised monocular depth and pose estimation, e.g. by warping deep features, improves the accuracy of both tasks [55]. With the same result, Zou et al. [60] jointly train for optical flow, pose and depth estimation simultaneously while Jiao et al. [23] mutually improve semantics and depth and GeoNet [53] jointly estimates depth, optical flow and camera pose from video. Fully unsupervised monocular depth and visual odometry can also be entangled [58] and 3D mapping applications [57] are realized by heavily relying on dense optical flow in 2D and 3D. Despite the superiority of these approaches, they suffer from larger computational burden or come at the cost of additional training data.

## 2.2. Refocusing

**Refocusing algorithms** are heavily utilized in video-games and animated movie production. A plethora of approaches has been proposed for shallow depth of field rendering in the computer graphics community. We follow the taxonomy in [38] and refer the reader to [5] for a complete survey. The first style of approach uses ray tracing to accurately reproduce the ray integration performed in a camera body. While some approaches focus on physical accuracy [39] and others on (relative) speed [52], these methods are very computationally expensive (up to hours per frame). It is also

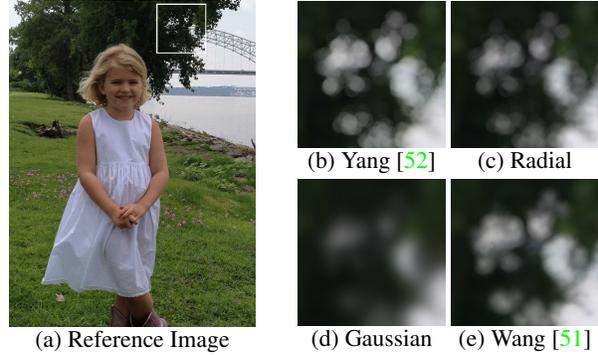


Figure 4: Refocusing results from different blurring techniques. We display the reference image (a) used by Wang et al. [51], and a crop for the different results from a pseudo ray-traced approach [52] (b), a simple radial blur (c), a simple Gaussian blur (d) and the result in [51] (e). We observe that the blurred regions for the Gaussian blur and [51] lack the distinctive bokeh aspect of DSLR, while physically motivated approaches such as the ray-traced approach and the radial blur, preserve well the bokeh. The latter serves as a backbone in our pipeline (Alg. 1).

possible to render a set of views, at different viewpoints, with fast classical rasterization techniques (*i.e.* creating a light-field [31]). Views are then accumulated to produce a refocused image [17]. However, this requires the scene to be rendered using an amount of time quadratic in the size of the maximum equivalent blur kernel, which is computationally intractable.

This point motivated approaches that seek to reproduce the blur in the image domain directly. Applying depth-adaptive blur kernels can be formulated as scatter [28] or gather [42] operations. While the first is hard to parallelize, the latter suffers from sharply blurred object edges and intensity leakage. Moreover, because the blur kernel is different for each pixel, these approaches are hard to optimize for GPUs [51]. Finally, the last class of algorithms represents the scenes as depth layers in order to apply blur with fixed kernels separately [27]. We give special attention to this type of algorithm in Sec. 3.1.

**Refocusing pipelines.** In contrast to computer graphics, where scene depth and occluded parts can be retrieved easily, in computer vision, the estimated depth is often noisy and background information is not necessarily available due to the projective nature of cameras.

This issue can be addressed through a hole filling task for missing pixel depth [8] or by leveraging an efficient bilateral solver for stereo-based depth [4]. Yu et al. [54] directly reconstruct such a light field from stereo images, similar to the techniques discussed previously. It leverages depth estimation, forward warping plus inpainting to reconstruct

a reasonable number of views that can be interpolated and summed to render the final image. The same idea is presented in [46], but the approach is fully learned with single image input and light field supervision. Zhu et al. [59] consider a refocus task using smartphone-to-DSLR image translation. However, authors concede that average performance is considerably worse than highlighted results.

More recently, in [50] is presented a complete pipeline that computes a person segmentation, a depth map from dual pixel and finally the refocused image. While the results are visually compelling, the method is limited to focusing on a person in the image foreground. In [51], the authors decompose the problem into three modules: monocular depth estimation, blurring and upsampling. While the approach provides visually pleasing images it is unclear how it generalizes, given its fully synthetic training set. Due to the blur step being completely learned, we observed that the images lack the distinctive circular bokeh that professional DSLR cameras produce (see Fig. 4).

Finally, Srinivasan et al. [45] proposes light field synthesis and supervision with refocused images (i.e. aperture supervision) to learn refocusing. Due to the synthesis of a multitude of views in the first approach, it does not scale well with large kernels. The latter uses the all-in-focus image with a variety of radial kernels and the network is trained to select, for each pixel, which blur value is most likely. The final image is a composition of these blurred images. This approach has the limitation that both the focal plane and the aperture are fixed and cannot be manipulated by a user. In contrast, we want to keep the system parameterizable.

### 3. Methodology

As illustrated in Fig. 3, our pipeline is split into two modules: depth estimation and refocusing. The inputs to the pipeline are a pair of rectified left and right stereo images. A focus plane and an aperture are two user-controllable parameters. In the following, we explain our proposal for the *SteReFo* module of Fig. 3 in four different variants which we compare subsequently.

#### 3.1. Disparity Estimation

To produce high quality depth maps, our architecture takes inspiration from two state-of-the-art pipelines for real-time disparity prediction, namely StereoNet [25] and ActiveStereoNet [56] which estimate a subpixel precise low-resolution disparity map that is consecutively upsampled and refined with RGB-guidance from the reference image. Our depth estimation network consists of two Siamese towers with shared weights that extract deep image features at 1/8 of the stereo pair resolution following the architecture described in [56]. We construct a cost volume (CV) by concatenation of the displaced features along the epipolar lines of the rectified input images. The discretization is chosen

---

**Algorithm 1:** The layered depth of field base algorithm used in our approach.

---

**Input :** All-in-focus Image  $I$  with associated disparity map  $D$ , focus plane  $d^f$ , aperture  $a$ , and disparity range  $[d_{min}, d_{max}]$

**Output:** Image  $I_b$  refocused on the depth plane  $d^f$

```

1  $I_s = [0]$ 
2  $M_s = [0]$ 
3 for  $d \leftarrow d_{min}$  to  $d_{max}$  by  $\frac{1}{a}$  do
4    $M^d = |D - d| < \frac{1}{a}$ 
5    $I^d = M^d \circ I$ 
6    $r = a \cdot (d - d^f)$ 
7    $M_b^d = M^d * K(r)$ 
8    $I_b^d = I^d * K(r)$ 
9    $M_s = M_s \circ (1 - M_b^d) + M_b^d$ 
10   $I_s = I_s \circ (1 - M_b^d) + I_b^d$ 
11 end
12  $I_b = I_s \circledast M_s$ 

```

---

to include 18 bins. A shifted version of the differentiable ArgMin operator [24] recovers disparities from  $i = 0$  to  $D_{max} = 17$  where the disparities are given by

$$d_i = \sum_{d=1}^{D_{max}+1} d \cdot \sigma(-C_i(d)) - 1 \quad (1)$$

with the softmax operator  $\sigma$  and the cost  $C_i$ . The low resolution disparity map defined in Eq. 1 is then hierarchically upsampled ( $\frac{1}{8} \rightarrow \frac{1}{4} \rightarrow \frac{1}{2} \rightarrow$  full resolution) using bilinear interpolation. Following the idea of Khamis et al. [25], we use residual refinement to recover high-frequency details. Prior to stacking the resized image and low-resolution disparity map, we pass both individually through a small network with 1 convolution and 3 ResNet [20] blocks, as we observed this robustifies our depth prediction quality.

The module is trained using a Barron loss [3] with parameters  $\alpha = 1$ ,  $c = 2$  and RMSProp [21] optimization with an exponentially decaying learning rate.

#### 3.2. Efficient Layered Depth of Field

Our refocusing module utilizes layered depth of field rendering to enable efficient refocusing. The core idea of layered depth of field rendering [27] is to first decompose the scene into depth layers in order to separately blur each layer before compositing them back together. In contrast to [51], which learns kernel weights, this physically motivated choice directly reflects the effects obtained by DSLR lenses while providing an appropriate balance between efficiency and accuracy for our runtime requirements. Using this approach, the blur operation is applied by combining fixed-kernel convolutions, that make it very efficient in practice

due to contemporary GPU convolutional implementations. We describe the algorithm in Alg. 1, where the  $\circ$  and  $\oslash$  notation are used for the entrywise Hadamard product and division respectively, and  $*$  denotes convolution. We start from an all-in-focus image  $I$  with its associated disparity map  $D$ , a user-set focus plane  $d^f$ , an aperture  $a$  and a disparity range  $[d_{min}, d_{max}]$  defined by the stereo setup capabilities.  $I_s$  and  $M_s$  are two accumulation buffers. We sweep the scene from back to front within a given disparity range using a step size of (optimally)  $\frac{1}{a}$ . A mask  $M^d$ , defining the zones within a disparity window around the disparity plane  $d$ , is used to extract the corresponding texture of objects within depth plane  $I^d$ . The corresponding blur radius  $r$  is computed from the distance of the focal plane  $d^f$  to the current depth plane  $d$  and the given aperture  $a$ . The extracted mask and texture are blurred with a radial kernel  $K(r)$  of diameter  $r$ . The blurred mask and texture are accumulated in the buffers  $I_s$  and  $M_s$ , overwriting the previous values where the mask is not 0, in order to handle clipping in the blur (i.e. prevent out of focus regions to bleed into in-focus regions). The final blurred image  $I_b$  is rendered by normalizing the accumulated blur texture with the accumulated masks.

**Adaptive downsampling.** We alter the base algorithm described in Section 3.1 in two ways. To further increase the efficiency of the pipeline, we set a maximum kernel size  $k_{max}$  that may be applied. For a given disparity plane  $d$ , we resize the input image by a factor of  $\gamma = \lceil 2r + 1 \rceil / k_{max}$  and apply the convolution with a kernel size  $K(\gamma \cdot r)$ . The blur result is then upsampled to full resolution using bilinear interpolation. While this is an approximation, the visual difference is marginal due to its application to out of focus regions. However, the computational efficiency is improved by several orders of magnitude (cf. Sec. 4.2).

**Differentiability.** The second modification we carry out is making this algorithm differentiable in order to use it in an end-to-end trainable pipeline. In Alg. 1, it can be observed that all operations carried out are differentiable, except for computation of the mask which relies on the non-smooth *less than* operator in line 4. By expressing this operator using the Heaviside step function, the mask computation can be written as:

$$M^d = H\left(\frac{1}{a} - |D - d|\right) \text{ where } H(x) = \begin{cases} 0 & x < 0 \\ 1 & x \geq 0 \end{cases} \quad (2)$$

While the Heaviside step function itself is non-differentiable, a smooth approximation is given by  $\hat{H}(x) = \frac{1}{2} + \frac{1}{2} \tanh(x)$ . Hence we can replace line 4 in Alg. 1 with

$$\hat{M}^d = \frac{1}{2} + \frac{1}{2} \tanh\left(\alpha \cdot \left(\frac{1}{a} - |D - d|\right)\right) \quad (3)$$

where  $\alpha$  controls the transition *sharpness* of the Heaviside step function approximation. We empirically set  $\alpha = 10^3$ .

### 3.3. Refocusing Architectures

To intertwine stereo depth estimation and refocusing in *SteReFo*, we investigate the four architectures illustrated in Fig. 5. The first ( $A_1$ ), dubbed *Sequential depth*, takes the disparity, estimated from the stereo network, at full resolution and uses it in the layered depth of field technique described in Section 3.1. While the first part is supervised with the ground truth depth, the second stage is not learned. *Sequential aperture* ( $A_2$ ) is a variation of the first architecture where aperture supervision is used to train the network end-to-end from the blur module. A ground truth blurred image is used instead of the depth, and the loss is defined in the final image domain by applying a pixel-wise Euclidean loss. This is possible thanks to the differentiability of the refocusing algorithm. We use an image refocused with the ground truth disparity for supervision.

The third technique,  $B$ , leverages the fact that the cost volume of the stereo network provides a scene representation very similar to the layer decomposition used in the blurring algorithm. We use each slice of the cost volume (after the StereoNet ArgMin step) directly as a mask  $M^d$ . We note these slices are of low resolution (1/8 of the input) and therefore bilinearly upsample and refine them using a network with shared weights, up to the resolution required to apply the blurring convolution with a kernel of maximum size  $k_{max}$ . To train this network, we again use images blurred with the ground truth depth map and supervise with an  $L_2$  loss. A pretrained StereoNet is utilized for which the weights are frozen before the cost volume computation step. The refinement network uses the same blocks as in a refinement scale of ActiveStereoNet [56]. We call this method *cost volume refinement*.

Branch  $C$  depicts a *blur refinement* for which we propose to start from the  $\frac{1}{2}$  resolution depth map provided by a StereoNet intermediate step, blur the image at half resolution, and then upsample the blurred images back to full resolution using an upsampling akin to [56]. Once again, the network is trained with an  $L_2$  loss on a ground truth blurred image and the weights of the StereoNet part, up to the second refinement scale are frozen.

## 4. Experimental Evaluation

In the following section we provide qualitative and quantitative analysis of our approach on synthetic and real imagery using the public datasets SceneFlow [33], KITTI [13] and CityScapes [9]. All experiments are conducted on an *Intel(R) Core(TM) i7-8700* CPU machine at 3.20 GHz and we trained all neural networks until convergence using an *Nvidia GeForce GTX 1080 Ti* GPU with Tensorflow [1].

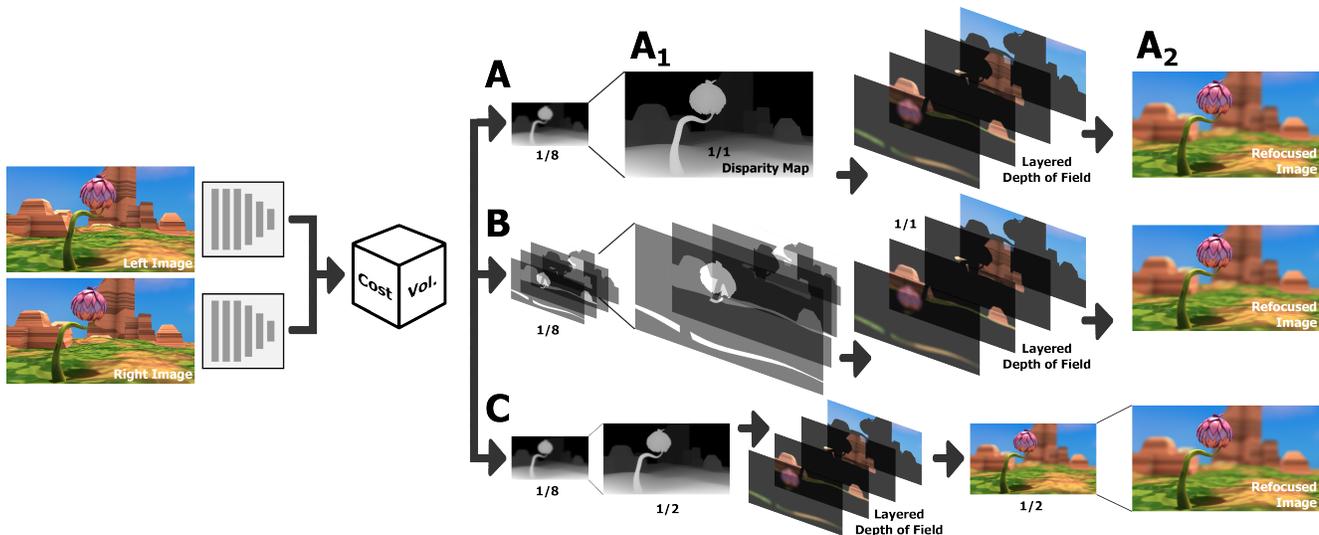


Figure 5: *SteReFo* architectures. A Siamese tower extracts deep features from a stereo pair which form a cost volume. Four instantiations of our pipeline are depicted. Branch **A**: the cost volume is then sequentially processed by a depth estimation and differentiable refocusing module and trained with disparity ( $A_1$ ) or aperture ( $A_2$ ) supervision. Branch **B**: cost volume refinement. Cost volume slices are adaptively upsampled and fed into the layered depth of field pipeline. Branch **C**: a low resolution depth map is used to predict a downsized refocus image which is consecutively upsampled.

**Qualitative Evaluation Metrics.** Comparing the quality of blurred images is a very challenging task. Barron et al. [4] propose to utilize structural metrics to quantify image quality with a light field ground truth. This modality is difficult to acquire and is therefore usually not present in real datasets of trainable size. Classical image quality metrics, like PSNR and SSIM, do not fully frame the perceptual quality of refocused images [6]. Because there is no consensus on what, quantitatively, makes for a good refocused images (bokeh-wise but also in terms of object boundaries and physical blur accuracy), subjective assessments are often used [19] and some papers exclusively focus on qualitative assessments [54, 52, 46, 51, 45]. In order to provide quantitative evaluation of our results, in addition to classical metrics, we propose to utilize a perceptual metric commonly used by the super resolution community [7, 6], the NIQE score [34].

In a first experiment, we use a synthetic dataset to train and test the four different approaches described in Section 3.3. In a second experiment we assess how our pipeline performs on real data.

#### 4.1. Architecture Comparison on Synthetic Data

We train the introduced approaches on the full *35mm driving* set of SceneFlow [33] and exclude 11 frames for testing. The virtual aperture and focal plane is fixed to  $a = 0.1$  and  $d^f = 100$ , while the disparity range is set to  $d \in [0, 300]$ . The maximum blur kernel size is  $k_{max} = 11$ .

In Fig. 6 we display the result of the forward pass on our test images and display a representative crop example in Fig. 7. Qualitatively we notice that, overall, the result of the sequential approach  $A_1$  outperforms the other three in terms of boundaries, bokeh appearance, and blur accuracy. The blur upsampling method  $C$  produces blurry output, even in areas that are intended to be sharp, and we observed a loss in the bokeh *circularity*. The cost volume refinement approach  $B$ , although a conceptually interesting idea, was found to introduce some high frequency artifacts in the blurred zones and also has generally lower quality boundaries. Finally, aperture supervision ( $A_2$ ) is by far the worst of the approaches, qualitatively, as we find high sensitivity to uniform areas in the image in addition to poor performance at object boundaries.

We further investigate the source of the quality drop in Fig. 8, where we compare the output of the depth using ground truth depth supervision and ground truth blurred images (i.e. aperture supervision). While the depth supervision retrieves disparity precision in particular along depth discontinuities, the supervision with aperture fails to recover small details and depth boundaries, ultimately destroying the depth map gradients.

**Quantifying the result.** The NIQE score [34] unifies a collection of statistical measures to judge the visual appearance of an image. We initially evaluate our introduced approaches for the test images numerically in Tab. 1 and analyze the absolute difference from the ground truth retrievals

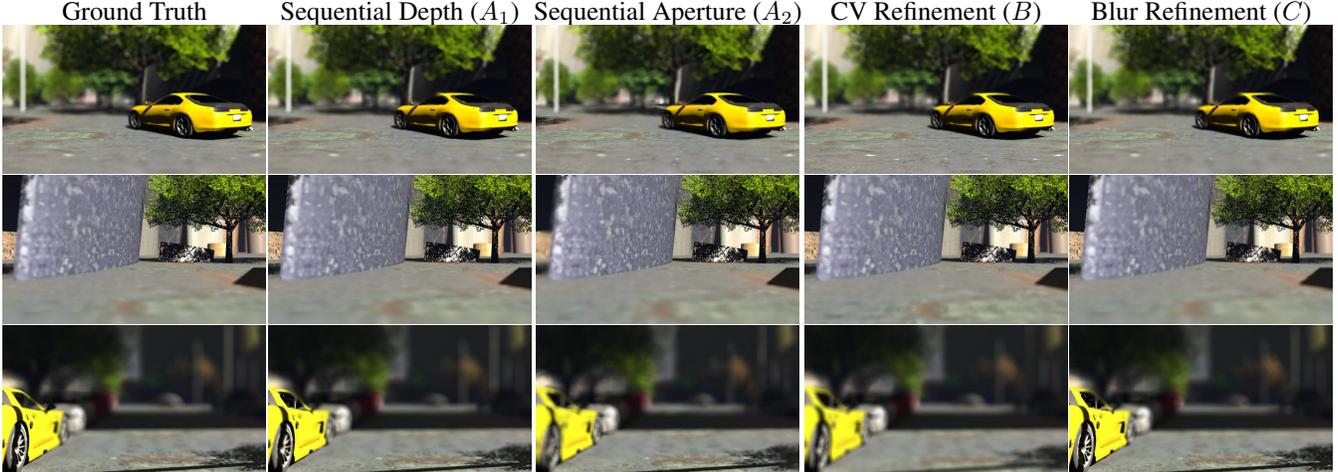


Figure 6: Comparison of the tested approaches. Each column correspond to one method and each row to one test image. We display on the very left column a ground truth image refocused using the provided ground truth disparity. While we invite the reader to zoom-in to see the details, more example images are included in the supplementary material.

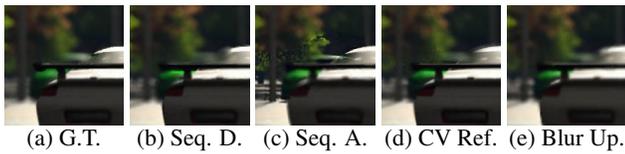


Figure 7: Crop on a representative artifact for the proposed methods. (a) is the ground truth, (b) the output of the sequential approach with depth supervision, (c) the sequential approach trained with aperture supervision, (d) refocusing from the cost volume, (e) the blur upsampling.



Figure 8: Depth map comparison. Left to right columns correspond to ground truth disparity, the disparity from depth supervision and from aperture supervision, respectively.

for a relative measure.

On inspection of this result, we observe that our sequential supervision with depth provides the best quantitative performance on all considered metrics which is in line with recent findings [15] that show artifact removal for simple depth estimation models. Approaches *B* and *C* are on par while the blur refinement *C* was found to have the highest (worst) relative score of 2.4 distance from the ground truth with a better structural similarity. While the NIQE score aids discovery of best performing methods for this problem ( $A_1$  vs. others), it is not well correlated with our visual judgment of

	Sequential	Seq. Apert.	CV Ref.	Blur Ref.
NIQE ↓	<b>6.5±1.4</b>	7.3±2.4	8.2±3.4	8.1±1.8
Rel. ↓	<b>0.1±0.04</b>	1.2±0.8	1.7±1.1	2.4±1.8
SSIM ↑	<b>0.98±0.01</b>	0.95±0.02	0.95±0.03	0.96±0.01
PSNR ↑	<b>39.16±1.1</b>	36.25±1.2	36.60±1.8	36.56±1.5

Table 1: Evaluation results for perceptual [34] and structural metrics. ↓ indicates that lower, ↑ that higher is better.

the aperture supervision result. We believe this is due to the fact that the aperture supervision image is indeed wrongly refocused, however, it does not show many high-frequency artifacts in contrast to the blur refinement which is also reflected in the classical metrics SSIM and PSNR.

**Discussion.** We believe our experimental work gives valuable insight into how the tasks of depth and refocusing can be entangled and the resulting benefits of doing so. Firstly, it suggest that depth supervision, and therefore high quality depth data, is essential for refocusing, even more so than retrieving images that are numerically close to the ground truth. This is quantitatively supported by the retrieved NIQE scores. Secondly, upsampling and refining the depth gives better quantitative and qualitative results than upsampling the blurred image. This suggests that the task of correcting the depth is superior to adjusting a blurred image with residual refinement, especially in the boundaries of in-focus objects.

Counter-intuitively, upsampling and filtering the cost volume reveals to be a difficult task, and while the results are still visually appealing, the high computational complexity makes this approach less tractable.



Figure 9: Experiments on real data. The region of interest is set with 2D tracking onto the number plate (middle) and the corresponding depth value is recovered. The left illustrates refocusing using a depth map generated by [14] while we show on the right the result of our sequential refocusing pipeline, together with the underlying disparity map.

## 4.2. Results on Real Data

The lack of a publicly available dataset for stereo-based refocusing approaches and the requirements of recent methods for additional information such as segmentation masks [51], varying aperture [45] or co-modalities given by dual-pixel sensing [50] and light fields [4] impede a standardized evaluation protocol. In order to assess how our approach performs on real data, we utilize that our pipeline does not require these additional cues and use the datasets proposed in [9] and [13]. We pick our best-performing approach, i.e. the sequential pipeline using depth supervision ( $A_1$ ) and pretrain on CityScapes [9] to perform static image refocusing (cf. Fig. 2) and refine with the coarse KITTI [13] ground truth for temporal evaluation.

To examine the efficiency of our pipeline, we apply SteReFo individually on consecutive frames of [13]. We utilize correlation-filter based 2D object tracking [49] to reason about the spatial location of objects of interest, per frame, prior to applying image refocusing. The 2D tracker provides state-of-the-art performance at high frame rates, a choice that makes a lightweight overall pipeline feasible. For the sake of comparison, we also retrained the monocular depth estimation approach in [14], and refocus the video using the generated depth as input for our layered refocusing pipeline.

For both approaches, we use disparity values of  $d \in [0, 80]$  and an aperture of  $a = 0.25$ , the focal plane is defined as the median value of the disparities inside the tracking bounding box, and  $k_{max} = 11$  (cf. Fig. 1). We compare the results directly in Fig. 9 and notice that blurring is significantly less consistent with respect to the scene geometry in the case of [14] compared to our approach. Indeed, the background is defocused as if it was not at infinity, the cars appear to reside in the same depth plane and the lower section of the car in the middle of the image is blurred (where it should not). The disparity maps for each approach support these observations. The interested reader is referred to the full video sequence <https://youtu.be/sX8N702uIag>.

FPS for $a =$	0.1	0.2	0.5	0.8
$k_{max} = \infty$	<b>76</b>	17	2	0.4
$k_{max} = 11$	<b>76</b>	<b>38</b>	<b>23</b>	<b>18</b>

Table 2: Runtime evaluation of the layered refocusing pipeline without ( $k_{max} = \infty$ ) and with ( $k_{max} = 11$ ) adaptive downsampling on images of [13]. Frames per second are evaluated by average of 10 runs.

**Timing.** We evaluate the timing of our approach on real data. The average runtime for the entire pipeline is 0.14 sec, including 0.11 sec to compute the disparity and 0.03 sec for refocusing. Tab. 2 shows how adaptive downsampling helps to reduce runtime complexity in particular for wider aperture values in contrast to naive refocusing, where the runtime grows exponentially with respect to the aperture size.

**Limitations.** The current approach has some limitations. The first one is inherent to all approaches relying on pyramidal depth estimation: small details that are lost at the lowest scale are difficult to recover at the upper scale which is why thin structures are problematic for our depth estimation (cf. the mirror of the truck in Fig. 9). The second observation we make is that StereoNet does not perform as well on real data as on synthetic data. Apart from the obvious difficulties (e.g., specularities, rectification errors, noise, optical aberrations) real data embeds, we also believe the sparse ground truth provided by the projected *Lidar* data used for supervision does not encourage the network to refine well at the object boundaries. Finally, the refocusing part suffers from the same problems as all image-based shallow DoF rendering techniques: it does not handle a defocus foreground very well. This is due to the fact that we miss occluded information when blurring from one view only.

## 5. Conclusion

The entanglement of stereo-based depth estimation and refocusing proves to be a promising solution for the task of efficient scene-aware image reblurring with appealing bokeh. Future improvements can address a fusion with segmentation methods to enhance boundary precision similar to [37] who entangle the task of semantic segmentation and depth estimation in real-time. The lightweight differentiable architecture with the insight about the value given by a sequential approach with depth supervision can be used for a variety of image and video refocusing applications in other vision pipelines that utilize our refocusing module. For instance in mobile applications, an image pair is taken at one point and different refocusing results may be calculated to help selection by a user afterwards, and the efficiency of our pipeline paves the way to real-time video editing applications on the edge.

## References

- [1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation*, pages 265–283, 2016. **5**
- [2] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski. Building rome in a day. In *2009 IEEE 12th International Conference on Computer Vision*, pages 72–79. IEEE, 2009. **2**
- [3] J. T. Barron. A general and adaptive robust loss function. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. **4**
- [4] J. T. Barron, A. Adams, Y. Shih, and C. Hernandez. Fast bilateral-space stereo for synthetic defocus. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015. **3, 6, 8**
- [5] B. A. Barsky and T. J. Kosloff. Algorithms for rendering depth of field effects in computer graphics. In *Proceedings of the 12th WSEAS International Conference on Computers*, pages 999–1010. World Scientific and Engineering Academy and Society (WSEAS), 2008. **3**
- [6] Y. Blau, R. Mechrez, R. Timofte, T. Michaeli, and L. Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *European Conference on Computer Vision*, pages 334–355. Springer, 2018. **6**
- [7] Y. Blau and T. Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237, 2018. **6**
- [8] H.-Y. Chou, K.-T. Shih, and H. Chen. Occlusion-and-edge-aware depth estimation from stereo images for synthetic refocusing. In *2018 IEEE International Conference on Multimedia & Expo Workshops*, pages 1–6, 2018. **3**
- [9] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. **1, 2, 3, 5, 8**
- [10] D. Eigen, C. Puhrsch, and R. Fergus. Depth map prediction from a single image using a multi-scale deep network. In *Advances in neural information processing systems*, pages 2366–2374, 2014. **2**
- [11] J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. *IEEE transactions on pattern analysis and machine intelligence*, 40(3):611–625, 2018. **2**
- [12] O. D. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence*, 2(03):485–508, 1988. **2**
- [13] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. **1, 2, 3, 5, 8**
- [14] C. Godard, O. Mac, A. Gabriel, and J. Brostow. UCL-Unsupervised Monocular Depth Estimation with Left-Right Consistency. In *CVPR*, page 7, 2017. **2, 8**
- [15] C. Godard, O. Mac Aodha, and G. Brostow. Digging into self-supervised monocular depth estimation. *arXiv preprint arXiv:1806.01260*, 2018. **3, 7**
- [16] X. Guo, H. Li, S. Yi, J. Ren, and X. Wang. Learning Monocular Depth by Distilling Cross-domain Stereo Networks. *arXiv preprint arXiv:1808.06586*, 2018. **2, 3**
- [17] P. Haeberli and K. Akeley. The accumulation buffer: hardware support for high-quality rendering. *ACM SIGGRAPH computer graphics*, 24(4):309–318, 1990. **3**
- [18] A. Handa, T. Whelan, J. McDonald, and A. J. Davison. A benchmark for rgb-d visual odometry, 3d reconstruction and slam. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1524–1531. IEEE, 2014. **2**
- [19] W. Hauser, B. Neveu, J.-B. Jourdain, C. Viard, and F. Guichard. Image quality benchmark of computational bokeh. *Electronic Imaging*, 2018(12):340–1, 2018. **1, 6**
- [20] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. **4**
- [21] G. Hinton, N. Srivastava, and K. Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Cited on*, 14, 2012. **4**
- [22] T. S. Huang and A. N. Netravali. Motion and structure from feature correspondences: A review. In *Advances In Image Processing And Understanding: A Festschrift for Thomas S Huang*, pages 331–347. World Scientific, 2002. **2**
- [23] J. Jiao, Y. Cao, Y. Song, and R. Lau. Look Deeper into Depth: Monocular Depth Estimation with Semantic Booster and Attention-Driven Loss. In *The European Conference on Computer Vision (ECCV)*, 9 2018. **3**
- [24] 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. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 66–75, 2017. **4**
- [25] S. Khamis, S. Fanello, C. Rhemann, A. Kowdle, J. Valentin, and S. Izadi. Stereonet: Guided hierarchical refinement for real-time edge-aware depth prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 573–590, 2018. **3, 4**
- [26] A. Knapitsch, J. Park, Q.-Y. Zhou, and V. Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics*, 36(4):78, 2017. **2**
- [27] M. Kraus and M. Strengert. Depth-of-field rendering by pyramidal image processing. In *Computer Graphics Forum*, volume 26, pages 645–654. Wiley Online Library, 2007. **3, 4**
- [28] J. Krivánek, J. Zara, and K. Bouatouch. Fast depth of field rendering with surface splatting. In *Proceedings Computer Graphics International 2003*, pages 196–201. IEEE, 2003. **3**
- [29] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *3D Vision (3DV), 2016 Fourth International Conference on*, pages 239–248. IEEE, 2016. **2**
- [30] N. Lazaros, G. C. Sirakoulis, and A. Gasteratos. Review of stereo vision algorithms: from software to hardware. *Inter-*

- national Journal of Optomechatronics*, 2(4):435–462, 2008. 2
- [31] M. Levoy and P. Hanrahan. Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, pages 31–42. ACM, 1996. 3
- [32] Z. Li and N. Snavely. MegaDepth: Learning Single-View Depth Prediction from Internet Photos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2041–2050, 2018. 2, 3
- [33] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016. 1, 2, 3, 5, 6, 11
- [34] A. Mittal, R. Soundararajan, and A. C. Bovik. Making a completely blind image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2013. 6, 7
- [35] R. Mur-Artal and J. D. Tardós. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017. 2
- [36] H. Nasse. Depth of field and bokeh. *Carl Zeiss camera lens division report*, 2010. 1
- [37] V. Nekrasov, T. Dharmasiri, A. Spek, T. Drummond, C. Shen, and I. Reid. Real-time joint semantic segmentation and depth estimation using asymmetric annotations. *arXiv preprint arXiv:1809.04766*, 2018. 8
- [38] H. Nguyen. *Gpu gems 3*. Addison-Wesley Professional, 2007. 3
- [39] M. Pharr, W. Jakob, and G. Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016. 3
- [40] M. Poggi, F. Tosi, and S. Mattoccia. Learning monocular depth estimation with unsupervised trinocular assumptions. *arXiv preprint arXiv:1808.01606*, 2018. 2
- [41] D. Präkel. *The visual dictionary of photography*. Ava Publishing, 2010. 1
- [42] A. Robison and P. Shirley. Image space gathering. In *Proceedings of the Conference on High Performance Graphics 2009*, pages 91–98. ACM, 2009. 3
- [43] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International journal of computer vision*, 47(1-3):7–42, 2002. 2
- [44] N. Smolyanskiy, A. Kamenev, and S. Birchfield. On the Importance of Stereo for Accurate Depth Estimation: An Efficient Semi-Supervised Deep Neural Network Approach. *arXiv preprint arXiv:1803.09719*, 2018. 3
- [45] P. P. Srinivasan, R. Garg, N. Wadhwa, R. Ng, and J. T. Barron. Aperture supervision for monocular depth estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6393–6401, 2018. 4, 6, 8
- [46] P. P. Srinivasan, T. Wang, A. Sreelal, R. Ramamoorthi, and R. Ng. Learning to synthesize a 4d rgbd light field from a single image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2243–2251, 2017. 4, 6
- [47] B. Tippetts, D. J. Lee, K. Lillywhite, and J. Archibald. Review of stereo vision algorithms and their suitability for resource-limited systems. *Journal of Real-Time Image Processing*, 11(1):5–25, 2016. 2
- [48] A. Tonioni, F. Tosi, M. Poggi, S. Mattoccia, and L. Di Stefano. Real-time self-adaptive deep stereo. *arXiv preprint arXiv:1810.05424*, 2018. 3
- [49] J. Valmadre, L. Bertinetto, J. Henriques, A. Vedaldi, and P. H. S. Torr. End-to-end representation learning for correlation filter based tracking. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 8
- [50] N. Wadhwa, R. Garg, D. E. Jacobs, B. E. Feldman, N. Kanazawa, R. Carroll, Y. Movshovitz-Attias, J. T. Barron, Y. Pritch, and M. Levoy. Synthetic depth-of-field with a single-camera mobile phone. *ACM Transactions on Graphics (TOG)*, 37(4):64, 2018. 1, 4, 8
- [51] L. Wang, X. Shen, J. Zhang, O. Wang, Z. Lin, C.-Y. Hsieh, S. Kong, and H. Lu. Deeplens: shallow depth of field from a single image. In *SIGGRAPH Asia 2018 Technical Papers*, page 245. ACM, 2018. 1, 3, 4, 6, 8
- [52] Y. Yang, H. Lin, Z. Yu, S. Paris, and J. Yu. Virtual dslr: High quality dynamic depth-of-field synthesis on mobile platforms. *Electronic Imaging*, 2016(18):1–9, 2016. 3, 6
- [53] Z. Yin and J. Shi. GeoNet: Unsupervised Learning of Dense Depth, Optical Flow and Camera Pose. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, 2018. 3
- [54] Z. Yu, C. Thorpe, X. Yu, S. Grauer-Gray, F. Li, and J. Yu. Dynamic depth of field on live video streams: a stereo solution. In *Proc. of the 2011 Computer Graphics Int. Conf., CGI 2011*. Citeseer, 2011. 3, 6
- [55] H. Zhan, R. Garg, C. S. Weerasekera, K. Li, H. Agarwal, and I. Reid. Unsupervised Learning of Monocular Depth Estimation and Visual Odometry with Deep Feature Reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 340–349, 2018. 3
- [56] Y. Zhang, S. Khamis, C. Rhemann, J. Valentin, A. Kowdle, V. Tankovich, M. Schoenberg, S. Izadi, T. Funkhouser, and S. Fanello. Activestereonet: End-to-end self-supervised learning for active stereo systems. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 784–801, 2018. 3, 4, 5
- [57] C. Zhao, L. Sun, P. Purkait, T. Duckett, and R. Stolkin. Learning monocular visual odometry with dense 3D mapping from dense 3D flow. *arXiv preprint arXiv:1803.02286*, 2018. 3
- [58] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe. Unsupervised learning of depth and ego-motion from video. In *CVPR*, page 7, 2017. 3
- [59] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017. 4
- [60] Y. Zou, Z. Luo, and J.-B. Huang. DF-Net: Unsupervised Joint Learning of Depth and Flow using Cross-Task Consistency. *arXiv preprint arXiv:1809.01649*, 2018. 3

## 6. Additional Results

In addition to Figure 6 found in the main paper, Figure 10 shows further results on the SceneFlow dataset [33] for the different pipelines,  $A_1$ ,  $A_2$ ,  $B$  and  $C$ , which we propose.

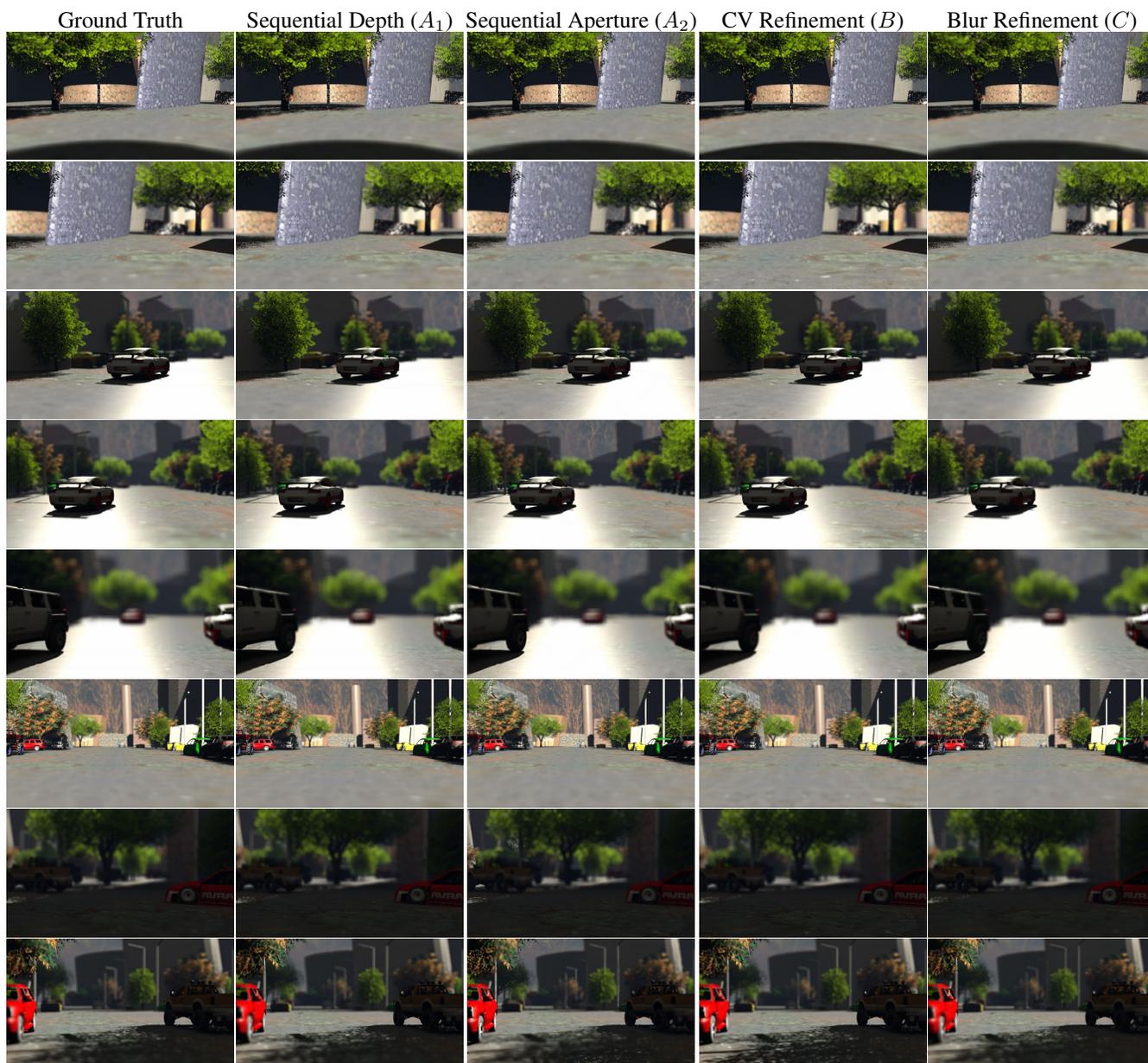


Figure 10: Comparison of the four different approaches for the rest of the test set. Each column corresponds to one method and each row to one test image. We display on the very left column the ground truth image refocused using the provided ground truth disparity. We invite the reader to zoom-in to see the details.