Saliency-aware Stereoscopic Video Retargeting

Hassan Imani[1], Md Baharul Islam[1,2], Lai-Kuan Wong[3]
[1]Bahcesehir University [2]American University of Malta [3]Multimedia University
hassan.imani1987@gmail.com, bislam.eng@gmail.com, lkwong@mmu.edu.my

Abstract

Stereo video retargeting aims to resize an image to a desired aspect ratio. The quality of retargeted videos can be significantly impacted by the stereo video's spatial, temporal, and disparity coherence, all of which can be impacted by the retargeting process. Due to the lack of a publicly accessible annotated dataset, there is little research on deep learning-based methods for stereo video retargeting. This paper proposes an unsupervised deep learning-based stereo video retargeting network. Our model first detects the salient objects and shifts and warps all objects such that it minimizes the distortion of the salient parts of the stereo frames. We use 1D convolution for shifting the salient objects and design a stereo video Transformer to assist the retargeting process. To train the network, we use the parallax attention mechanism to fuse the left and right views and feed the retargeted frames to a reconstruction module that reverses the retargeted frames to the input frames. Therefore, the network is trained in an unsupervised manner. Extensive qualitative and quantitative experiments and ablation studies on KITTI stereo 2012 and 2015 datasets demonstrate the efficiency of the proposed method over the existing state-of-the-art methods. The code is available at https://github.com/z65451/SVR/.

1. Introduction

3D video technology is growing in popularity due to the rising demand for augmented and virtual reality (AR/VR) devices used in various applications, e.g., mobile phones, autonomous vehicles, and robots. As 3D videos can be viewed on display devices with varying aspect ratios, stereo image and video retargeting techniques are becoming increasingly important for modifying aspect ratios of media content to correspond to those of target screens and devices. Stereo video retargeting aims to convert a stereo video to the desired aspect ratio. Notably, changes in the aspect ratios of videos could result in spatial distortion, and temporal inconsistency, such as jittering and flickering. Content distortion can be even more severe for stereo videos if depth preservation is not considered during retargeting. Changes in the depth of salient objects can negatively affect the 3D viewing experience [19]. The efficacy of stereo video retargeting approaches depends mainly on the ability to discern between salient and non-salient regions.

In traditional approaches, the stereo image and video retargeting problem is formulated as a constrained optimization problem. [2], [16] and [11] proposed discrete approaches that extends 2D pixel fusion methods for 3D image retargeting. [2] performs seam-searching by considering depth energy and appearance energy, while in [16], seam selection and seam matching are considered simultaneously to maintain the relationship between objects and disparity. Hu et al. [11] combine the occluding masks with the energy optimization of pixel fusion. [17] introduced a depth-preserving stereo image retargeting technique, a continuous approach. Shao et al. [23] described a Quality of Experience(QoE)-guided warping strategy in response to the effect of QoE on visual attributes.

Kopf et al. [14] proposed one of the first stereo video retargeting methods to preserve salient frame content, avoid flickering, and maintain stereo consistency. Liu et al. [20] formulate distortion energies to prevent significant areas of the videos from deforming. In [13], volume warping with non-homogeneous scaling optimization resizes the stereoscopic video. During the warping, the depth is remapped using a depth remapping constraint and a saliency constraint that protects the salient regions. Temporal and depth constraints are considered in [18, 19]. Li et al. [19] proposed a method based on depth fidelity constraint. To reduce conflicts between depth, shape, and temporal constraints and prevent perceptually degrading temporal coherence, Li et al. [18] loosen temporal constraints for non-paired regions at frame boundaries. More recently, Wang et al. [31] presented a depth trajectory-aware stereoscopic video retargeting technique by optimizing the spatial location and depths, along with a temporal depth distortion energy to preserve the depth trajectory in the temporal direction.

Driven by the proven performance of deep learning in many computer vision tasks, some researchers employed deep neural networks for stereo image retargeting. [7] and

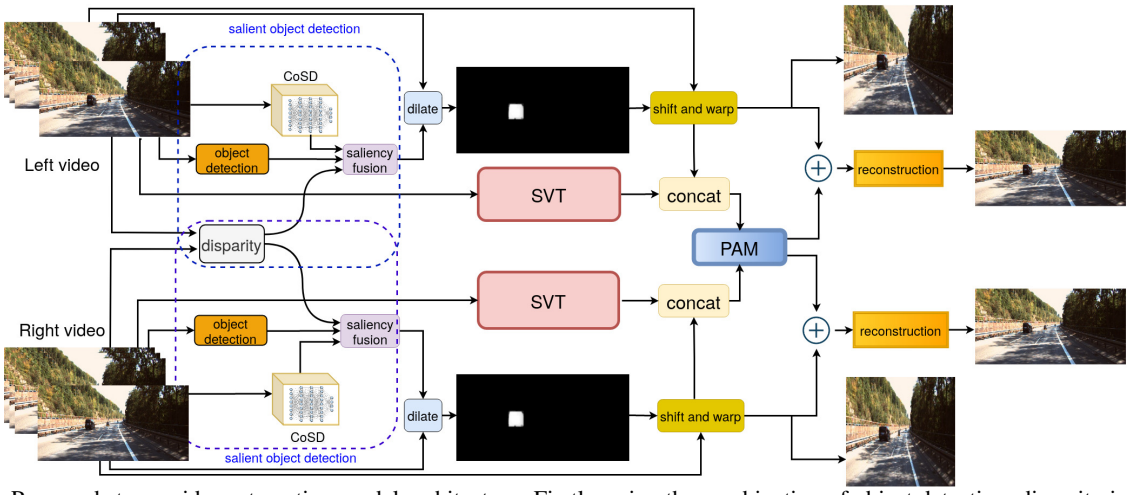

Figure 1. Proposed stereo video retargeting model architecture. Firstly, using the combination of object detection, disparity information, and Co-Saliency detection (CoSD), the salient areas of the frames are detected and segmented. The stereo video Transformer SVT helps with attention generation. Then, the middle frame is shifted and warped based on the salient regions. The PAM module uses the cross-view information, and a reconstruction block generates the input middle frame.

[8] proposed convolutional neural network (CNN)-based models to estimate the disparity, which is then utilized to assist in salient objects detection. Fan et al. [8] created a cross-attention extraction method to build an attention map, and a disparity-assisted 3D importance map preservation module is used to calculate the depth information. Fan et al. [7] proposed two loss functions for training an unsupervised retargeting model; the view synthesis loss guarantees the generation of high-quality stereoscopic images with inter-view correspondences, and the stereo cycle consistency loss that preserves the structure and prevents disparity variations. However, the local receptive fields of plain CNN make it difficult to capture correspondence with large disparities [29]. To overcome this limitation, Wang et al. [29] integrated epipolar constraints with an attention mechanism to estimate feature similarities along the epipolar line and proposed PAM to handle different stereo frames with extreme disparity changes to cope better with large disparity changes. To our best knowledge, no research attempted the deep learning approach for stereo video retargeting.

This paper proposes an unsupervised deep learning-based method for stereo video retargeting. Identifying significant stereo video content is essential to retargeting process. In our approach, we devise a salient object detection scheme that fuses the output of the saliency and object detection models to segment the important content of the stereo video accurately. To resize the video to the target aspect ratio, we shift and warp the salient content based on the loss of each pixel's shift using a 1D convolutional layer. We also design the Stereo Video Transformer. Finally, we re-create the input stereo video frames using cross-view information from the parallax attention mechanism (PAM) [30] and propagate the loss to train our model without supervision. Our main contributions are listed as follows:

- A novel unsupervised model for stereo video retargeting. By re-creating the input stereo video frames from the retargeted ones, we use the input frames as labels and train the model completely unsupervised.
- A shifting layer that uses convolution and warping for retargeting the video frames.
- A Stereo Video Transformer with self-attention and a sequence of spatial, temporal, and disparity tokens extracted using the stereo patch embedding method.
- A loss function that combines the spatial, temporal, and disparity losses to guide the model to obtain a more consistent retargeted stereo video.

2. Proposed Method

The architecture of the proposed method for stereo video retargeting is shown in Fig. 1. The input to the framework is a batch of n consecutive left and right frames. First, salient objects are detected using disparity information, Co-Saliency detection (CoSD), and object detection techniques. Then, a dilation operation is applied to expand the salient areas, and the shift-and-warp operation is employed to move the objects in the frames to the appropriate location, given the aspect ratio. To integrate attention to the model, the proposed stereo video Transformer (SVT) factorizes the input video's spatial, temporal, and depth channels, and its results are concatenated with the warped frames. The PAM module then uses the cross-view information to fuse both views. In the reconstruction part, using convolutional blocks, the objects are relocated to their location within the original aspect ratio, re-generating the input frames, which is then used to calculate and minimize the loss in the training phase.

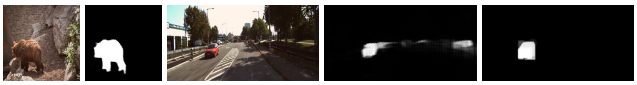

Figure 2. Results of saliency detection on the Davis [22] and KITTI stereo 2015 [21] datasets. From left to right: image from Davis, its segmentation with CoSD, image from KITTI stereo 2015, its segmentation with CoSD, segmentation with fusion.

2.1. Salient Object Detection

Detecting salient objects as accurately as possible in a stereo video is an essential step of our model; otherwise, primary object deformation will likely occur. Deep neural networks have primarily been trained independently for related tasks such as segmentation and salient object detection, without capitalizing on the inter- and intra-feature cues for a collection of sequential video frames, which may potentially improve the accuracy of object extraction.

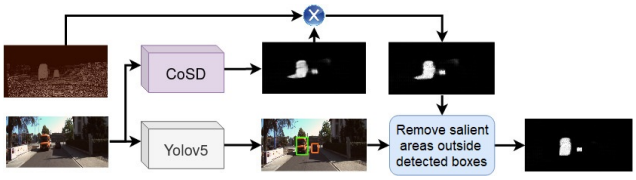

Figure 3. Fusion of disparity information, CoSD, and Yolov5 object detection to get better masks for the salient objects.

Recently, Co-Saliency detection (CoSD), which finds the common salient objects among an image group, is preferred over the normal saliency detection (SD) methods for many computer vision tasks. CoSD discriminate co-occurring objects over consecutive frames [6] considering other objects in the scene, and both intra-class compactness and inter-class distinctness are maximized simultaneously. Inspired by Su et al.'s [26] unified framework that jointly detects salient objects and performs segmentation, we adopt their CoSD component and combine it with object detection and disparity information to locate the essential areas of a stereo video more accurately. The CoSD module contains a transformer block that treats the input frame features as patch tokens and then uses the self-attention technique to extract their long-range dependencies. The network then uses these dependencies to determine the patch-structured similarities between the relevant components. A self-mask is generated using an intra-multi-layer perceptron (MLP) learning module to strengthen the network and prevent partial activation. However, when the camera moves, CoSD alone does not perform well as it fails to detect all salient objects, and some parts of the scene which are not salient are detected as salient areas (see Figure 2). To solve this problem, we propose a fusion strategy that combines CoSD, Yolov5 [28] object detector, and depth cues from disparity map to generate a more accurate saliency map. The saliency

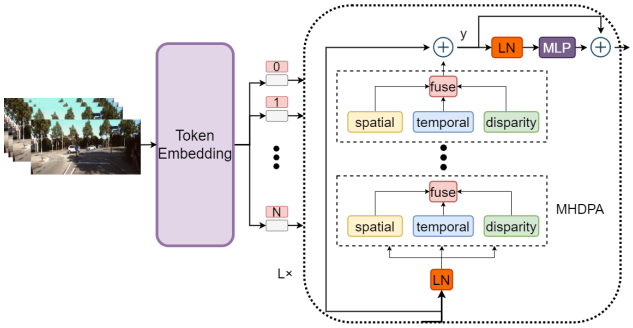

Figure 4. Stereo video Transformer architecture. We factorize all encoder parts into spatial, temporal, and disparity channels.

fusion block is shown in Figure 3. We first detect the salient objects with CoSD and fuse its results with the disparity map. Then, we apply Yolov5 to detect the bounding box for each object and remove the salient scene outside of these bounding boxes. This way, we obtain clean salient objects.

2.2. Stereo Video Transformer

Attention-based frameworks are rational for modeling long-range contextual relations in the video. Inspired by Vision Transformer (ViT) [5], which uses a multi-head self-attention mechanism, we propose Stereo Video Transformer (SVT). This model factorizes the stereo video's spatial, temporal, and disparity channels to cope with the long token sequences present in stereo videos. The proposed SVT architecture is shown in Figure 4.

The Transformer has a flexible architecture that works on the provided tokens. ViT [5] extracts N patches $f_i^{h\times w}$ from each video frame $F^{H\times W}$ and apply a linear projection to convert them to d-dimensional tokens. The tokens are then passed to the Transformer encoder of L layers, where each layer l contains a multi-head dot-product attention (MHDPA), a layer normalization (LN), and a multi-layer perceptron (MLP). Let V=$\mathbf{V}^{T\times H\times W\times C}$ represent the left or right stereo video. For each patch of size $t\times h\times w$ in the lth layer, it is mapped to a sequence of tokens, $\mathbf{tokens}_l^{n_T\times n_H\times n_W\times d}$, where $n_t=[T/t]$, $n_h=[H/h]$, and $n_w=[W/w]$, and d is the token's dimension. Instead of sampling n_t video frames simply as proposed by ViT [5] and concatenating them together independently for the consecutive frames, we propose the stereo patch embedding. The positional embedding is added to each input token in the same way as the original ViT [5]. The key difference is that the stereo video has more tokens than the pre-trained image model. Therefore, we initialize the positional embeddings by repeating them temporally.

Stereo patch embedding. In addition to spatial and temporal information, depth cues are included for tokenizing the stereo video in the form of disparity information. Figure 5 illustrates the extraction of spatial, temporal, and disparity patches from the stereo video. For each patch of

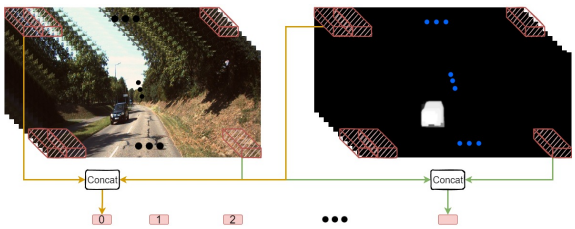

Figure 5. Patch embedding for stereo video. We extract the spatial, temporal, and disparity patches from each left and right video.

size $t \times h \times w$ with the same disparity size, we first extract $n_t \times n_h \times n_w$ tokens from the temporal and spatial dimensions. The same process is then performed on the disparity channel. Differing from the original ViT [5] where the tokens of the temporal data are combined inside the Transformer encoder, the disparity tokens are fused to the spatial and temporal tokens before feeding them into the Transformer encoder.

Spatial, temporal, and disparity self-attention. We individually calculate attention weights for every token over the spatial, temporal, and disparity channels at various heads. For each head, the attention is as the following:

$$Attn(Q, K, V) = Soft(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

where *Attn* and *Soft* refer to the attention and SoftMax, respectively, and the query $Q = VW_q$, key $K = VW_k$, and value $V = VW_v$ are the projection of the left or right video *V*. The primary concept is to create $(K_s, V_s)^{n_h n_w d}$, $(K_d, V_d)^{n_h n_w d}$, and $(K_t, V_t)^{n_t d}$ for spatial, disparity, and temporal indices, respectively, and then adjust the keys and values for each query such that they only look for tokens from the same index. Finally, we concatenate the outputs of different heads with a linear projection.

2.3. Shift and Warp

To map the pixels from the left and right source to the left and right target frames, we need to shift the pixels based on the computed saliency map. Additionally, we should be aware of the non-salient regions of the frames and properly warp them to avoid the deformation of the non-salient objects. Before shifting, we dilate the salient areas of the frames to recover parts of the salient object that could have been missed. We first apply a Gaussian blur to the saliency map and then use a 2D convolution with a kernel size of 11×11 for dilation. This size is selected experimentally. We apply this method to each channel. We then shift the pixels as below:

$$F_{trg}(x, y) = F_{src}(x + shift(x, y), y) \tag{2}$$

where F_{trg}, F_{src}, *and shift* are the retargeted frame, source frame, and the amount of the shift applied to each pixel based on the saliency map of the source frame, respectively. The shifting process should constrain the salient regions to be kept as rigid as possible to preserve the salient content. Additionally, pixels in the same columns should experience comparable shifts to keep the entire structure of the objects in the frame and prevent deformation on the shape of the main objects. Therefore, we use a 1D convolution to restrict the shape of a salient area to be consistent along the column axis. The kernel size for this convolution is k=(fr_{height}, 1), where fr_{height} is the height of the frame:

$$S1(x, y) = Conv(F_{src}(x, y), k) \tag{3}$$

Next, we calculate the summation of the elements on the y-axis and then tile the elements on the y-axis with dim = (1,fr_{height},1,1) to get the salient columns:

$$S2(x, y) = Tile(Sum(S1(x, y)), dim) \tag{4}$$

where *Sum* is the summation of elements in the y-axis, and *Tile* constructs a tensor by repeating the elements in the x-axis. The final shift of each pixel based on the saliency map is calculated as the weighted addition of *S1* and *S2*:

$$shift(x, y) = \alpha S1 + \beta S2 \tag{5}$$

where α and β are experimentally set to 1.9 and 1.

Finally, an input frame is warped into the desired aspect ratio using Eq. (2). Four adjacent pixels are interpolated since the shifting map has sub-pixel accuracy.

2.4. Parallax Attention Mechanism

Based on self-attention approaches [9, 32], Wang et al. [30] introduced the parallax attention mechanism (PAM) to determine matching in stereo images. PAM effectively merges the characteristics of the left and right image pair. The PAM structure has been modified in [12] to make it suitable for video-based inputs. After applying a 1×1 convolutional layer, the attention mappings from the left to right and vice versa are built using batch-wised matrix multiplication in a SoftMax block. Features for the left disparity are then merged with the corresponding right features at all disparity levels. To generate more features and increase learning capacity, we use the method in [12], in which 2D CNNs are applied to the output features, followed by a ReLU and a batch normalization (BN) layer. Three convolution layers are used with 128, 128, and 64 filters.

2.5. Reconstruction

After applying the PAM module, we re-generate the input stereo video frames. For this purpose, we use 5 2D CNN blocks, with 64, 128, 512, 128, and 3 output filters, respectively. We use the last layer of this block for loss calculation. The first layer accepts the addition of the outputs of

PAM and shifted and warp modules and produces a feature map of size 64. Its kernel size is 5. The other convolution layers use a kernel size of 3. A stride and padding of 1 and ReLU function are used for each convolutional layer.

2.6. Loss Functions

Pixel-based metrics such as L2 or logistic regression are often utilized to determine the loss between the source and recreated frames. However, pixel-based loss functions may not accurately represent the subjective difference and spatial relationship between two consecutive frames. For instance, a similar frame that has been moved a few pixels may not substantially impact human perception, but its pixel-by-pixel loss can be severe.

We combine four loss functions for training the proposed model on the KITTI stereo 2012 [10] and 2015 [21] datasets. The first loss computes the difference between the source and the retargeted frames. The second loss computes the dissimilarities between the source and output of the reconstruction module. The third loss measures the difference between the disparity of the source and retargeted frames. Please note that the aspect ratio of the source and retargeted frames are different. For example, for 50% resizing, the source frames are with size 224×448, and the retargeted frames are 224×224.

The first loss computes the difference between the VGG19 [25] features extracted from the source and retargeted frames. Specifically, we use VGG19 features of layers conv1_2, conv2_2, conv3_3 feature before the ReLU activation layer:

$$L_{VGG19} = MSE(VGG19_{src} - VGG19_{ret}) \quad (6)$$

where *MSE* denote the mean square error. The total VGG19 features loss is computed as the summation of the feature difference (1) between the entire frames of source and retargeted frames, and (2) between their salient regions:

$$L^{total}_{VGG19} = L^{entire}_{VGG19} + L^{salient}_{VGG19} \quad (7)$$

The second loss term computes the frequency domain differences by computing the MSE between the forward and inverse 2D discrete wavelet transform (DWT) decompositions between source and retargeted frames:

$$\begin{aligned} L_{DWT} = MSE(FDWT_{src} - FDWT_{ret}) \\ + MSE(IDWT_{src} - IDWT_{ret}) \end{aligned} \quad (8)$$

where *FDWT* and *IDWT* denote the forward and inverse 2D DWT decompositions, respectively. L_{DWT} is calculated as the average loss of the left and right frames.

The final loss functions are the photometric L_p and smoothness L_s losses [29] respectively. The photometric loss includes a mean absolute error (MAE) loss and a structural similarity index (SSIM) loss term. The photometric loss is defined as follows:

$$L_p = \frac{1}{N} \sum_{p \in V_l} \gamma \frac{1 - S(I_l(p), \hat{I}(p))}{2} + (1 - \gamma)||I_l(p), \hat{I}(p)|| \quad (9)$$

where $\hat{I}$ is the warped version of the right frame. S is the SSIM operator, p indicates a valid pixel covered by the valid mask, N is the number of valid pixels, and γ is a constant.

The smoothness loss is an edge-aware loss that encourages local smoothness of the disparity values:

$$\begin{aligned} L_s = \frac{1}{N} \sum_{p} (||\nabla_x \hat{D}_r(p)||e^{-||\nabla_x \hat{I}_l(p)||} + \\ ||\nabla_y \hat{D}_r(p)||e^{-||\nabla_y \hat{I}_l(p)||}) \end{aligned} \quad (10)$$

where ∇ is the gradient operator. The final loss function is the union of the losses above:

$$loss = L^{total}_{VGG19} + \alpha L_{DWT} + L_s + L_p \quad (11)$$

where α is the regularization term empirically set to 0.05.

3. Datasets and Experiments

Datasets: Due to the nature of the SVR, no publicly available dataset is specifically designed for the SVR task. Some works, such as citeli2020perceptual, used videos from commercial 3D movie films for experiments. The difference between our method and [18] (e.g., not a learning-based method) is the training phase that we have. Based on the available existing stereo video datasets, we chose the KITTI stereo 2012 [10] and 2015 [21] datasets because of the large disparity range between foreground and background objects with significant temporal disparity changes. The video scenes are dynamic, and the camera is moving. However, our method can also retarget the commercial stereo video (single-scene) to the target aspect ratio. The KITTI stereo 2012 dataset contains 194 training frame pairs and 195 test frame pairs. There are 200 training and 200 test sequences in the KITTI stereo 2015 benchmark (4 frames per scene). Since the disparity values are published only for the training sets, we divide their training sets into the train-test sets with an 80:20 split ratio and use them for the experiments.

Experiments: The specification of the computer system used for our experiments is Intel i7-10875H, 64GB memory, Nvidia RTX3090 24GB. We trained our model with ADAM optimizer, learning rate initialized to 0.05 and the model waas trained with 4000 iterations. The training took 2 days to complete on our RTX3090 GPU.

Evaluation Criteria: We use qualitative and quantitative comparisons to evaluate the proposed method. For qualitative and quantitative studies, we compare our method with 4 other methods: linear scaling, manual cropping, fast video [4], and seam carving [1] methods. We cannot compare our results with the stereo video retargeting techniques [13, 14, 19, 20] due to the unavailability of codes, including two more recent 2D video retargeting methods [15, 27]. For quantitative comparisons, we use three metrics. The first one is the bidirectional similarity metric [24], a frequently used metric in the image and video retargeting. We compare different retargeting methods for the second metric based on the perceptual distance between the source and retargeted stereo video's VGG19 [25] features. In [19], an objective metric named Disparity Distortion ratio (DDr) is proposed to quantify the spatial and temporal depth distortion. We use DDr to compute the mean of the disparity variation of pixels between the retargeted and original videos, normalized to the disparity range as follows:

$$DDr = \frac{1}{|d_{max}| \times H \times W \times T} \sum (D - \tilde{D}) \qquad (12)$$

where d_{max} is the maximum disparity in the source stereo video, H, W, and T are the dimensions of the video, and D and $\tilde{D}$ is the disparity maps of the source and retargeted videos, respectively.

4. Results and Discussion

4.1. Computational Performance

Table 1 compares the computational complexity of our method to that of other methods for a single pair of frames by computing the sum of the running times for the left and right frames. All methods are implemented on an Intel i7-10875H workstation with Nvidia RTX3090 GPU. We can observe that our proposed method achieved the fastest speed, about 30x and 2x times faster than Seam carving [1] and Fast video [4] methods, respectively.

Table 1. Comparison of computational complexity.

method	Seam carving [1]	Fast video [4]	Ours (GPU)
complexity(s)	42.184	3.014	1.831

4.2. Qualitative Results

We randomly select stereo video sequences from KITTI stereo 2012 [10] and 2015 [21] test sets for our qualitative comparisons. We provide the qualitative results for 3 aspect ratios: reduction of horizontal video size by 20%, 30%, and 50% respectively. For example, the aspect ratio of 20% means that the size of the original video frames is reduced by 20% horizontally. Figure 6 compares the proposed method's results with 4 methods for 50% aspect ratio on 3 randomly selected videos from the KITTI stereo 2015 [21] test set. Each row belongs to one left frame of one of the videos in the dataset. Since both the left and right views need more space, we report the results based on the left frames and provide all results in the supplementary materials. Each stereo video contains one main object (foreground) and the background. Since KITTI stereo 2012 [10] and 2015 [21] datasets mostly contain the car videos, the foreground object of the stereo videos includes the cars. It is apparent from visual results in this figure that our method can preserve both the salient object and the background well. Noticeably, the main object size is resized less than the background.

Figure 7 shows the retargeting results with horizontal size reduction of 30% and 20%. These visual results demonstrate that our method is superior to the other methods. These two aspect ratios require a lesser shift of pixels. Thus, the main structure of the video frames is well-preserved by all methods. When we resize the frames with a higher reduction in size, e.g., 50%, shape deformation of objects is observed in some methods. Figure 8 illustrates the results of retargeting one video from KITTI stereo 2012 [10] dataset with horizontal video size reduction of 50%, 20%, and 150% (enlarge). It is apparent in these results that our method can effectively perform stereo retargeting in all cases, from very extreme (50% and 150%) to lower resizing ratio (20%) cases. In addition, this figure's results demonstrate our method's ability to enlarge the frames.

4.3. Quantitative Results

We use two methods for computing the similarity between two frames in our quantitative comparisons. The first method is the bidirectional similarity metric [24], the most widely accepted criteria for assessing the video retargeting quantitative performance [3]. When evaluating the quality

Table 2. Comparison of the bidirectional similarity metric [24]. The results are the average values for the left and right frames. Videos #1 and #2 are taken from KITTI 2015 [21], and videos #3 and #4 from 2012 [10] datasets. The last column shows the average bidirectional similarity.

Video No.	**#1**	**#2**	**#3**	**#4**	**Avg**
Manual cropping	5.065	4.950	2.170	4.320	4.126
Seam carving [1]	3.347	2.693	3.822	3.011	3.218
Fast video [4]	3.211	2.901	3.736	2.882	3.182
Ours	**2.190**	**1.970**	**2.588**	**1.870**	**2.154**

Table 3. Comparison of the similarity between the input and retargeted videos based on VGG19 [25] feature difference.

Video No.	**#1**	**#2**	**#3**	**#4**	**Avg**
Manual cropping	1.140	1.374	1.016	0.888	1.104
Seam carving [1]	0.595	1.137	0.796	0.776	0.826
Fast video [4]	0.592	0.940	0.743	0.690	0.741
Ours	**0.326**	**0.739**	**0.721**	**0.678**	**0.616**

Figure 6. Qualitative results of stereo video retargeting on randomly selected left video frames from the KITTI stereo 2015 [21] (top) and 2012 [10] (bottom) datasets for reducing the horizontal video size at 50%. Left to right: original frame, linear scaling, manual cropping, seam carve [1], fast video [4], and ours.

of the retargeted video, bidirectional similarity looks at the coherence and completeness between the source and retargeted frames. Completeness assesses the impairment in the shape of the retargeted objects relative to the objects in the source frames. In contrast, coherence measures the deformity that occurs when an area that does not exist in the original frames appears in the retargeted frames.

Figure 7. Qualitative results of retargeting on randomly selected frames from the KITTI stereo 2015 [21] dataset for 30% (first row) and 20% (second row) reduced the horizontal size. From left to right: input frame, LS, seam carve [1], fast video [4], and Ours.

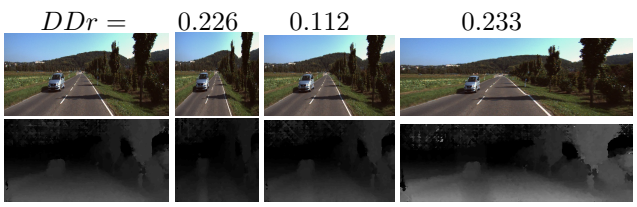

Figure 8. Different retargeting results with respective depth maps. Left to right: Input video frame and their retargeted results with horizontal size reduction at 50%, 20%, and 150%($enlarge$).

Table 2 compares the bidirectional similarity between the source and target videos. A total of 4 videos are used for this study. A lower value represents better shape preservation or less object deformation during the retargeting. Our method achieves the best results in terms of bidirectional similarities for all of the videos. A value of 1.870 for video 4 shows the retargeted video is very similar to the source.

Next, we compare the deep features between the source and retargeted stereo video frames. For this purpose, we compute the difference between the VGG19 [25] features extracted from the source and target frames, respectively. Table 3 depicts the quantitative comparison of the VGG19 features. Video #1 obtained the best results of 0.3262, indicating that the retargeted stereo video is similar to its source.

To evaluate depth preservation, we compute the depth distortion with the DDr metric for the video shown in Figure 8. The DDr results are reported in the figure for each aspect ratio, together with the illustration of the corresponding disparity maps. It can be observed that the distortion is lower for smaller size reduction (size reduction of 30%, 20%), as compared to more extreme cases (50% and 150%). For horizontal size reduction of 20%, a low distortion ratio of $DDr = 0.112$ is reported.

4.4. Ablation Study

The key idea of the proposed framwork is to preserve the salient regions during the retargeting process. With-

Table 4. Ablation study. Comparison of the similarity between the input and retargeted videos based on the VGG19 [25] features. The results are for 3 cases: without (*w/o CoSD*) CoSD saliency detection, without (*w/o Trans*) Transformer block, and will all of the blocks (*with all*). The best results are shown in **bold**.

Video No.	#1	#2	#3	#4	Avg
w/o CoSD	0.8755	0.9400	1.3301	0.8497	0.9988
w/o Trans	0.7737	0.9221	1.2107	0.7087	0.9038
with all	**0.5584**	**00.5971**	**0.6244**	**0.3515**	**0.5328**

out detecting the salient parts, our method works like linear scaling. Therefore, instead of removing the whole saliency detection block that combines CoSD, object detection, and disparity information for the ablation study, we only remove the CoSD module to investigate how it affects the retargeting process. We further ablate with removing the SVT from our model and seeing its impact. Figure 9 shows the results of the ablation study. With the CoSD module removed, the results show that the salient parts are not well detected, affecting the final retargeting results. The deformation of the main object is apparent in both examples. Without the CoSD module, the other parts of the frame are not affected much, but the main objects are deformed. The situation differs when SVT is removed. The training process is affected without the Transformer, and all parts of the frames are affected. More detailed ablation studies are reported in the supplementary materials. In Table 4, we study the influence of CoSD and SVT modules for stereo video retargeting based on the VGG19 feature comparison. As expected, the results show that without the CoSD module (*w/o CoSD*) or SVT module (*w/o Trans*), the results degrade significantly, with the feature difference score increased to almost double in some cases. In addition, in the supplementary materials, we show that our method's performance slightly degrades without uthe use of the disparity information. From

Figure 9. Ablation study. Performance comparison of our model without using CoSD (*w/o CoSD*), without using the SVT (*w/o Trans*), and with all modules (*ours*). Videos are selected from the KITTI stereo 2015 [21] dataset.

Figure 10. Example of failure cases due to: (top) videos with many salient objects and (bottom) wrong detection of salient objects.

Table 4 and Figure 9, we can conclude that removing the CoSD model is more detrimental than removing the SVT model. The performance of the other blocks will be affected without accurately detecting the salient regions. In the future, we will explore assigning more weights to the attention generation in the Transformer block, when fusing SVT and CoSD so that our approach would depend less on saliency detection.

5. Conclusions

In this paper, we proposed a new unsupervised stereo video retargeting method. Our model detects the salient objects, and shifts and warps all the objects in a manner that gives more attention to the salient parts of the stereo frames. We use 1D convolution for shifting the salient objects and design a stereo video Transformer (SVT) to assist the retargeting process. In addition, we reconstruct the source frames from retargeted ones using the PAM module, and a convolutional reconstruction block is used to train the model in an unsupervised manner. Extensive quantitative and qualitative experimental results on the KITTI stereo 2012 and 2015 datasets demonstrate the effectiveness of our proposed stereo video retargeting framework in preserving spatial, temporal, and disparity information.

However, our method fails in some extreme retargeting cases ($> 50\%$ reduction in size). For example, when there are significant salient objects or complex scenes, the shape of some salient objects can be deformed. The first row of Figure 10 shows an example of this case. The black car shape is deformed due to the existence of other salient objects. The other case is related to the failure of the saliency detection method. The two trees in this scene are mistakenly detected as salient objects. To overcome these limitations, additional constraints might be required to preserve the shapes for extreme retargeting cases, which warrants further investigation. Another aspect of video retargeting worth investigating is formulating a benchmark metric for evaluating the performance of retargeted stereo videos.

Acknowledgements. This work is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) 2232 Leading Researchers Program, Project No. 118C301.

References

[1] Shai Avidan and Ariel Shamir. Seam carving for content-aware image resizing. In *ACM SIGGRAPH 2007 papers*, pages 10–es. 2007. 6, 7

[2] Bahetiyaer Bare, Ke Li, Bo Yan, Xiaoyu Qi, and Hamid Gharavi. Pixel fusion based stereo image retargeting. In *2015 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2015. 1

[3] Sung In Cho and Suk-Ju Kang. Temporal incoherence-free video retargeting using foreground aware extrapolation. *IEEE Transactions on Image Processing*, 29:4848–4861, 2020. 6

[4] Zhu Chuning. Fast video retargeting based on seam carving with parental labeling. *arXiv preprint arXiv:1903.03180*, 2019. 6, 7

[5] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 3, 4

[6] Deng-Ping Fan, Tengpeng Li, Zheng Lin, Ge-Peng Ji, Dingwen Zhang, Ming-Ming Cheng, Huazhu Fu, and Jianbing Shen. Re-thinking co-salient object detection. *IEEE transactions on pattern analysis and machine intelligence*, 44(8):4339–4354, 2021. 3

[7] Xiaoting Fan, Jianjun Lei, Jie Liang, Yuming Fang, Xiaochun Cao, and Nam Ling. Unsupervised stereoscopic image retargeting via view synthesis and stereo cycle consistency losses. *Neurocomputing*, 447:161–171, 2021. 1, 2

[8] Xiaoting Fan, Jianjun Lei, Jie Liang, Yuming Fang, Nam Ling, and Qingming Huang. Stereoscopic image retargeting based on deep convolutional neural network. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(12):4759–4770, 2021. 2

[9] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019. 4

[10] Andreas Geiger, Philip Lenz, and Raquel 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. 5, 6, 7

[11] Jiangchuan Hu, Kun Zeng, Kanoksak Wattanachote, and Yongyi Gong. Occlusion-guided vertical retargeting for stereoscopic images based on pixel fusion. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 2775–2779. IEEE, 2020. 1

[12] Hassan Imani, Md Baharul Islam, and Lai-Kuan Wong. A new dataset and transformer for stereoscopic video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 706–715, 2022. 4

[13] Md Baharul Islam, Lai-Kuan Wong, Kok-Lim Low, and Chee Onn Wong. Warping-based stereoscopic 3d video retargeting with depth remapping. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1655–1663. IEEE, 2019. 1, 6

[14] Stephan Kopf, Benjamin Guthier, Christopher Hipp, Johannes Kiess, and Wolfgang Effelsberg. Warping-based video retargeting for stereoscopic video. In *2014 IEEE International Conference on Image Processing (ICIP)*, pages 2898–2902. IEEE, 2014. 1, 6

[15] Seung Joon Lee, Siyeong Lee, Sung In Cho, and Suk-Ju Kang. Object detection-based video retargeting with spatial–temporal consistency. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(12):4434–4439, 2020. 6

[16] Jianjun Lei, Min Wu, Changqing Zhang, Feng Wu, Nam Ling, and Chunping Hou. Depth-preserving stereo image retargeting based on pixel fusion. *IEEE transactions on multimedia*, 19(7):1442–1453, 2017. 1

[17] Bing Li, Ling-Yu Duan, Chia-Wen Lin, Tiejun Huang, and Wen Gao. Depth-preserving warping for stereo image retargeting. *IEEE Transactions on Image Processing*, 24(9):2811–2826, 2015. 1

[18] Bing Li, Chia-Wen Lin, Shan Liu, Tiejun Huang, Wen Gao, and C-C Jay Kuo. Perceptual temporal incoherence-guided stereo video retargeting. *IEEE Transactions on Image Processing*, 29:5767–5782, 2020. 1, 5

[19] Bing Li, Chia-Wen Lin, Boxin Shi, Tiejun Huang, Wen Gao, and C-C Jay Kuo. Depth-aware stereo video retargeting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6517–6525, 2018. 1, 6

[20] Yi Liu, Lifeng Sun, and Shiqiang Yang. A retargeting method for stereoscopic 3d video. *Computational Visual Media*, 1(2):119–127, 2015. 1, 6

[21] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. *ISPRS annals of the photogrammetry, remote sensing and spatial information sciences*, 2:427, 2015. 3, 5, 6, 7, 8

[22] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016. 3

[23] Feng Shao, Wenchong Lin, Weisi Lin, Qiuping Jiang, and Gangyi Jiang. Qoe-guided warping for stereoscopic image retargeting. *IEEE Transactions on Image Processing*, 26(10):4790–4805, 2017. 1

[24] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. Summarizing visual data using bidirectional similarity. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 6

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5, 6, 7, 8

[26] Yukun Su, Jingliang Deng, Ruizhou Sun, Guosheng Lin, Hanjing Su, and Qingyao Wu. A unified transformer framework for group-based segmentation: Co-segmentation, co-saliency detection and video salient object detection. *IEEE Transactions on Multimedia*, 2023. 3

[27] Weimin Tan, Bo Yan, Chuming Lin, and Xuejing Niu. Cycle-ir: Deep cyclic image retargeting. *IEEE Transactions on Multimedia*, 22(7):1730–1743, 2019. 6

[28] Ultralytics. Ultralytics/yolov5: Yolov5 in pytorch; onnx; coreml; tflite. 3

[29] Longguang Wang, Yulan Guo, Yingqian Wang, Zhengfa Liang, Zaiping Lin, Jungang Yang, and Wei An. Parallax attention for unsupervised stereo correspondence learning. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 2, 5

[30] Longguang Wang, Yingqian Wang, Zhengfa Liang, Zaiping Lin, Jungang Yang, Wei An, and Yulan Guo. Learning parallax attention for stereo image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12250–12259, 2019. 2, 4

[31] Xuejin Wang, Pengfei Li, and Feng Shao. Depth trajectory-aware stereoscopic video retargeting. *IEEE Access*, 9:30335–30346, 2021. 1

[32] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International conference on machine learning*, pages 7354–7363. PMLR, 2019. 4