This is a preliminary version of an article published by
Stephan Kopf, Benjamin Guthier, Christopher Hipp, Johannes Kiess, and Wolfgang Effelsberg.
Warping-based Video Retargeting for Stereoscopic Video.
Proc. of IEEE International Conference on Image Processing (ICIP), pp. 2898-2902, 2014.

WARPING-BASED VIDEO RETARGETING FOR STEREOSCOPIC VIDEO

Stephan Kopf, Benjamin Guthier, Christopher Hipp, Johannes Kiess, Wolfgang Effelsberg

Department of Computer Science IV, University of Mannheim, Germany
{kopf,guthier,hipp,kiess,effelsberg}@informatik.uni-mannheim.de

ABSTRACT

In this paper, a novel warping-based retargeting approach for stereoscopic video is proposed. It considers the three conflicting goals: preserving salient image content, avoiding flickering and maintaining consistency between the two views. The first step of the approach focuses on content-aware image resizing and considers image saliency, motion saliency, and depth saliency. In the second step, temporal coherence is preserved by tracking and optimizing deformed pathlines. The proposed algorithm maps the mesh from the left to the right view to guarantee consistency between the deformation of objects between the views. Users rated the quality of the adapted videos as good. In particular, distortions in the perceived depth are not noticeable, and the temporal stability is significantly higher compared to seam carving approaches.

1. INTRODUCTION

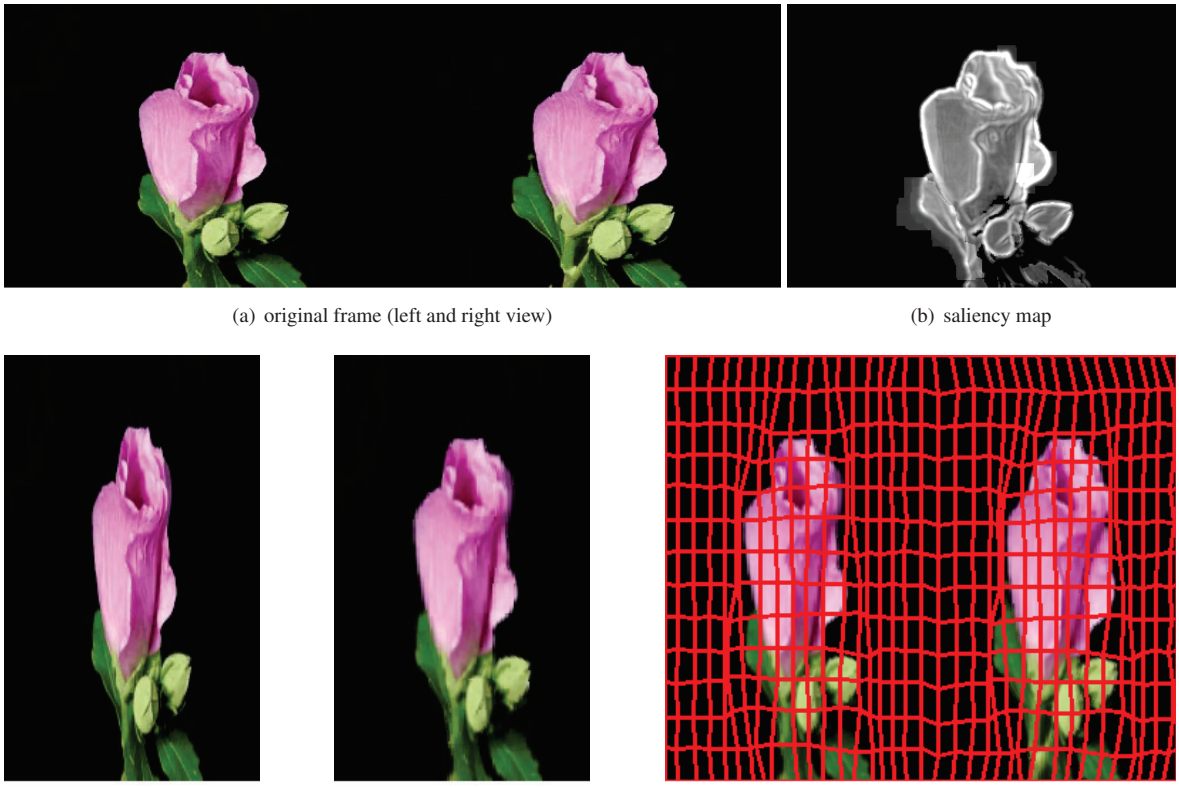
In recent years, multiview video has been getting increasingly popular in the entertainment sector in the form of stereoscopic video. With stereoscopic cinemas, televisions, and the Nintendo 3DS™, a lot of commercial products that can present this kind of visual content have become available. However, stereoscopic movies are produced in a fixed aspect ratio. In order to display these videos on devices with a different screen resolution, the videos need to be retargeted in a way that preserves the shape and motion of visually important objects and regions.

Two different video retargeting approaches have been developed that produce good results. They are *seam carving* and *warping*. However, out of the two approaches, only seam carving has been applied to the retargeting of stereoscopic video [1]. This paper proposes a method for retargeting of stereoscopic video based on warping.

Seam carving [2] removes horizontal or vertical paths of connected pixels from within the image. These paths are called seams. The goal is to remove seams that will not be noticed by the viewer. Several optimizations have been proposed, e.g., seam carving for videos [3, 4, 5], using an improved energy map to preserve important image content [6, 3], novel saliency detection algorithms for stereoscopic video [7], or an efficient GPU implementation that allows real-time video retargeting with seam carving [8]. Utsugi et al. [9] use seam carving for stereoscopic image retargeting. The left view is used as a reference image where most of the resizing is done. Information from the right view is integrated and the seams are then moved to the right view via disparity map. Seam carving is also used by Basha et al. [10]. A seam is searched in both views simultaneously with regard to geometric constraints to prevent distortions in appearance and depth of an image. Guthier et al. [1] present the first system that uses seam carving for stereoscopic video. The work builds upon concepts from image-based stereo seam carving as well as video-based monoscopic seam carving. The energy function used is composed of an appearance term to avoid artifacts in the frame, a disparity term that incorporates 3D information, and a temporal term that reduces flicker in the resulting video. An overview of video retargeting techniques is presented by Kopf et al. [11].

Warping places a rectangular grid mesh over the image and deforms it in a way such that important regions in the image are resized homogeneously while non important regions are allowed to be stretched or squeezed. The goal is to find a mapping that warps the mesh from the source resolution to the target resolution. An optimization problem is formulated with the corner points of the quads being the unknown variables. Fig. 1 (e) shows an example of a warped mesh.

Wang et al. [12] proposed an *image warping* algorithm that considers image saliency such that non salient quads are distorted more. In addition, a grid line bending energy term is introduced because warping of quads that is based solely on saliency can lead to excessive shearing of the edges. Yoo et al. have extended warping to stereoscopic images [13]. They place a rectangular grid mesh over the left view of the image. To find the corresponding mesh in the right view, they perform vertex matching on a multi-



(a) original frame (left and right view) (b) saliency map

(c) linear (d) warped (e) warped mesh

Fig. 1. Reducing the width of the *Flower* video by 50%. (a) shows an original left and right view of a frame and (b) shows its saliency. This video was retargeted by linear scaling (c) and warping (d). (e) illustrates the target mesh after warping.

scale level. To preserve image depth during retargeting, every node that is shifted into one direction on the x-axis in the left view is shifted by the same amount and direction in the right view.

When warping is used for *video retargeting* [14], time-dependent constraints need to be taken into account in order to avoid introducing temporal artifacts. Wang et al. [15] use an optimization over the entire video cube to produce temporal coherent and content-aware resizing. A new energy term based on camera and object motion is added to the frame resizing energy to preserve temporal coherence.

The main contribution of this work is an algorithm that automatically warps a stereoscopic video to fit different display sizes. None of the approaches mentioned above consider stereoscopic videos, and only the technique presented by Yoo et al. considers stereoscopic images [13]. Since their technique was developed for images, Yoo et al. only use disparity and image intensity, but not motion as an indicator for visual saliency. Our saliency detector is composed of image saliency, motion, and depth information. We also present a pathline tracking and optimization algorithm for stereoscopic videos which allows us to balance between the conflicting goals of preserving salient image content and avoiding flickering artifacts.

The outline of the paper is as follows. The next section describes the proposed algorithm for the warping of stereoscopic videos. In Section 3, the quality of the proposed algorithm is evaluated. Section 4 concludes the paper.

2. RETARGETING OF STEREOSCOPIC VIDEO

This section describes our algorithm for warping-based retargeting of stereoscopic video. It focuses on three partially contradicting aspects of the retargeting process. They are:

- Resizing frames while preserving important content

- Preserving temporal stability to avoid motion artifacts
- Maintaining consistency between left and right view

The basis for warping is one rectangular grid mesh $M = (V, E, F)$ for each view of each frame. V is a set of vertices $v_i \in R^2$ which are the corner points of the rectangular grid. E is a set of directed edges (i, j) that connect each vertex to its four neighbors to form the grid. F contains the faces (or "quads") of the grid. A face $f \in F$ is defined by the indices of its four corner vertices. Only the sets of vertices differ between the views. By our notation, a vertex v_i^L in the left view corresponds to the vertex v_i^R in the right view with the same index i. To avoid clutter, we omit the superscripts L and R unless required for better understanding.

The goal of warping a frame is now to find new vertex positions $v_i' \in V'$ in a target frame with a different resolution such that the important areas are preserved as well as possible. For this purpose, we formulate and solve an optimization problem consisting of energy terms that take the three aspects into account. This is the main focus of this paper. The 2D locations of the target vertices are the unknown variables that are being optimized. Once new vertex positions are known for every frame of the video, the pixels of each quad of the original video are transformed into the deformed quads to warp the video.

To achieve the goals mentioned above, the algorithm works in three sequential steps which are described in the following three sections. Warping is first performed on each frame individually. This is done in a way that preserves the important content in the best way possible while completely ignoring temporal consistency. As a second step, the motion of the original video is compared to the motion after warping. The difference is temporal inconsistency. We thus find a trade-off between the content-preserving warp and consistency with the motion in the original video. Based on this trade-off, the video is warped again in the third step.

In order to judge the relevance of each pixel in the stereoscopic video, we compute a saliency map using the approach presented by Dittrich et al. [7]. The left and right mesh is placed over the saliency map respectively in order to determine the saliency of the quads.

2.1. Content-Aware Warping

In the first step, the video is warped in a content-aware manner frame by frame while temporal consistency is ignored. The vertices $v_i' \in V'$ of the warped mesh are determined by minimizing an energy function with the v_i' being the unknown variables. The target energy function consists of several energy terms that are explained in the following.

Quads with a high saliency contain important image content and should thus be scaled uniformly. We define an energy term that measures the difference between the deformed quad and a uniformly scaled version of it. In an optimal case, there exists an unknown scaling factor s_f for each quad face $f \in F$, such that for each vertex v_i in the quad, $v_i' = s_f \cdot v_i + t$, where t is a constant translation vector. The quad deformation energy of the left mesh is then defined as

$$d_q(f) = \sum_{(i,j)\in E(f)} || \left(v_i' - v_j'\right) - s_f \left(v_i - v_j\right) ||^2 \quad (1)$$

where $E(f)$ denotes the set of edges surrounding quad f. This energy term becomes smaller, the better the four vectors of a quad agree on a single scaling factor s_f. To add saliency information into the energy term, the energy of a single quad f is weighted by its quad saliency w_f as $D_q = \sum_{f\in F} w_f \cdot d_q(f)$. This definition as well as (1) were taken from [12]. In order to apply the energy term D_q to stereoscopic video, it must be formulated for the left and the right view. The result is two values D_q^L and D_q^R that represent the quad energy of the left and the right view, respectively. The total quad energy D_q is the sum of D_q^L and D_q^R [13].

To preserve the consistency between the left and right view, the energy term of Yoo et al. [13] is used. It becomes smaller, when vertices v_i^L from the left mesh get deformed in a way similar to the corresponding vertices v_i^R. This can be expressed as:

$$D_d = \sum_i || \left(v_i^L - v_i'^L\right) - \left(v_i^R - v_i'^R\right) ||^2 \quad (2)$$

To obtain the two target meshes of a frame, the sum of the quad and the disparity energy is minimized. The target energy function is thus $D = D_q + D_d$. Our minimization approach is very similar to the one used in [12].

2.2. Motion Analysis

Now that the input video has been warped by a frame-by-frame approach, we compare the motion in the warped video to the motion of the original one. Instead of computing the full optical flow of the video, it is sufficient to only track the motion of individual points. We refer to the motion of these points through the video as *pathlines*. In a video that was scaled by the same factor for every frame, all pathlines undergo the same transformation. However, in a video that was warped, this may not be the

case. Inconsistency between the pathlines in the warped video causes temporal artifacts like flicker. Therefore, this step of the algorithm calculates a set of optimized pathlines. These optimized pathlines are a combination of the deformed pathlines and pathlines of the input video. This approach is taken from [16].

Let $\mathcal{P}$ denote the set of all pathlines in the *original* video. Each pathline $P_i \in \mathcal{P}$ is a sequence of pixels $P_i = (p_i^1, p_i^2, \ldots, p_i^T\}$, with $p_i^t = (x_i^t, y_i^t)$ being the location of the pathline pixel in frame t. T is the number of frames in the video. The pathline index i indicates that the vertex v_i in the first frame with the same index was used as a seed. Two pathlines P_i and P_j are adjacent to one another if their seeds v_i and v_j are adjacent in the first frame, i.e., $(i,j) \in E$.

Ideally, to achieve temporal consistency, the video would be warped in such a way that the original P_i would be scaled uniformly. This means that all offsets $p_i^t - p_j^t$ between adjacent pathlines should undergo a scaling, expressed as a multiplication by a scaling matrix $S_{ij} \in M(2\times 2, \mathbb{R})$. Using this criterion, the new pathlines $\hat{p}_i \in \hat{P}_i$ after warping can be calculated by minimizing the energy term

$$E_p = \sum_{(i,j)\in E} \sum_{t=1}^{T} ||(\hat{p}_i^t - \hat{p}_j^t) - S_{ij}(p_i^t - p_j^t)||^2 \qquad (3)$$

for the unknown variables S_{ij} and $\hat{p}_i^t$.

Note that there are a large number of variables $\hat{p}_i^t$ (one tuple per frame per vertex). In order to reduce the number of variables to optimize for, the deformed pathline $\hat{P}_i$ is modeled by scaling and translating the original one:

$$\hat{P}_i = S_i P_i + t_i \qquad (4)$$

Here, S_i is a 2×2 matrix and t_i is a translation vector. If this is applied to every pathline, the reduced model only contains one unknown scaling matrix and one unknown translation vector *per pathline*. Equation 3 can then be rewritten as:

$$E_p = \sum_{(i,j)\in E} \sum_{t=1}^{T} || \left((S_i p_i^t + t_i) - (S_j p_j^t + t_j)\right) - S_{ij}(p_i^t - p_j^t)||^2 \qquad (5)$$

This energy term encourages the warp to scale the pathlines uniformly leading to perfect temporal consistency (ignoring the importance of content). By applying the warp that was determined in the previous content-aware step to the pathlines P_i of the original video, they are warped into new pathlines P_i' that may be temporally inconsistent. The ideal pathlines $\hat{P}_i$ should be a compromise between temporal consistency and content-awareness. The latter can be formulated by an energy term that aims to reduce the distance between the ideal pathlines $\hat{P}_i$ and the content-aware ones P_i'. By using Equation 4, this can be expressed as

$$E_c = \sum_i \sum_{t=1}^{T} ||(S_i p_i^t + t_i) - p_i'^t||^2 \qquad (6)$$

Equation 5 and 6 are then combined [16] to yield the final pathline consistency energy $E = E_p + \lambda E_c$, where λ is a balance factor to balance between temporal consistency of the pathlines and content-awareness. E is minimized to solve for S_i, S_j, S_{ij}, t_i, and t_j. The obtained variables can be used to calculate the optimized pathline according to Equation 4.

2.3. Temporally Consistent Warping

The video is now warped again using the optimized pathlines $\hat{P}_i$ as guide points. When warping frame t, we would like the vertices v_i to get deformed according to the positions $\hat{p}_i^t$ of the optimal pathlines at time t. We thus add an energy term to the objective function [16] of frame t: $D_p = \sum_i ||\tilde{p}_i^t - \hat{p}_i^t||^2$ where $\hat{p}_i^t$ is the optimized pathline point in frame t and $\tilde{p}_i^t$ is the final pathline point position which is unknown. The unknown pathline point can be expressed in terms of the quad it lies in by averaging the four surrounding vertices: $\tilde{p}_i^t = \sum_{k\in\Phi(p_i^t)} \frac{1}{4} \cdot v_k'^t$ where $\Phi(\tilde{p}_i^t)$ are the indices of the vertices that surround $\tilde{p}_i^t$. Now the energy term can be written as

$$D_p = \sum_i || \sum_{k\in\Phi(p_i^t)} \frac{1}{4} \cdot v_k'^t - \hat{p}_i^t||^2 \qquad (7)$$

v' are the unknown final vertex positions in this energy term. This term is formulated once for the vertices in the left view and once for the right view. The resulting energy term is the sum of the terms for the two views. Now D_P is added to the target energy function: $D = D_q + D_d + \lambda D_p$. Here, λ is the same weighting factor as above. D is used as the final objective function to determine the final vertex positions for warping.

3. EVALUATION

To evaluate the visual quality of the implemented warping algorithm, we used five stereoscopic video sequences[1] (see Table 1). The width of each video is either reduced or increased by a factor of two. They were presented to

[1] car, flower: www.stereomaker.net/sample/; person, bouquet: http://sp.cs.tut.fi/mobile3dtv/stereo-video/; moon: www.youtube.com

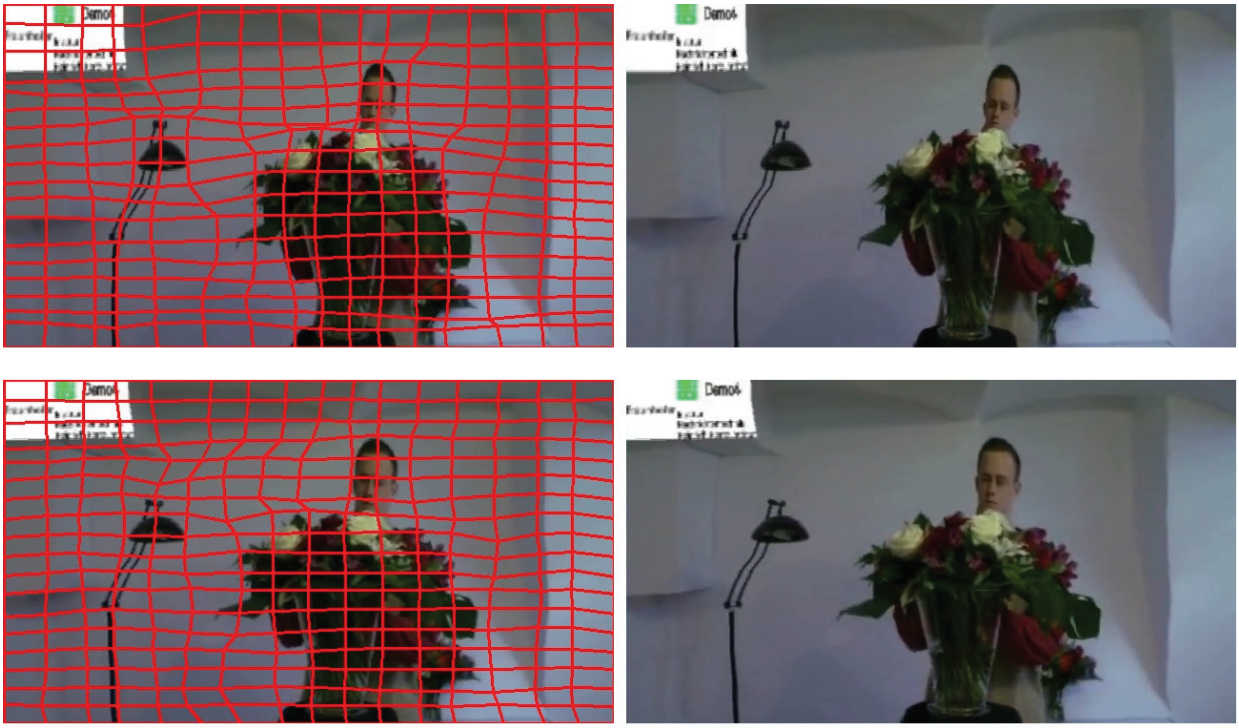


Fig. 2. Warped frame of the *bouquet* sequence without (top) and with pathline optimization (bottom).

Table 1. Parameters of the test videos

video	original size	target size	number of frames
car	600×240	300×240	450
flower	720×262	360×262	260
moon	640×480	1280×480	80
person	640×360	320×360	225
bouquet	640×360	1280×360	375

eight test viewers wearing 3D shutter glasses. Two resizing approaches were compared: framewise warping (see Section 2.1) and temporally consistent warping with optimized pathlines (see Sections 2.2 and 2.3). The retargeted videos were shown in random order next to the original one for reference. The size of the quads was 20×20 pixels and λ was set to $\frac{1}{4}$.

Both methods preserve important image content equally well and no user complained about important regions being removed. Fig. 1 shows warping results of the *flower* video. The flower blossom is clearly salient (Fig. 1b) and the quads that are placed over the flower blossom maintain their aspect ratios relatively well (Fig. 1e).

Most users noticed temporal inconsistencies in at least some of the videos. This is a general problem with video retargeting [1]. Noise leads to differing saliency maps in two consecutive frames which causes changes of the retargeting parameters and results in flicker. Such undesirable artifacts immediately attract the attention of the viewer because the human eye is highly sensitive to motion. Major temporal defects were noticed in both versions of the *person* sequence. The background in the video is highly structured but not marked as salient. A lot of waving in the background was observed by almost all users. Except for this sequence, which produces a large number of visual defects with both methods, temporally consistent warping with pathline optimization leads to comparable or significantly better results. This is because the optimized pathlines stabilize the motion of the quads. This becomes evident from the users' comments about the *bouquet* video (see Fig. 2), where a man enters the scene to work on a bouquet of flowers. Due to the colorful flowers, the contrast saliency of the man's face is low and the face thus gets deformed heavily by the framewise warping algorithm. However, when considering the motion of the face, it becomes temporally salient. The tracked pathlines then stabilize the quads, which was perceived as a decrease of artifacts.

To measure the run-time[2], the width of the *flower* video has been doubled. With a computation time of 6.56 seconds per frame, 88.5% of the overall time is spent for solving the optimization problems. We use the NLopt library[3] from MIT for this task. This high computational load is caused by the large number of vertices and the 20 alternating optimization steps. The computation time of pathline tracking (5.6%) and image warping (3.2%) is significantly lower. All the other computations like saliency, disparity, I/O operations, or SURF feature matching for detecting corresponding vertices in the left and right view as well as in following frames require less than 3%.

4. CONCLUSIONS

In this paper, a warping-based system that automatically retargets stereoscopic videos was presented. The video is first warped frame by frame by using a saliency map. By analyzing the motion in the video before and after frame-wise warping, optimized motion pathlines are computed. These optimized pathlines are used as guide points during the final temporally stable warping step.

5. REFERENCES

[1] B. Guthier, J. Kiess, S. Kopf, and W. Effelsberg, "Seam carving for stereoscopic video," in *IEEE IVMSP Workshop*, 2013, pp. 1–4.

[2] S. Avidan and A. Shamir, "Seam carving for content-aware image resizing," *ACM Trans. Graph.*, vol. 26, no. 3, 2007.

[3] M. Rubinstein, A. Shamir, and S. Avidan, "Improved seam carving for video retargeting," *ACM Trans. Graph.*, vol. 27, no. 3, pp. 16:1–16:9, 2008.

[4] J. Kiess, B. Guthier, S. Kopf, and W. Effelsberg, "SeamCrop for image retargeting," in *IS&T/SPIE Electronic Imaging (EI) on Multimedia on Mobile Devices*, 2012, vol. 8304.

[5] S. Kopf, J. Kiess, H. Lemelson, and W. Effelsberg, "FSCAV: Fast seam carving for size adaptation of videos," in *ACM international conference on Multimedia (MM)*, 2009, pp. 321–330.

[6] J. Kiess, S. Kopf, B. Guthier, and W. Effelsberg, "Seam carving with improved edge preservation," in *IS&T/SPIE Electronic Imaging (EI) on Multimedia on Mobile Devices*, January 2010, vol. 7542, pp. 1–11.

[2]Intel Core i7-3770, 4 GB RAM, Windows 7 (32 bit)

[3]http://ab-initio.mit.edu/wiki/index.php/NLopt

[7] T. Dittrich, S. Kopf, P. Schaber, B. Guthier, and W. Effelsberg, "Saliency detection for stereoscopic video," in *ACM Multimedia Systems (MMSYS)*, 2013, pp. 12–23.

[8] J. Kiess, D. Gritzner, B. Guthier, S. Kopf, and W. Effelsberg, "GPU video retargeting with parallelized seamcrop," in *ACM Multimedia Systems Conference*, 2014, pp. 139–147.

[9] K. Utsugi, T. Shibahara, T. Koike, K. Takahashi, and T. Naemura, "Seam carving for stereo images," in *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video*, 2010, pp. 1–4.

[10] T. Basha, Y. Moses, and S. Avidan, "Geometrically consistent stereo seam carving," in *Computer Vision (ICCV)*, 2011, pp. 1816–1823.

[11] S. Kopf, T. Haenselmann, J. Kiess, B. Guthier, and W. Effelsberg, "Algorithms for video retargeting," *Multimedia Tools and Applications (MTAP)*, vol. 51, pp. 819–861, 2011.

[12] Y.-S. Wang, C.-L. Tai, O. Sorkine, and T.-Y. Lee, "Optimized scale-and-stretch for image resizing," in *ACM SIGGRAPH Asia*, 2008, pp. 118:1–118:8.

[13] J. W. Yoo, S. Yea, and I. K. Park, "Content-driven retargeting of stereoscopic images," *Signal Processing Letters, IEEE*, vol. 20, no. 5, pp. 519–522, 2013.

[14] P. Krähenbühl, M. Lang, A. Hornung, and M. Gross, "A system for retargeting of streaming video," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 126:1–126:10, 2009.

[15] Y.-S. Wang, H. Fu, O. Sorkine, T.-Y. Lee, and H.-P. Seidel, "Motion-aware temporal coherence for video resizing," in *ACM SIGGRAPH Asia*, 2009, pp. 127:1–127:10.

[16] Y.-S. Wang, J.-H. Hsiao, O. Sorkine, and T.-Y. Lee, "Scalable and coherent video resizing with per-frame optimization," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 88:1–88:8, 2011.