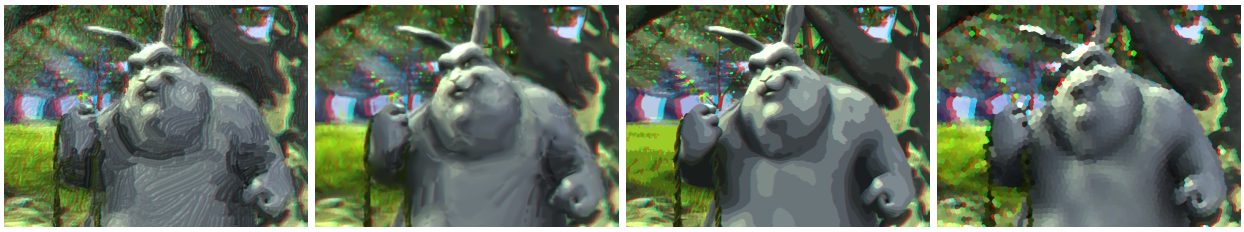
Vision, Modeling, and Visualization (2015)
D. Bommes, T. Ritschel, and T. Schultz (Eds.)

Rotoscoping on Stereoscopic Images and Videos

Dennis R. Bukenberger, Katharina Schwarz, Fabian Groh† and Hendrik P. A. Lensch

Department of Computer Graphics, Eberhard Karls University, Tübingen, Germany



(a) *SBR - Paint Texture* (b) *SBR - Diffusion Curves* (c) *k-Means Posterization* (d) *Window Glass Mosaic*

Figure 1: ‡ *Stereo consistent abstraction with different styles. Based on the same stereoscopic image, our framework renders various different styles such that the effect is consistent for the left and right image. Source image: [GR08]*
‡ *Figures marked with this icon are best examined on-screen with red(left)-cyan(right) anaglyph glasses.*

Abstract

Creating an animation based on video footage (rotoscoping) often requires significant manual work. For monoscopic videos diverse publications already feature (semi-)automatic techniques to apply non-photorealistic image abstraction (NPR) to videos. This paper addresses abstraction of 3D stereo content minimizing stereoscopic discomfort in images and videos. We introduce a completely autonomous framework that enhances stereo and temporal consistency. Stereoscopic coherence with consistent textures for both eyes is produced by warping the left and right images into a central disparity domain followed by mapping them back to the left and right view. Smooth movements with reduced flickering are achieved by considering optical flow in the propagation of abstract features between frames. The results show significant improvements of stereo consistency without discomforting artifacts in the depth perception. We extend existing stroke based rendering (SBR) for higher accuracy at strong image gradients. Furthermore, we demonstrate that our stereo framework is easily applicable to other point-based abstraction styles. Finally, we evaluate the stereo consistency of our results in a small user study and show that the comfort of the visual appearance is maintained.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Viewing algorithms I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—Animation I.3.m [Computer Graphics]: Miscellaneous—Visual arts

1. Introduction

Rotoscoping is a traditional animation technique which allows the animator to use real video footage as basis for his artistic abstraction process. The goal is to keep proportions and movements of the captured scenery in the animation physically correct. Previous work already featured methods for artistic stylization of images and even animations; sometimes still based on human interaction to determine important image regions or to draw whole key frames manually. *NPR*-methods that work for single images are not trivially applicable to stereoscopic footage nor to animations because abstracted results must stay consistent in stereo and over time. Both consistency constraints form the key issues to ensure visual comfort. In order to address those challenging constraints, we now introduce a completely autonomous framework with the following main contributions:

† This work has been partially funded by the DFG Emmy Noether fellowship (Le 1341/1-1).

DOI: 10.2312/vmv.20151265

Central Warp: Abstracting stereoscopic images individually leads to left-right inconsistencies which can result in very uncomfortable depth perception artifacts. As solution for stereo consistent image abstraction, we propose the *Central warp*, which empowers our process to extract depth information before the stylization and to reintroduce disparities just before rendering.

Optical Flow Warp: In order to render an animation with smooth movements, two succeeding frames must not differ too much. Therefore, our framework maintains temporal consistency with an *optical flow warp*. Abstracted image information from one frame is warped to its successor and automatically corrected where necessary.

Dual Stroke: As improvement to the abstraction style we chose for demonstration (*SBR*), we introduce the *dual stroke*. This technique allows a more accurate representation of strong image gradients. Dual strokes increase the overall achievable sharpness impression of the rendering and, thus, lower the need for incremental *SBR* refinements.

Our proposed stereoscopic rotoscoping process causes only marginal interference with established image abstraction workflows. While focusing on the expressionistic *SBR*-style (Figure 1 (a)) to demonstrate stereo and temporal consistent rotoscoping, we present other compatible abstraction methods (Figure 1 (b-d)) in Section 4. However, any other point based abstraction method can be applied easily.

1.1. Related Work

The concept of rotoscoping was patented by Max Fleischer [Fle17] and brought to a wider audience in Snow White and the Seven Dwarfs [HMD37]. First manual rotoscopings were realized by backprojecting single film frames on frosted glass while tracing it with pen and paper. As artistic style of abstraction, our work employs popular *SBR* approaches proposed by Litwinowicz [Lit97], Hertzmann [Her98], and Hays and Essa [HE04]. We adapted the paint texture simulation proposed by Hertzmann [Her02] to become motion and stereo compatible. Besides real footage, we also allow for synthetic input data to explore stereo quality.

In contrast to interactive systems, e.g., Kim et al. [KWL13] provide a manual framework based on still images, we aim at a completely autonomous rotoscoping framework. Hertzmann and Perlin [HP00] studied the difficulties of naive application of painterly abstraction to animation. Their initial approach uses the stroke mesh of a previous frame as basis and to overdraw regions which became defective due to motion. They also proposed the use of optical flow to warp strokes to consecutive frames. Another automated approach for time consistent stroke based animation rendering was proposed by Collomosse et al. [CRH05]. Instead of user based labeling, some interactive systems employ color based image segmentation with the option of interactive refinement for the motion prediction. However, our approach does not employ segmentation but extends the usage of motion vectors from optical flow estimation. All our strokes are warped with their according motion vectors to match the next frame; the majority passes automated sanity checks. Therefore, only a few areas require corrections with new strokes which are properly incorporated to the rendering hierarchy during creation. Furthermore, our strokes smoothly blend in and out using variable opacity [HE04].

Besides the *SBR*, we utilize additional styles of abstraction to demonstrate the application independence of our framework. The cartoonish posterization style was inspired by [Sab97, WOG06, LMY*13]. Whereas Sabiston's software [Sab97] interpolates between manually drawn keyframes, we use k-Means vector quantization to realize the color clustering. Instead of applying pixel-based filters to the image as Winnemöller et al. [WOG06], our approach is point and contour based and extracts vectorizable image information. Similar to Haeberli [Hae90] and Hausner [Hau01], our third *NPR* style creates mosaics using Voronoi-cells.

Richardt et al. [RŚDD11] stated that deviations between the left and right image of a stereoscopic image pair cause discomfort for the viewer. Applying *NPR* approaches naively to both stereo images individually introduces stereo inconsistency and disturbs the subjective depth impression of the result. Stavrakis and Gelautz [SG05] handled this issue for stereo consistent painterly renderings with a layer based vice versa stroke warping approach, while occluded areas are corrected. Similarly, Northam et al. [NAK12] introduced an approach to decompose the stereoscopic image into discretized disparity layers as basis for the painterly reproduction. With the removal of the occluding object and information from both image views, occluded areas are restored within their distinct depth layer. However, those layer based approaches are not suitable for temporal consistency. Therefore, we focus on point based methods directly within our monoscopic space. Inspired by the work of Bleyer et al. [BGRR09] to generate artificial left and right views, our approach reestablishes disparity after the abstraction process, which works well for many abstraction styles. Further, the publication of Lang et al. [LHW*10] targets the remapping of stereoscopic content for different display situations. In contrast to their approach with sparse stereo correspondences, we employ frame sized disparity maps for our Central warp.

1.2. Overview

First, depth information is extracted from the stereoscopic input serving as the basis for the Central warp (Sect. 2) to create a monoscopic image space. For this space, desired *NPR*-methods are generalized to point based vectorizations. The inverse disparity warp is applied on the abstracted point sets to render left and right views. We achieve temporal consistency (Sect. 3) by using optical flow motion vectors to propagate point sets to succeeding frames. We provide examples for *NPR*-methods and extensions (Sect. 4).

2. Stereo Consistency

In this section, we analyze the concept of stereoscopic imaging and state the main challenge for *NPR*-methods, namely to abstract and to render stereo consistent images.

2.1. Depth and Disparity

The human visual system judges distance to an object based on a variety of depth cues. However, the main depth cue exploited in stereoscopic imaging is retinal disparity, which is the relative offset of a feature projected on the viewers retinas. This retinal disparity is induced by image disparity, showing a feature with different horizontal offset in both images, presented individually to the left and right eye of the observer. In the simplest case, a stereo image can be captured by two parallel aligned cameras with a small horizontal offset (the baseline). From those two images the image disparity can be computed by measuring the distance between the positions of a feature in the left and the right image. We use the OpenCV implementation of Hirschmüller's *SGBM*-algorithm [Hir08] to compute image disparity.

Definitions: For the following sections, let a stereoscopic image $I = (\mathcal{L}, \mathcal{R})$ be defined as a left and right image, shot or, rendered from two distinct view points. These two images correspond to the left and right eye of the observer. For a stereoscopic image two disparity maps are defined:

$$\begin{aligned} D_{L\to R} &= \text{disparityMap}(\mathcal{L}, \mathcal{R}) \\ D_{R\to L} &= \text{disparityMap}(\mathcal{R}, \mathcal{L}) \end{aligned} \tag{1}$$

One map holds the offset values from left to right ($D_{L\to R}$) and one holds the values from right to left ($D_{R\to L}$).

Challenges: In terms of *NPR* stereo consistency, a stereoscopic image addresses similar challenges as two temporal succeeding frames: Abstracted representations of the same object must not differ fundamentally over time nor in stereo. Independently extracted information for $\mathcal{L}$ and $\mathcal{R}$ typically results in two dissimilar images. Furthermore, the computational costs to handle both images separately would effectively double. Therefore, we formulate an approach to separate stereo information from the image before the abstraction and combine it again for rendering by introducing the concept of a monoscopic warp.

2.2. Stereo Merging with the Central Warp

Along with the computation of $D_{L\to R}$ and $D_{R\to L}$, the averaged monoscopic image $\mathcal{M}$ is initialized by:

$$\mathcal{M} = \frac{\text{warp}(\mathcal{L}, \frac{D_{L\to R}}{2}) + \text{warp}(\mathcal{R}, \frac{D_{R\to L}}{2})}{2} \tag{2}$$

The disparity values in $D_{L\to R}$ and $D_{R\to L}$ are scaled by $\frac{1}{2}$ and used for warping. With these scaled disparity values, the images $\mathcal{L}$ and $\mathcal{R}$ are central-warped into monoscopic space $\mathcal{M}$, each side providing 50% of color information per pixel. The image abstraction process is applied on $\mathcal{M}$.

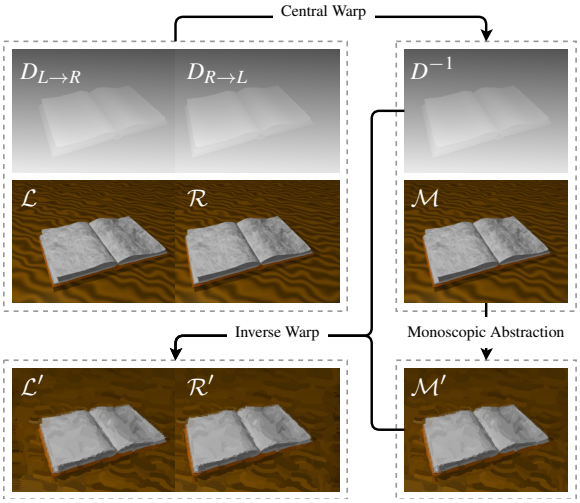


Figure 2: *Equations 2, 4 visualized on a synthetic stereo image with ground truth disparity. Source images: [ROD*10]*

Scaling the warp values is necessary since the disparity of a pixel denotes the total offset to its equivalent in the other image. A warp with full disparity would result in artificial left and right images:

$$\mathcal{L} \approx \text{warp}(\mathcal{R}, D_{R\to L}) \qquad \mathcal{R} \approx \text{warp}(\mathcal{L}, D_{L\to R}) \tag{3}$$

The listed $warp(im, D)$ function implements a forward mapping of the first argument, the image *im*, using Lanczos interpolation. Offset values for each pixel in the image are provided as second argument by the disparity map D. During the warp occurring occlusions are resolved with the disparity values itself. Higher absolute disparity of a pixel denotes higher priority and, thus, keeps it staying visible. Occurring undefined regions are replenished with the lowest absolute disparity of valid adjacent areas.

To be able to regain a stereoscopic result after the abstraction, the inverse operation is applied to results of $\mathcal{M}$. The inverse D^{-1} of the monoscopic remap function is created by merging the disparity maps with the value-scaled versions of themselves:

$$D^{-1} = \frac{\text{warp}(D_{L\to R}, \frac{D_{L\to R}}{2}) + \text{warp}(-D_{R\to L}, \frac{D_{R\to L}}{2})}{2} \tag{4}$$

Furthermore, the sign of one disparity map has to be inverted, so that they can be averaged; otherwise they would cancel out each other when summed. A remap with the inverse D^{-1} reestablishes stereo consistent versions of every element from $\mathcal{M}$. This can be applied to any abstracted point-based image information or $\mathcal{M}$ itself:

$$\mathcal{L}' = \text{warp}(\mathcal{M}', D^{-1}) \qquad \mathcal{R}' = \text{warp}(\mathcal{M}', -D^{-1}) \tag{5}$$

The described steps from Equations 2 and 4 are illustrated as the *Central Warp* in Figure 2. $\mathcal{L}'$ and $\mathcal{R}'$ are the synthetic stereoscopic results, created from $\mathcal{M}'$ and D^{-1}.

Figure 3: *Two SBR sequences showing every 3rd frame. First row: Temporal inconsistency results in a different pattern in each frame. Second Row: Consistent rendering due to our motion warp with optical flow data. Source images: [Aal12]*

3. Temporal Consistency

The main challenge in creating a temporal smooth sequence consists in rendering the *SBR*-animation without flickering. We employ the approach of Hertzmann and Perlin [HP00] to use optical flow information and incremental corrections for animating our stroke meshes.

3.1. Motion Warp

Processing every frame of a video stream individually is neither efficient nor would it result in a smooth animation. Even small noise-like differences from frame to frame can produce different stroke meshes, caused by a slightly different interpretation of the gradients. An example is given in Figure 3. If stroke meshes of two succeeding frames are too different to each other, the animation will begin to flicker. To prevent this, the stroke mesh is warped with motion vectors based on the estimated optical flow from the source video. Hays and Essa [HE04] implemented motion in their stroke mesh by only moving anchorpoints and adapting the angles of straight strokes. However, we need to shift multiple points with distinct 2D motion vectors to warp one stroke. Thus, the optical flow map is used as a lookup table providing individual offsets for each of the strokes points. As non-pixel-accurate optical flow estimations can lead to defaced and wrinkled strokes, motion vectors are median filtered over a set of three points along the stroke. Besides, the validity of a stroke is checked after each warp. A stroke does not qualify for further use if any of its validity criteria is violated. Criteria thresholds are defined for the mean color difference between the stroke's initial color and the sampled color of its new position, the total change of the stroke's length ($\pm 25\%$), and the change in disparity of stroke points.

3.2. Fadings

In order to replace old strokes that became invalid, new strokes are established. The occurrence of new strokes or the removal of invalid ones can be smoothed with fading. Therefore, the alpha value of a stroke is altered over time.

3.3. Optical Flow

For our bidirectional approach, we need to buffer all required frames before the abstraction process. The OpenCV implementation of Farnebäck's algorithm for two-frame motion estimation [Far03] is applied recursively on the buffered frames. The recursion implies the use of $flow_{n-1\to n}$ for the computation of $flow_{n\to n+1}$. This step is also applied to the frame stack in reversed order. To ensure a more reliable and robust motion vector estimation, the in-order flow and the reverse-order flow are combined accordingly.

4. Rotoscoping Methods

The presentation of stereo consistent results with different abstraction styles indicates the applicability of our framework to a whole class of *NPR*-techniques. Although, we mainly utilized *SBR*-methods and their extensions, we also provide examples for more styles of abstraction. The main feature that the presented *NPR*-techniques have in common is, that they are vectorizable in the sense of being point-based. Image data is extracted and rendered in different ways but the core of abstracted information is handled as multiple sets of points. The following extraction methods take place in mono-image space, obeying our framework principles introduced in Section 2.

4.1. Stroke Based Rendering with Dual Stroke

For our version of *SBR* we extend the work of Hertzmann [Her98] and Hays and Essa [HE04] with a dual stroke technique. The dual stroke is a different geometric rendering-interpretation of a single stroke that simply renders two different colored strokes instead of one to draw a gradient. This technique is illustrated in Figure 4. For each stroke point, two new points are placed on both sides of the stroke with

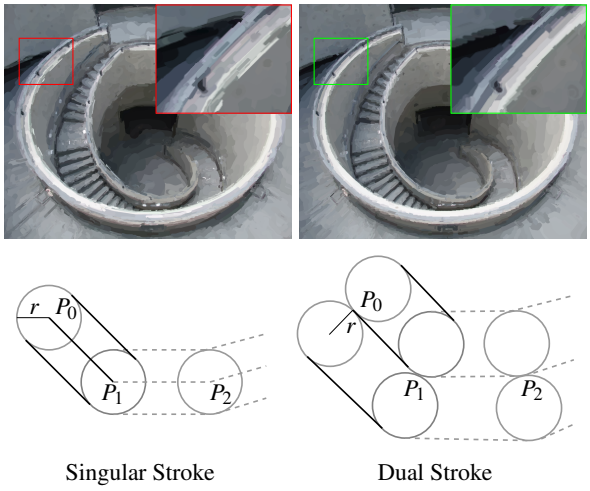


Figure 4: *The first row compares exemplary renderings from both stroke types. The drawings in the second row visualize the different geometric constructions for the same set of points* $(P_0, P_1, P_2, \ldots)$. *Source Image: [Tak09]*

radial distance. These new points are positioned perpendicular to the Catmull-Rom-tanget of their origin. It emphasizes fine details, allows more accurate renderings and, therefore, results in less incremental correction steps. Contrary to intuition, the improved result on the right in Figure 4 is achieved with less computation-costs ($\sim 50\%$) compared to the left image. We use a set of multiple points to define curved brush strokes. Our inverse stereo warp in Section 2.2 and the optical flow warp in Section 3.1 operate on these sets of points. The *fast paint-texture* concept of Hertzmann [Her02] is also used in some of our examples (e.g., Figure 1 (a)).

4.2. More Abstraction Styles

Diffusion Curves: Inspired by Orzan et al. [OBB*13] and Pérez et al. [PGB03], we additionally experimented with diffusion curves as extension to *SBR*. Color information is only given along spline-like curves and smoothly interpolated over the enclosed canvas area. Therefore, dual strokes are drawn with a radius and radial offset of only 1 pixel. The blank regions between the colored pixels are interpolated. A stereo consistent rendering of *SBR* based diffusion curves is shown in Figure 1 (b).

Posterization: In order to generate a posterization effect, we cluster regions of similar RGB color using k-Means vector quantization. Figure 1 (c) illustrates an example of this style. The outer silhouettes of emerging color-clusters are approximated with polygonal contours containing at least three points. The stereo warp in Section 2.2 operates on these sets of points. Figure 6 features this *NPR*-style and compares naive depth-disturbed results to our stereo consistent renderings. Temporal consistency with the optical flow warp would be possible with more accurate optical flow estimations. Alternatively, we reuse the k centers from frame f_n as base for the quantization of frame f_{n+1}.

Mosaic Tiles: This artistic abstraction method creates decorative mosaic images extending the approach proposed by Hausner [Hau01]. In the stereo consistent example (Figure 1 (d)), the regions between the tiles are filled up, leading to a similar expression as in mosaic glass-windows. For this style, the stereo warped set of points (Section 2.2) contains only one point which is the center position of a tile. For temporal consistency, the optical flow warp would disturb the desired relaxed distribution of tiles in the mosaic. Therefore, tiles roughly keep their position, only re-sample their color and reorient according to one of the next frames.

5. Discussion

We discuss our results on several video frames as well as on stereoscopic photographs. More examples are shown in the attached supplemental material.

5.1. Analogies in Abstraction Methods

The common feature of all methods presented in Section 4 are the sets of points. Points are created in the averaged disparity domain, described in Section 2.2. A point's disparity offset is queried from the averaged disparity map and will reposition it temporarily for stereo consistent rendering. Since every point is treated individually, the formulation, handling and cardinality of point sets are not relevant with respect to the stereo warp. However, the desired method of abstraction should allow an intermediate step between extraction and rendering, where the stereo warp can be executed.

5.2. Data and Parameters

Stereoscopic Input Data: Stereoscopic datasets created for education and research are quite suitable for computer vision benchmarking, since they often provide most valid stereo information. We experimented with stereoscopic photographs, cinematographic stereo footage and various publicly available clips we retrieved from YouTube. Our processing handles stereoscopic side-by-side images and videos of any common resolution. Comparing synthetic to real footage, one noticeable difference is the achievable precision of supplementary computations as optical flow and disparity maps. In the attached supplementary video material, one can observe the effect of varying optical flow accuracy, comparing renderings from synthetic and real footage.

Tunable Parameters: Good parameters for the OpenCV implementations like disparity and optical flow computation were determined heuristically as they largely depend on the source material. A critical issue, regarding our temporal consistency approach is the fading time of strokes: If the fading-time is too short, subjects complain about stroke popping artifacts. However, too long in-fading can leave canvas areas

uncovered from strokes whereas too long out-fading smears invalid old strokes. Thus, we chose 2 frames for in- and 4 frames for out-fading in most examples.

5.3. Results

Stereo consistency: For the example shown in Figure 5, we chose standard disparity estimation as measure for the stereo inconsistency errors. $\mathcal{L}_1$ and $\mathcal{R}_1$ were abstracted independently without our consistency approach. By comparing $\mathcal{L}_1$

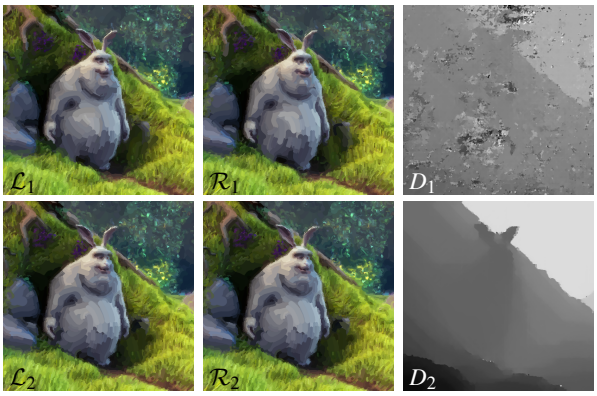


Figure 5: *Left and center column: $\mathcal{L}_i$ and $\mathcal{R}_i$ from a stereoscopic frame. Right column: Derived disparity Map. Top row: SBR without stereo consistency. Bottom row: SBR with our consistency approach. Source images: [GR08]*

and $\mathcal{R}_1$, one can observe the different stroke meshes, especially on the bunny's torso. Furthermore, the defected disparity map D_1 visualizes the lost consistency due to individual processing. $\mathcal{L}_2$ and $\mathcal{R}_2$ show our stereo consistent renderings and the resulting valid disparity map D_2.

Figure 6: *Stereo consistent posterization. As pointed out in the marked areas, the naive individual abstraction in the upper rendering lacks of consistency. Our results in the bottom row are stereo consistent. Source image: [Fro12]*

Colored ellipses mark the clearly visible differences between $\mathcal{L}'_{\text{individual}}$ and $\mathcal{R}'_{\text{individual}}$ in the posterization renderings of Figure 6. The individual abstractions in the top row contain contours of different shape and color, whereas the contours in our renderings (bottom row) are consistent.

The source photograph in Figure 7 was chosen for its high stereo quality. Only small glitches are observable, most obvious on the left and right image borders. The stroke mesh tends to become inconsistent due to small disparity imprecision. We attribute visible problems at depth discontinuities to both, incorrect depth estimations as well as averaging over foreground and background pixels. Further, this rendering demonstrates how our approach handles depth in reflections, e.g., where the building is mirrored in parts of the ponds surface. In scenarios with semi-transparent or refracting objects, 3D points of different depths blend together in one pixel. Our Central warp operates in image space with one depth value per pixel and reproduces such effects based on the computed disparity input. Thus, as long as the disparity is correctly detected, such view dependent effects are also reproduced correctly.

Figure 7: *SBR rotoscoping with paint texture of a stereoscopic photograph featuring stereo consistent reflections on the water surface. Source image: [Pow14]*

Temporal consistency: Our apporach of temporal consistency already includes considerable saving of computation time. Due to our motion vector warp, only small parts in a new frame need to be corrected. Besides, optical flow computation is relatively cheap compared to the retracing of a complete frame. Our footage for the stereoscopic rotoscoping involved smooth camera movement and various moving objects/ characters in the scene. Still, the necessary corrections caused fluctuation of about only 5% of all strokes. The majority of 95% was successfully warped each frame and needed no replacement. Figure 3 compares the naive *SBR*-animation approach (top row) with our successful stroke propagation results (bottom row). Some examples of the provided supplementary video material feature more temporal consistency than others. Most likely,

one will notice a difference between real camera source material and synthetic footage, which provides the more accurate motion estimations. Therefore, renderings from real footage often require more automated corrections. Too rapidly alternating corrections of the same region can become visible as uncovered canvas areas as well as strokes that fade in or out too quickly. Furthermore, if optical flow maps are fuzzy around object borders, strokes from the background will be moved along too and might cause disturbing halos around the moving objects.

Evaluation of stereo comfort: In a small user study, we evaluated the general comfort of our stereo consistency results on a 3D display with shutter glasses. Therefore, we generated random subsets containing original, stereo consistent, and stereo inconsistent versions of stereo images. 30 participants rated the test images in a certain subset on a 5 point comfort scale from 1 ("sick") to 5 ("super"). The original images scored $\mu = 4.55$ ($\sigma = 0.66$), the consistent renderings $\mu = 3.71$ ($\sigma = 0.95$), and the inconsistent versions $\mu = 2.10$ ($\sigma = 0.93$). In average, our stereo consistent *SBR* renderings show higher comfort values than the inconsistent versions of the test images. However, as expected, our stereo consistent results retrieved slightly lower comfort scores than the original images. Overall, the relatively high values of the consistent images indicate that our algorithm generates renderings with a very comfortable stereo appearance. Generally, we experienced disparity offsets $\geq 10\%$ of the image's width as very discomforting; in anaglyph as well as on autostereocopic displays.

Applicability: As mentioned before, our framework supports a powerful stereo solution for multiple styles of animation. Figure 1 compares the results of four different rendering styles. All are rendered stereo consistent with our framework. Further, Figure 6 illustrates a direct comparison of the posterization results. The upper row shows inconsistent individual abstractions of a stereoscopic image. The results in the bottom row were created with our stereo consistency approach. Colored ellipses denote obvious differences in the upper images which are resolved in the bottom row.

5.4. Performance and Limitations

As explained in Section 2.2, information from both stereo images is combined during the Central warp. Thus, in stereo-vision occurring dis- and occlusions are averaged as well. Figure 8 illustrates an example with high disparity at certain boundaries. Zooming in shows resulting ghosting artifacts in disoccluded areas. However, considering the whole image, those artifacts are barely noticeable.

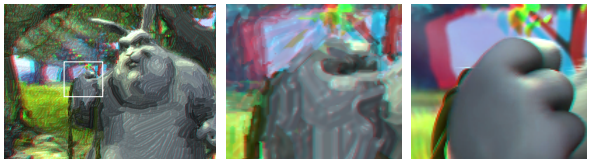


Figure 8: *Occlusion example. From left to right: Whole image, zoom in of example image, same zoom of source image. Despite small artifacts along the silhouette, the overall stereo appearance is maintained. Source images: [GR08]*

Figure 9 illustrates a comparison of our results to the stereo consistent *SBR* approach of Northam et al. [NAK12]. Although both *SBR* results show small artifacts in different parts of the image, they are comparable in terms of stereo consistency. Besides, contrarily to the layer based approach, our work also supports temporal consistency.

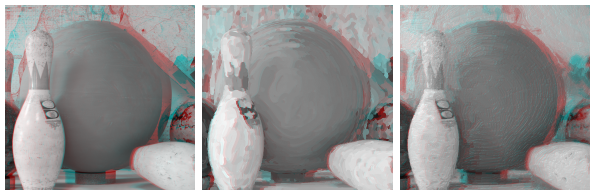


Figure 9: *Stereo consistency comparison. From left to right: Original image [HS07], result by Northam et al. [NAK12], our approach. Although both SBR results have some artifacts, they are comparable in stereo consistency.*

The depth accuracy of our rendering largely depends on the precision of the disparity estimation of the underlying material. Modern stereo correspondence algorithms still tend to contain areas with errors and inaccurate object boundaries. We share the position of Northam et al. [NAK12], that this field of computer vision still has potential for improvement. Achieved results could be improved by more accurate disparity estimations, as well as more precise optical flow maps. One can compare the results for synthetic and real source footage in our supplementary video material. Our *SBR* on synthetic footage moves more steady than on real footage because the motion estimation is more precise or based on ground truth data.

The implementation of our framework is realized in Python under use of OpenCV and NumPy. Performance is improved due to vectorizable NumPy arrays and CPU multithreading for the stroke extraction and warping. A complete *Full-HD* frame (*SBR* in 3840×1080p) is created in 120-180 seconds ($8 \times 2.9GHz$). This does not scale linearly for an animation, as the majority of strokes is reused. The extraction and warping process would benefit from a proper GPU port.

6. Conclusion

Our approach combines and extends existing methods for painterly image and video abstraction to automatically handled stereoscopic footage. Thereby, stereo and temporal consistency as well as visual comfort are maintained.

For stereoscopic rotoscoping we extract depth information from the stereoscopic image, merge both views into an average monoscopic view, abstract a single image, and warp

the abstracted information back into both views before rendering. Combined with improved temporal smoothing we can render painterly-style stereoscopic animations smoothly without major flickering nor unpleasing depth perception artifacts. The same framework can be easily applied to a range of different abstraction styles, as we demonstrated in some examples. More cartoonish styles, vectorization or even interactive rotoscoping techniques could be realized on stereoscopic footage with our stereo framework.

As improvement to existing *SBR* methods, our introduced dual stroke construction decreases the workload of necessary refinements by increasing the achieved precision. A more accurate representation of strong edges and fine details also allows the rendering of stroke meshes in the same size of the source image, without the mandatory upscaling.

References

[Aal12] AALBERS I.: Big Meat Flip 2014, 2012. [Online; accessed June-2015]. URL: https://vimeo.com/107837219. 4

[BGRR09] BLEYER M., GELAUTZ M., ROTHER C., RHEMANN C.: A stereo approach that handles the matting problem via image warping. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* (2009), IEEE, pp. 501–508. 2

[CRH05] COLLOMOSSE J. P., ROWNTREE D., HALL P. M.: Stroke surfaces: Temporally coherent artistic animations from video. *Visualization and Computer Graphics, IEEE Transactions on 11*, 5 (2005), 540–549. 2

[Far03] FARNEBÄCK G.: Two-frame motion estimation based on polynomial expansion. In *Image Analysis*. Springer, 2003, pp. 363–370. 4

[Fle17] FLEISCHER M.: Method of producing moving-picture cartoons, Oct. 9 1917. US Patent 1,242,674. 2

[Fro12] FROST J.: Mustang show car, 2012. [Online; accessed June-2015]. URL: https://flic.kr/p/cdr6dd. 6

[GR08] GOEDEGEBURE S., ROOSENDAAL T.: Big Buck Bunny, 2008. Blender Foundation. URL: http://www.bigbuckbunny.org/. 1, 6, 7

[Hae90] HAEBERLI P.: Paint by numbers: Abstract image representations. In *ACM SIGGRAPH Computer Graphics* (1990), vol. 24, ACM, pp. 207–214. 2

[Hau01] HAUSNER A.: Simulating decorative mosaics. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), ACM, pp. 573–580. 2, 5

[HE04] HAYS J., ESSA I.: Image and video based painterly animation. In *Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering* (2004), ACM, pp. 113–120. 2, 4

[Her98] HERTZMANN A.: Painterly rendering with curved brush strokes of multiple sizes. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques* (1998), ACM, pp. 453–460. 2, 4

[Her02] HERTZMANN A.: Fast paint texture. In *Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering* (2002), ACM, pp. 91–ff. 2, 5

[Hir08] HIRSCHMULLER H.: Stereo processing by semiglobal matching and mutual information. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 30*, 2 (2008), 328–341. 3

[HMD37] HAND D., MORGAN M., DISNEY W.: Snow White and the Seven Dwarfs, 1937. Produced by Walt Disney Productions. 2

[HP00] HERTZMANN A., PERLIN K.: Painterly rendering for video and interaction. In *Proceedings of the 1st international symposium on Non-photorealistic animation and rendering* (2000), ACM, pp. 7–12. 2, 4

[HS07] HIRSCHMÜLLER H., SCHARSTEIN D.: Evaluation of cost functions for stereo matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on* (2007), IEEE, pp. 1–8. 7

[KWL13] KIM Y., WINNEMÖLLER H., LEE S.: WYSIWYG Stereo Painting. In *Proceedings of ACM Symposium on Interactive 3D Graphics and Games 2013* (Mar. 2013). 2

[LHW*10] LANG M., HORNUNG A., WANG O., POULAKOS S., SMOLIC A., GROSS M.: Nonlinear disparity mapping for stereoscopic 3d. *ACM Transactions on Graphics 29*, 4 (2010), 75. 2

[Lit97] LITWINOWICZ P.: Processing images and video for an impressionist effect. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques* (1997), ACM Press/Addison-Wesley Publishing Co., pp. 407–414. 2

[LMY*13] LIU X., MAO X., YANG X., ZHANG L., WONG T.-T.: Stereoscopizing Cel Animations. *ACM Transactions on Graphics 32*, 6 (Nov. 2013), 223:1–223:10. 2

[NAK12] NORTHAM L., ASENTE P., KAPLAN C. S.: Consistent stylization and painterly rendering of stereoscopic 3D images. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering* (2012), pp. 47–56. 2, 7

[OBB*13] ORZAN A., BOUSSEAU A., BARLA P., WINNEMÖLLER H., THOLLOT J., SALESIN D.: Diffusion curves: a vector representation for smooth-shaded images. *Communications of the ACM 56*, 7 (2013), 101–108. 5

[PGB03] PÉREZ P., GANGNET M., BLAKE A.: Poisson image editing. In *ACM Transactions on Graphics (TOG)* (2003), vol. 22, ACM, pp. 313–318. 5

[Pow14] POW M.: Photograph Bistra Cathusian Monastery, 2014. [Online; accessed June-2015]. URL: http://phereo.com/image/534a0b57cb85777803000166. 6

[ROD*10] RICHARDT C., ORR D., DAVIES I., CRIMINISI A., DODGSON N. A.: Real-time Spatiotemporal Stereo Matching Using the Dual-Cross-Bilateral Grid. In *Proceedings of the European Conference on Computer Vision (ECCV)* (Sept. 2010), pp. 510–523. 3

[RŚDD11] RICHARDT C., ŚWIRSKI L., DAVIES I. P., DODGSON N. A.: Predicting stereoscopic viewing comfort using a coherence-based computational model. In *Proceedings of the International Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging* (2011), ACM, pp. 97–104. 2

[Sab97] SABISTON B.: Rotoshop, 1997. [Online; accessed June-2015]. URL: http://www.flatblackfilms.com/Flat_Black_Films/Rotoshop.html. 2

[SG05] STAVRAKIS E., GELAUTZ M.: Stereoscopic painting with varying levels of detail. In *Electronic Imaging 2005* (2005), International Society for Optics and Photonics, pp. 450–459. 2

[Tak09] TAK1701D: Photograph of Tadao Ando's Hyogo Prefectural Museum of Art in Kobe, 2009. [Online; accessed June-2015]. URL: http://commons.wikimedia.org/wiki/File:Hyogo_prefectural_museum_of_art16_2000.JPG. 5

[WOG06] WINNEMÖLLER H., OLSEN S. C., GOOCH B.: Real-time video abstraction. *ACM Transactions On Graphics (TOG) 25*, 3 (2006), 1221–1226. 2