

Split-Depth Image Generation and Optimization

Jingtang Liao^{†1}, Martin Eisemann² and Elmar Eisemann¹

¹Delft University of Technology, The Netherlands
²TH Köln, Germany

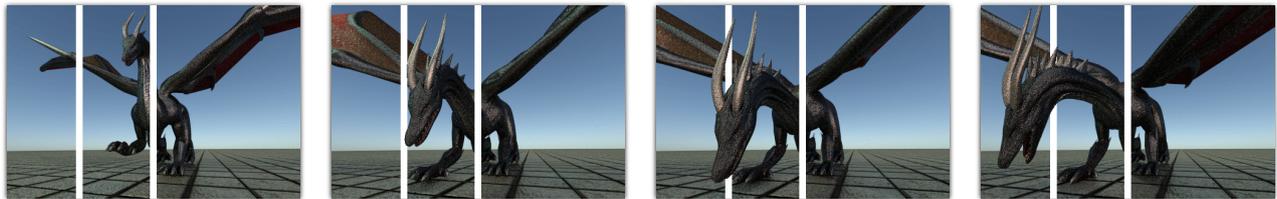


Figure 1: Split-depth frames over time generated by our approach. Via an occlusion cue, split-depth images can induce a 3D effect.

Abstract

Split-depth images use an optical illusion, which can enhance the 3D impression of a 2D animation. In split-depth images (also often called split-depth GIFs due to the commonly used file format), static virtual occluders in form of vertical or horizontal bars are added to a video clip, which leads to occlusions that are interpreted by the observer as a depth cue. In this paper, we study different factors that contribute to the illusion and propose a solution to generate split-depth images for a given RGB + depth image sequence. The presented solution builds upon a motion summarization of the object of interest (OOI) through space and time. It allows us to formulate the bar positioning as an energy-minimization problem, which we solve efficiently. We take a variety of important features into account, such as the changes of the 3D effect due to changes in the motion topology, occlusion, the proximity of bars or the OOI, and scene saliency. We conducted a number of psycho-visual experiments to derive an appropriate energy formulation. Our method helps in finding optimal positions for the bars and, thus, improves the 3D perception of the original animation. We demonstrate the effectiveness of our approach on a variety of examples. Our study with novice users shows that our approach allows them to quickly create satisfying results even for complex animations.

CCS Concepts

•Computing methodologies → Image processing; Perception;

1. Introduction

Preserving or even enhancing the 3D impression of a scene on a 2D display can be a powerful means to attract attention, amplify scene layout, and enhance scene understanding, yet, it is difficult to achieve. Artists throughout the centuries developed techniques on how to use effective pictorial (or monocular) cues to enhance the depth perception on a canvas.

Occlusion is one of the strongest cues of the human visual system to interpret depth ordering. In consequence, it is also one of the main factors to exploit when conveying a 3D arrangement. One creative solution to exploit this effect for paintings is the use of a passepartout (a paper, more usually, cardboard sheet with a cutout).

We are so accustomed to a passepartout being not part of the image itself that incorporating it into the actual painting leads to a surprisingly convincing 3D effect, e. g., in "Escaping Criticism" by Pierre Borell del Caso as shown in Figure 2. With digital media, virtual passepartouts have become a popular variant for photography and static virtual scenes [SCRS09] [RTMS12]. The resulting occlusion effect separates the image into a front and back layer, which produces a strong "popping out" sensation or a "floating on the window" illusion. Split-depth images utilize similar reference spaces, usually bars (but we will use the general term *splits* throughout the paper), to increase the 3D effect of a short animation or movie clip. They have recently gained in popularity and are employed by an increasing number of companies to catch the consumer's attention and interest. This paper will present a novel algorithm to help in the generation of split-depth images.

[†] j.liao@tudelft.nl



Figure 2: The use of reference spaces in paintings, images and animations increases 3D impression. a) "Escaping Criticism" by Pierre Borell del Caso; b) Out of bounds photography [OOB15]; c) Virtual passepartouts [VP09]; d) Split-depth image.

As for passepartouts, splits induce a plane in the virtual scene, creating a division between the mental fore- and background. If an object overlaps this plane, it is considered in front. The same holds for animations where this information is usually interpreted as an object moving out of the image towards the viewer.

Currently, the generation of such split-depth images is a purely manual and time-consuming task relying on image editing tools to segment each image in an animation and add the splits. Further, there are no known rules for producing the effect and choices were made in an adhoc manner, although the quality of the resulting animations relies heavily on several factors such as position, width, scene content, physical correctness, etc. In consequence, designing split-depth images is a tedious and time-consuming task, which resulted in many low-quality examples on the internet.

In our work we propose an approach to automatically create split-depth images using an RGBD (color plus depth) image sequence as input. We investigate the possible factors, which contribute to the enhanced 3D perception and build a computational model to automatically generate splits that lead to a convincing result. Overall, we make the following contributions:

- A perceptual study to investigate the contributions of different factors to the split-depth illusion;
- A multi-objective split-optimization procedure respecting various perceptual cues, such as occlusion, split proximity, and scene saliency;
- A framework to support the split-depth image generation.

2. Related work

In this section, we will briefly discuss the related work of optical illusions in relation to depth.

Optical illusions. The research of optical illusions has a long history in vision science. Michael Bach provides a vast collection of

optical illusions on <http://www.michaelbach.de/ot/>, which use different perceptual cues such as motion (dotted line [IAC09], reverse Phi illusion [AR86]), luminance and contrast (Hermann grid [Spi94], the pyramid illusion [RMR83]), color (color fan [ZECL12] [RE12]), geometric (Zöllner illusion, disjointed arch), size constancy (moon illusion [RP02]), etc. Gregory et al. [Gre97] classify the phenomena of illusions into four main classes: ambiguities (Necker cube), distortions (Ponzo figure), paradoxes (Tribar impossible figure) and fictions (Kanizsa square). Among them, some optical illusions already have a long history, whose mechanisms are well studied while others still lack a successful explanation [Oli06]. It has been only a short time that split-depth illusions are produced and little investigation was done in automating this process.

Occlusion depth cue. Depth perception helps us perceive the world in 3D and there are various kinds of depth cues, which are typically classified into binocular cues and monocular cues. Without 3D devices, we typically encounter monocular cues in animations - depth information that can be perceived with just one eye. Motion parallax [KDR*16] [LSE17], size, texture gradient [BL76], contrast, perspective, occlusion [PBL07], and shadows [BG07] are examples of these. Occlusion is a particularly strong depth cue [Cut95], which can be used for various purposes, such as depth recovering [SCN08] [SSN07], or depth enhancement. In this work, we focus on the latter one. Ritschel et al. [RTMS12] provided a framework to improve the perceived 3D effect by adding a virtual passepartout to RGBD images. Later, Zheng et al. [ZZS13] extended this work by incorporating scene saliency into the optimization. A similar work [SCRS09] presented an intuitive user interface for fast "Out of Bounds" prototyping by adding 3D frames to 2D photographs. These approaches are mostly restricted to static images. Finding the optimal splitting plane in animations is difficult and often error prone as the object's motion is in general unrestricted and the 3D impression is quickly reduced by a non optimal placement. Nonetheless, well-placed simple splits in the form of horizontal or vertical bars can provide a strong occlusion depth cue. In this paper, we investigate the creation of split-depth images and various factors that contribute to their effectiveness.

3. Overview

Our framework is illustrated in Figure 3. Given an RGBD image sequence, a mask to encode the OOI, and a choice for the number of splits and their width, we seek to generate optimal splits (potentially, in combination with virtual passepartouts). Video input with depth has become wide-spread (e.g., Kinect). In this case, an OOI can be extracted using segmentation and rotoscoping.

The core of our solution builds upon an energy optimization. To this extent, we first conduct a few psycho-visual experiments to validate assumptions that we will then integrate into our energy formulation (Sec. 4). We summarize the motion of the OOI through space and time (Sec. 5.1), which serves as a hard constraint and basis of our approach. Given these elements, we build a formulation for the optimal positioning of splits (Sec. 5.2). We then demonstrate our approach on various examples and show its effectiveness via an evaluation with novice users (Sec. 6) on complex datasets, before concluding (Sec. 7).

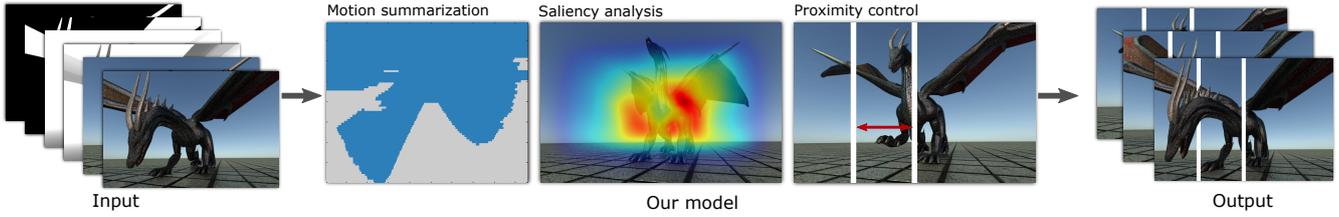


Figure 3: Overview: Given an RGBD image with a mask channel to indicate an object of interest, we summarize the motion spatially and temporally. This summary, together with other factors, such as saliency and proximity, will guide the split finding process.

4. Preliminary study

In this section, we study the assumptions that will guide our optimization. Throughout our experiments, unless otherwise stated, we used two splits, as it is the most common setup found in the many examples that are presented online, although there are a few cases where the number of splits varies or non-vertical bars are used. We formulate the following hypotheses, which we base on our observations from various online examples:

- Preference for split width varies on an individual basis;
- Splits should be at the same depth layer, otherwise the result might seem implausible;
- People prefer splits with a narrower gate (bar distance to the object is smaller);
- Main features in the scene should not be obstructed by the splits.

For the first assumptions, we deliberately avoid content influence. We thus investigate scenes using abstract cubes before testing two more complex scenes. In total, we performed four experiments in this preliminary study and involved 45 users with normal or corrected-to-normal vision. During the training session, we introduce split-depth images to the participants by showing them some previously collected examples. Once the concept was clear, we started the experiments. To avoid absolute scales for preference, we used a forced-paired comparisons [MTM12] [LCTS05]. Here, a preference choice has to be made between two shown exemplars. We avoid biased results by randomizing the tests.

Experiment 1: Testing preference for split width. We used two different scenes from which we created three split-depth images with different split width (small (2% screen width), middle (4% screen width), and big (8% screen width)) and tested two different scenes. For each pair, we asked participants to choose the one which looks as if the object is moving closer to them.

The result is recorded in the preference matrix shown in Table 1. In total, it records $270 = 45 * 3 * 2$ evaluations, used as a *Score*. The numbers indicate the number of times that the corresponding image sequence was preferred. E.g., the cell in row *Small*, column *Middle* has a value of 52 indicating that 52 times *Small* was preferred to *Middle* in a direct comparison. The results are not entirely conclusive, even though the splits with the biggest width score the highest, there is no clear decreasing or increasing tendency shown as the split width increases. Our assumption that width is based on personal preference seems thus valid. In consequence, we made the split width a user-defined parameter.

Table 1: Preference matrix for split width.

	Small	Middle	Big	Score
<i>Small</i>	—	52	35	87
<i>Middle</i>	38	—	34	72
<i>Big</i>	55	56	—	111

Experiment 2: Splits should share the same depth to avoid an implausible appearance. Again, we used two scenes, one with splits at the same depth and one with different depth layers but with the same distance of the splits in 3D. Note that due to perspective foreshortening, the distance between the splits that are located at different depths appears smaller. For each pair, we asked, which one is more plausible. The result is illustrated in Figure 4. 74 out of 90 (binomial test, $p < .000001$) choices favor the ones, where splits are placed at the same depth layer. Even though these results are significant, we cannot exclude that other factors, such as motion direction, might influence a user’s preference, which should be investigated in further experiments.

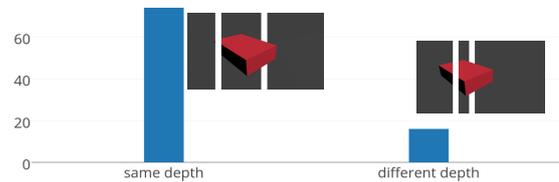


Figure 4: Preference comparison between bar placed at the same depth and different depth.

Experiment 3: Narrower gates are preferred. We used two scenes and created three split-depth image sequences each, where the opening between the gates ranges in width (narrow (around 15% screen width), middle (around 30% screen width), wide (around 45% screen width)), respectively. Again, participants perform 6 pair comparisons and were asked to choose the one, which they perceive as having a stronger depth.

The result is recorded in Table 2. The study shows that people preferred narrower gates as shown in the last column - the score increases when bars get closer. While not entirely conclusive, the results indicate that placing the splits as close as possible to the OOI is generally preferable if no other factors, such as scene content influence the perception, see Experiment 4.

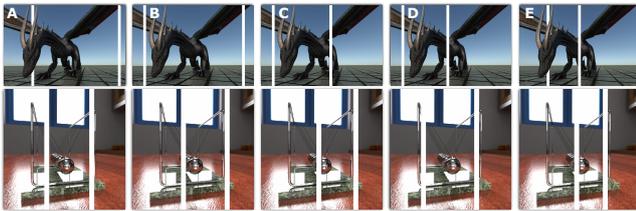
Table 2: Preference matrix for split proximity.

	Narrow	Middle	Wide	Total
<i>Narrow</i>	—	54	56	110
<i>Middle</i>	36	—	57	93
<i>Wide</i>	34	33	—	67

Experiment 4: Main features should not be obstructed by bars.

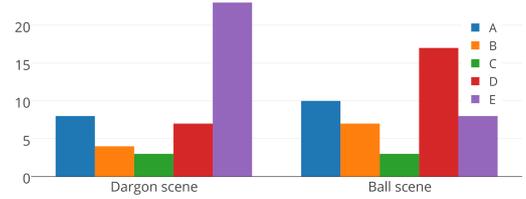
We do want to maintain the visibility of the main features in the animation. In practice, these would have to be estimated or otherwise derived (e.g., eye tracking). This constraint may conflict with the expected preference for a narrow opening between splits. In this study we want to validate that in some cases proximity of splits are more important than the scene saliency and vice versa. We present two scenes (Dragon, Balls) and placed the bars at five different locations having different proximity of the bars and covering differently salient regions, Fig. 5. For each animation, we generated five split-depth image sequences and inquired regarding preference. As there are many factors that can play a role, we asked for the reasoning via a textbox.

As shown in Figure 6, for the dragon scene 23 out of 45 (binomial test, $p < .000001$) participants chose the version with the narrow gate, whereas for the ball scene 17 out of 45 (binomial test, $p = .002594$) chose the one where the main object is more visible. It is important to note that for the dragon scene, which has a simplistic background, 8 participants mentioned in their reasoning that they preferred the increased depth perception due to the narrow gates. For the ball scene, which has a more complex background, 12 participants mentioned that they preferred a wider gate not because it provided a stronger 3D impression but because the narrow gate occluded salient parts of the scene. These findings illustrate the complexity of the problem, as it indicates that it is scene-dependent. The preferred balance between covered salient elements and proximity of the bars can thus vary. For this reason, our approach lets the user determine the balancing between these two factors, e.g., if salient parts are hidden the impact of saliency preservation can be adjusted by increasing the according parameter in our framework. Future studies with more diverse scenes could give more insight into the impact of individual features on the perceived result.


Figure 5: Scenes (dragon and ball scenes) used in experiment 4.

5. Our approach

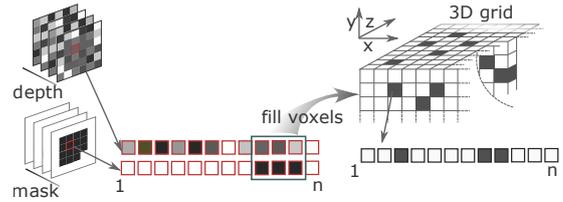
We will now use the findings of our preliminary study to develop an optimization procedure for the placement of splits. First, in Sec. 5.1, we will explain our motion summarization, which is used to ensure that no intersections between the scene and the splits occur. Then we derive an energy formulation that is used in the


Figure 6: Votes of people's preference in experiment 4.

actual optimization process in Sec. 5.2. The input to our framework are a set of RGB images ($\mathbf{I}_1, \mathbf{I}_2, \dots, \mathbf{I}_n$), as well as corresponding depth images ($\mathbf{D}_1, \mathbf{D}_2, \dots, \mathbf{D}_n$), and masking images of the OOI ($\mathbf{M}_1, \mathbf{M}_2, \dots, \mathbf{M}_n$). All images have size $w \times h$. Two user-defined parameters that will be involved in the optimization immediately are the number N and width w_b of the splits.

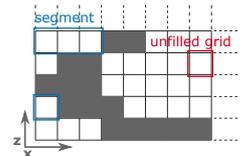
5.1. Motion summarization

Motion in split-depth images can be arbitrary and parts or entire objects might change. Consequently, object centroids or similar approximations will not well characterize the animation. Instead, we summarize the spatial and temporal information via a 3D histogram approach.


Figure 7: Motion summarization scheme. For each voxel inside the 3D grid, we record the contained depth values of the OOI.

As shown in Figure 7, we discretize our scene into a 3D voxel grid with dimensions of $w \times h \times k$, where k is a number of bins between the object's minimum and maximum extent along the z direction during the animation. As the intersection with the static scene would be simple to test and it is not as disturbing as intersecting a moving object, we typically only insert the depth values of the OOI into this 3D grid. All unfilled voxels indicate room to add potential splits. In consequence, it becomes possible to enumerate all options and test an energy function, derived in Sec. 5.2.

To accelerate computations, we rely on a strategy from path finding and compute the Minkowski sum between the inversed 2D split shape and the 3D grid. Voxels that remain empty after this convolution will exactly correspond to valid positions of the split [dBvKOS00]. If the splits are represented by axis-aligned bars, we can project the entire grid on this axis and reduce the problem's dimensionality. To facilitate explanations, we will assume this case in the following. The figure to the right illustrates the resulting representation. The Minkowski sum



is similar to a convolution and has the positive side effect to reduce noise. Figure 8 illustrates this effect for different split widths.

To enumerate all possible split configurations, we should remember our findings from Sec. 4. Splits should form gates through which the OOI moves in order to induce a 3D effect. Additionally, these splits should share the same depth. In consequence, if we slice the motion summarization grid with a plane at a certain depth, the splits should separate the OOI intersections with this plane.

For example, if we assume the user indicated that two splits should be used and a given depth layer leads to three connected regions that indicate potential placements, $(\mathbf{A}, \mathbf{B}, \mathbf{C})$, we will test the combinations (\mathbf{A}, \mathbf{B}) , (\mathbf{A}, \mathbf{C}) , and (\mathbf{B}, \mathbf{C}) . Generally, there will be $\binom{K}{N} = \frac{K!}{(K-N)!N!}$ possible solutions, where N is the number of splits and K the number of regions for possible placements. If a given depth does not allow for the user defined number of splits to form gates, we can proceed to the next discrete depth level. A last condition is that splits need to form gates, which the OOI traverses. To ensure this condition, we can test the OOI against each split along the sequence and two consecutively overlapped splits form a gate if the OOI is in front of the first, then behind the second split, or vice versa. Only if all splits form gates, the configuration is tested.

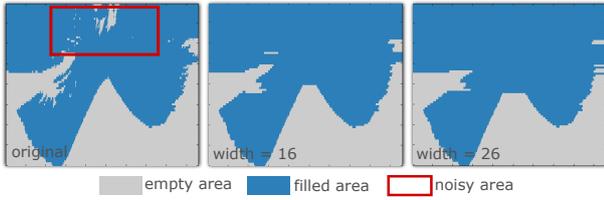


Figure 8: Larger bars lead to fewer potential placements, but reduce noise in the sequence, as highlighted in the red rectangle.

5.2. Split Optimization

We define an energy functional to encode and optimize various energy terms, denoted as E_{fb} , E_t , E_p , and E_s :

$$\min(\lambda_{fb}E_{fb} + \lambda_t E_t + \lambda_p E_p + \lambda_s E_s). \quad (1)$$

E_{fb} , E_t relates to the visibility of the occlusion by the splits, E_p to the proximity of the splits, and E_s to the saliency in the scene. By default, we propose the parameters $\lambda_{fb} = 1.0$, $\lambda_t = 0.1$, $\lambda_p = 0.5$, $\lambda_s = 0.5$, which work usually well in practice. Nonetheless, the user has the possibility to adjust settings, as Sec. 4 showed that some elements, such as preference for proximity and covering of salient elements vary from scene to scene and individual to individual. This energy functional is optimized by evaluating various split configurations.

Occlusion cue. Occlusion is key in producing the depth effect. In consequence, a user might want to make sure that the occlusion is well noticed by the observer. We translate this condition into how many pixels the OOI is actually in front or behind the given splits over the duration of the video.

To compute this result, we calculate the number of pixels T_i of the OOI that overlap with the split i for the current configuration.

Let T_{max} be the maximal number among all splits, then we define the energy:

$$E_{fb} = \sum_i 1 - |T_i/T_{max}| \quad (2)$$

A related energy E_t , for trailing, will ensure that the OOI is not hidden in the beginning of the sequence, as it will otherwise not be visible to the observer and the frames would be useless for the animation. As not all sequences will allow us to fulfill a hard time constraint, we formulate this condition as an energy as follows:

$$E_t = \sum_i 1 - \min(V/U, 1), \quad (3)$$

where U is a user-defined constant (the desired number of frames that the OOI is not occluded by the split), V is a constant (the actual number of frames that the OOI is visible, thus, not occluded). A similar condition can be added for the end of the sequence, if no overlap is wanted.

Saliency cue. Main features of the scene, even if they are not part of the OOI, should not be blocked by the splits. As importance is difficult to derive, we use a mean saliency term as an estimate. To this extent, we compute the saliency of all input RGB frames and sum the contributions [HKP06], Fig. 9. The energy is then defined as:

$$E_s = \sum_i S_i \quad (4)$$

where S_i is the integrated mean saliency distribution underneath split i . In other words, split placements will be preferred that cover less salient regions.

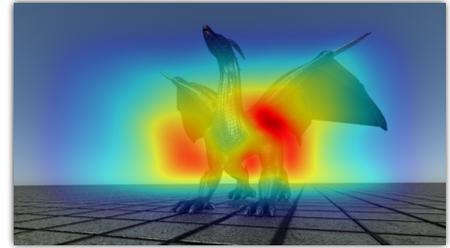


Figure 9: Mean saliency map for the input RGB frames overlaid with a single frame from the input video for visualization purposes.

Proximity cue To integrate a preference for narrower gates, as was investigated in our study in Sec. 4, we measure the distance between the two neighboring splits and encode it in the following energy formulation:

$$E_p = \sum_i D_i \quad (5)$$

where D_i is the distance between two neighboring splits (for parallel bars, it is just their distance, for general splits, one could use the Hausdorff distance). The terms are normalized by the screen width. As the splits are not allowed to intersect with the OOI there is a natural lower bound for D_i , namely the width of the OOI. If the user wants to only use one bar, this term is ignored.

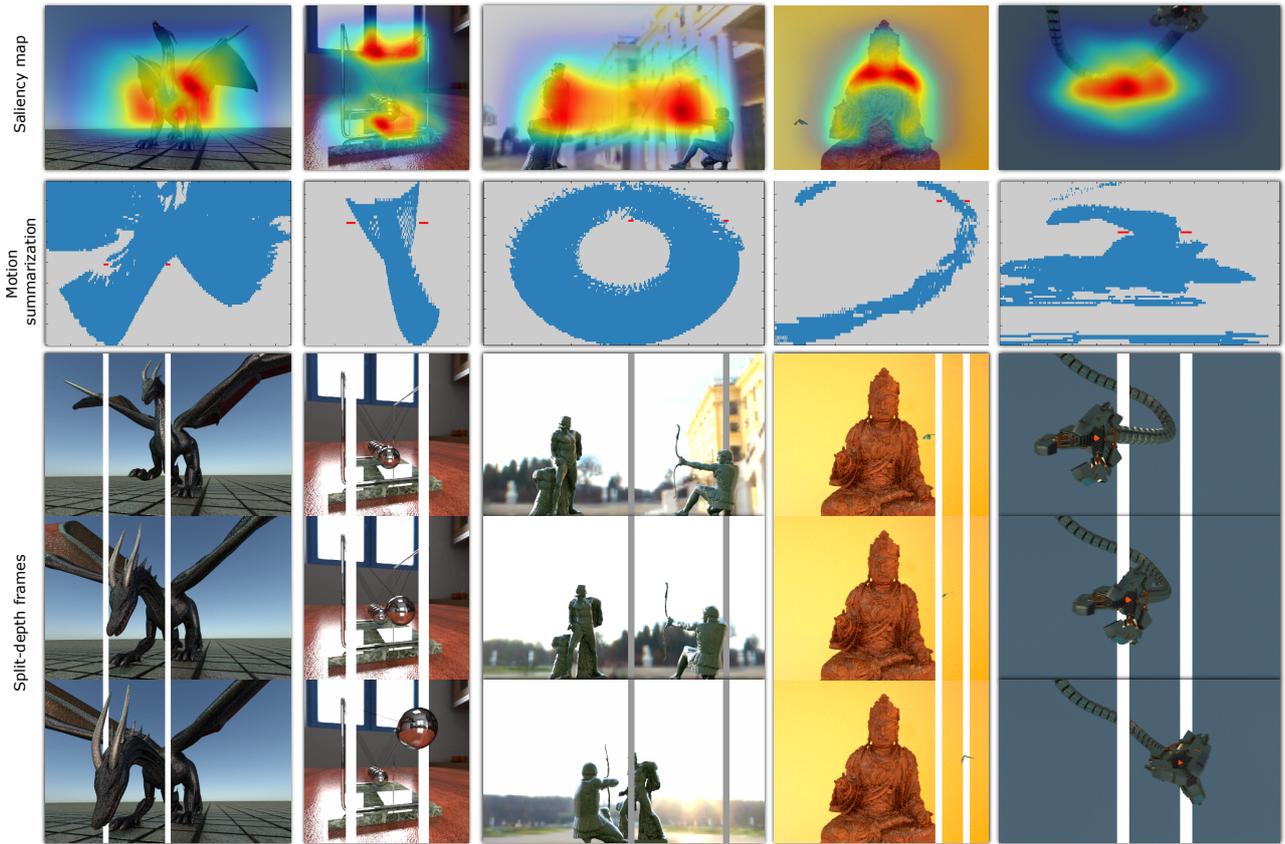


Figure 10: Examples of our approach on various data sets. Please refer to the supplemental videos for the animations. From top to bottom, row 1: mean scene saliency map; row 2: projection of the motion summarization with optimal positions for splits in red; row 3 - 5: example split-depth frames.

6. Results and discussion

We have implemented our framework in Matlab on a desktop computer with an Intel Core i7 3.7 GHz CPU. We did not optimize the code for performance. The timing for the motion summarization is linear in the number of input images, ranging from a few seconds to several minutes, but could be easily parallelized. The optimization to look for the bar positions can be done within a few seconds.

6.1. Split-depth GIFs results

We demonstrate our results on a range of scenes. Please refer to our supplemental material for several split-depth images, of which we show a few example frames in Figure 10 (row 3 - 5). We also visualize the final optimized position on the motion summarization map, Figure 10 (row 2). We created scenes with similar motion to the most popular split-depth gifs currently available. Interestingly the artists avoid much clutter in the background or strong camera motion. Strong camera motion would also contradict with the static positioning of the splits and affect depth perception.

The improvement using split-depth images is diminished if white splits are used in front of a bright background. We, therefore, offer the option to adapt the color of the splits to different gray-scales,

Fig. 11. The user is also free to choose the number of splits for each scene, although 2 is the default. 1 or 3 is rarely needed and we encountered no case, where more than 3 was beneficial. A useful maximal number of splits can be derived directly from the topology of the summarization.



Figure 11: Adaption of color and number of splits.

An interesting 3D effect can also be achieved by combining the split-depth images with virtual passepartouts. Our system automatically proposes to enlarge the bar towards the image boundary if the integrated saliency is comparably low (per default the splits should not cover more than 15% of the total saliency in the image), Fig. 12.

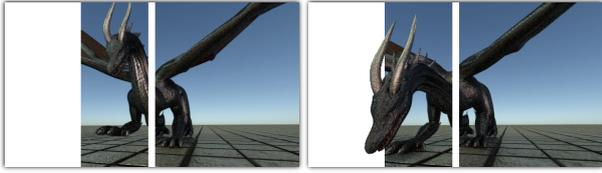


Figure 12: Combination using bars and virtual passepartouts

6.2. User validation

To test the applicability of our model in scenes with comparably complex animations, we have performed a validation user study to compare our automatic method with manual split placement. As it

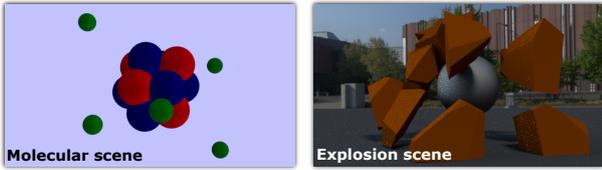


Figure 13: Example frames of two data sets we used in our validation study.

is a young artform, professional split-depth image artists are scarce. Additionally, as our work targets users with little experience, we conducted our study with three users, novice to creating split-depth images but with differing knowledge on image editing. Their task was to manually place the splits in the 3D scenes shown in Fig. 13. Occlusion was then automatically derived from the corresponding depth values. Before the study, we showed the participants examples of well done split-depth images. During the experiment, they could position the splits freely in 3D until they were pleased with the results. Only vertical splits were allowed. There was no time limitation throughout the whole study. During the experiment, we kept track of the number of split position adjustments and the result is shown in Table 3. It is worth mentioning that, in all test sequences, participants had to experiment with several positions to get to their final result.

Table 3: Number of bar adjustments.

	User A	User B	User C
Explosion scene	6	8	8
Molecular scene	10	20	17

Figure 14 row 1 shows the final selected position of each participant and the result of our algorithm in the scene summarization. Row 2 to 5 depicts some example frames for each user. In most of the users' results, the scene objects penetrated the splits resulting in implausible scene constellations, whereas our framework found acceptable positions.

To further verify if our framework can generate more appealing results than those created by novice users, we conducted an

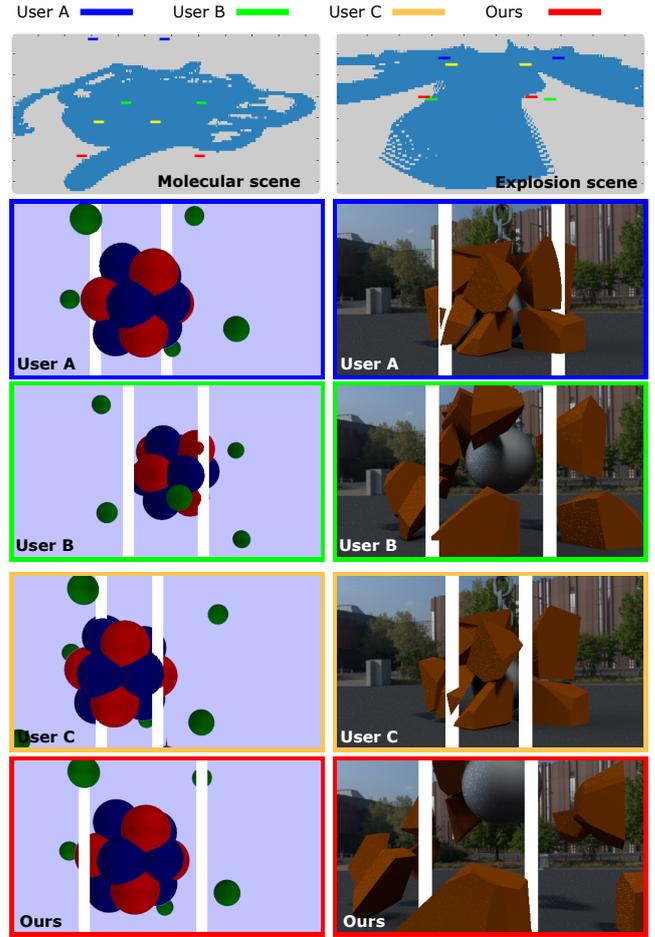


Figure 14: Results of novice users and example frames. Note the intersections in the results of novice users.

additional user study with 13 subjects with normal or corrected-to-normal vision and asked for their preference by showing them the results of the novice users and ours. The user study setup is similar to that in Sec. 4. Figure 15 shows the results of people's preference. For both datasets, 7 out of 13 (binomial test, $p = .018686$) subjects liked ours best. The other results were comparably similar and no clear preference exists.

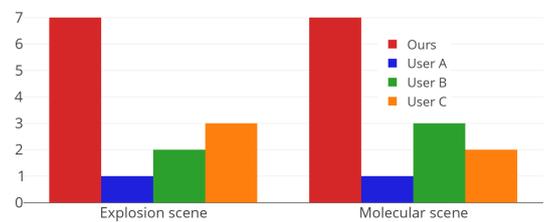


Figure 15: Comparison of people's preference between novice users' results and ours.

7. Conclusion and future work

We presented an algorithm for automatic split-depth image creation from RGBD image sequences, which takes important factors, such as spatial and temporal motion information, scene saliency and proximity of the splits to the object of interest into account. We also provided means to manually set and test different parameters, such as color and width of the splits, while the succeeding optimization is fully automatic, which enables an easy exploration of effective solutions. We validated the importance of these factors through a preliminary study and demonstrated the usefulness of our presented model and the optimization in a second user study.

Our method is subject to certain limitations. Imperfect masking or depth-of-field in the animation, invalidate our current input assumptions. Using natural image matting techniques, one could separate fore- and background, although (especially for videos) these techniques are still highly error-prone and require substantial manual effort. As mentioned in Sec. 6.1, scenes with much clutter in the background or strong camera motion are often avoided by split-depth image artists. Complex motion in itself is not a problem for our algorithm. However, many objects with complex motion can potentially lead to a case where the motion summarization is filled and our algorithm fails to find good positions for the splits. This is, however, no real limitation, as it simply implies that there is no possible intersection-free position for the splits. In fact, it can be seen as a benefit of our algorithm, as it tells the user that the scene in its current form is not well suited for an effective split-depth image. A solution might be non-linear splits (e. g., circle, ellipse) or animated splits which fade in/out or move throughout the animation but this is left for future work as the effect on depth perception is unclear. Another fruitful direction for further research is the extension of our approach to plain RGB image sequences without depth information.

Acknowledgments We thank the anonymous reviewers for their insightful comments and subjects who participated in our perceptual studies. We would also like to show our gratitude to Tim Schürheck for designing the animations in the used datasets. We are also immensely grateful to free3d artist: Dennis Haupt and Blender-Swap artists: ColeHarris, clive9, nerotbf and TheTemporalist for providing the datasets.

References

- [AR86] ANSTIS S. M., ROGERS B. J.: Illusory continuous motion from oscillating positive-negative patterns: implications for motion perception. *Perception* 15, 5 (1986), 627–640. 2
- [BG07] BRUCKNER S., GRÖLLER E.: Enhancing depth-perception with flexible volumetric halos. *IEEE Transactions on Visualization and Computer Graphics* 13, 6 (2007), 1344–1351. 2
- [BL76] BAJCSY R., LIEBERMAN L.: Texture gradient as a depth cue. *Computer Graphics and Image Processing* 5, 1 (1976), 52–67. 2
- [Cut95] CUTTING J. E.: Potency, and contextual use of different information about depth. *Perception of space and motion* (1995), 69. 2
- [dBvKOS00] DE BERG M., VAN KREVELD M., OVERMARS M., SCHWARZKOPF O.: Computational geometry: Algorithms and applications. Second ed. Springer-Verlag, 2000, ch. 13, pp. 290–297. URL: <http://www.cs.uu.nl/geobook/>. 4
- [Gre97] GREGORY R. L.: Visual illusions classified. *Trends in cognitive sciences* 1, 5 (1997), 190–194. 2
- [HKP06] HAREL J., KOCH C., PERONA P.: Graph-based visual saliency. In *Proceedings of the 19th International Conference on Neural Information Processing Systems* (Cambridge, MA, USA, 2006), NIPS'06, MIT Press, pp. 545–552. 5
- [IAC09] ITO H., ANSTIS S., CAVANAGH P.: Illusory movement of dotted lines. *Perception* 38, 9 (2009), 1405–1409. 2
- [KDR*16] KELLNHOFFER P., DIDYK P., RITSCHER T., MASIA B., MYSZKOWSKI K., SEIDEL H.-P.: Motion parallax in stereo 3d: model and applications. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 176. 2
- [LCTS05] LEDDA P., CHALMERS A., TROSCIANKO T., SEETZEN H.: Evaluation of tone mapping operators using a high dynamic range display. In *ACM Transactions on Graphics (TOG)* (2005), vol. 24, ACM, pp. 640–648. 3
- [LSE17] LIAO J., SHEN S., EISEMANN E.: Depth map design and depth-based effects with a single image. In *Proceedings of Graphics Interface 2017* (2017), GI 2017, pp. 57–63. 2
- [MTM12] MANTIUK R. K., TOMASZEWSKA A., MANTIUK R.: Comparison of four subjective methods for image quality assessment. In *Computer Graphics Forum* (2012), vol. 31, Wiley Online Library, pp. 2478–2491. 3
- [Oli06] OLIVER S.: Optical illusions and their causes: Examining differing explanations. *AHS Capstone Projects*. 7. (2006). 2
- [OOb15] Out of bounds photography. <https://pixabay.com/en/out-of-bounds-image-editing-horses-940381/>, 2015. Accessed: 2017-07-01. 2
- [PBL07] PALMER S. E., BROOKS J. L., LAI K. S.: The occlusion illusion: Partial modal completion or apparent distance? *Perception* 36, 5 (2007), 650–669. 2
- [RE12] RITSCHER T., EISEMANN E.: A computational model of after-images. *Comp. Graph. Forum (Proc. Eurographics 2012)* 31, 2 (2012). 2
- [RMR83] RATLIFF F., MILKMAN N., RENNERT N.: Attenuation of mach bands by adjacent stimuli. *Proceedings of the National Academy of Sciences* 80, 14 (1983), 4554–4558. 2
- [RP02] ROSS H., PLUG C.: *The Mystery of The Moon Illusion-Exploring Size Perception*. 2002. 2
- [RTMS12] RITSCHER T., TEMPLIN K., MYSZKOWSKI K., SEIDEL H.-P.: Virtual passepartouts. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering* (2012), Eurographics Association, pp. 57–63. 1, 2
- [SCN08] SAXENA A., CHUNG S. H., NG A. Y.: 3-d depth reconstruction from a single still image. *International Journal of Computer Vision* 76, 1 (2008), 53–69. 2
- [SCRS09] SHESH A., CRIMINISI A., ROTHER C., SMYTH G.: 3d-aware image editing for out of bounds photography. In *Proceedings of Graphics Interface 2009* (2009), Canadian Information Processing Society, pp. 47–54. 1, 2
- [Spi94] SPILLMANN L.: The hermann grid illusion: a tool for studying human perceptive field organization. *Perception* 23, 6 (1994), 691–708. 2
- [SSN07] SAXENA A., SCHULTE J., NG A. Y.: Depth estimation using monocular and stereo cues. In *IJCAI* (2007), vol. 7. 2
- [VP09] Virtual passepartouts. <https://www.flickr.com/photos/29412527@N04/3905723380>, 2009. Accessed: 2017-07-01. 2
- [ZECL12] ZAIDI Q., ENNIS R., CAO D., LEE B.: Neural locus of color afterimages. *Current Biology* 22, 3 (2012), 220–224. 2
- [ZZS13] ZHENG Z., ZHANG Y., SUN Z.: Generating Pseudo-3D Painting Based on Visual Saliency and Composition Rules. In *Eurographics 2013 - Short Papers* (2013), Otaduy M.-A., Sorkine O., (Eds.), The Eurographics Association. 2