

Soft Edge and Soft Corner Blending

Marcel Lancelle*, Dieter W. Fellner*[†]

*Institute of ComputerGraphics and
KnowledgeVisualization (CGV)
TU Graz
Graz, Austria
E-Mail: cgv@cgv.tugraz.at

[†]GRIS, TU Darmstadt
& Fraunhofer IGD
Darmstadt, Germany
E-Mail: office@gris.tu-darmstadt.de

Abstract: We address artifacts at corners in soft edge blend masks for tiled projector arrays. We compare existing and novel modifications of the commonly used weighting function and analyze the first order discontinuities of the resulting blend masks. In practice, e.g. when the projector lamps are not equally bright or with rear projection screens, these discontinuities may lead to visible artifacts. By using first order continuous weighting functions, we achieve significantly smoother results compared to commonly used blend masks.

Keywords: Soft Edge Blending, Tiled Projection

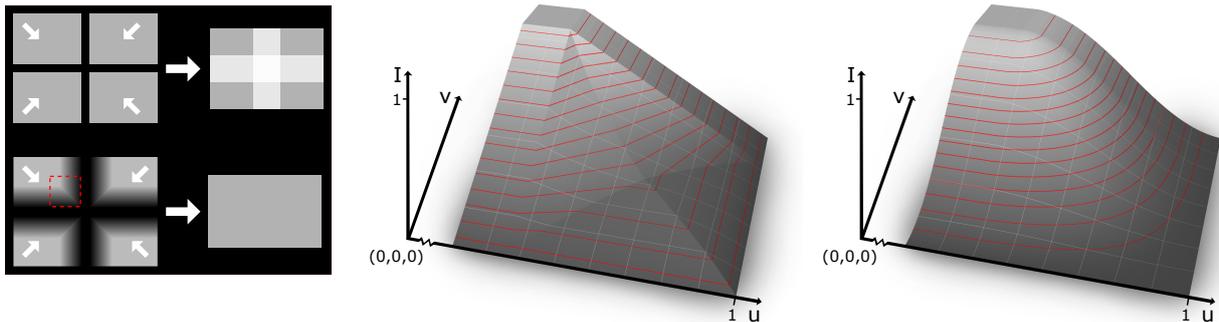


Figure 1: In a tiled overlapping projector setup a white image is displayed (top left). To hide the transitions, light is attenuated by a software blend mask (bottom left). A graph of the resulting light intensity I in a corner of one projector image (red dashed box) is shown for the commonly used blend mask (center, by Raskar et al. [RBY⁺99]) and for our approach (right), where (u, v) denote the pixel coordinates of the blend mask.

1 Introduction

Tiled projector arrays are usually set up so that the projected images overlap each other. This is done to avoid an exact mechanical calibration which can be very time consuming, and also to hide the transition, as even small brightness and color differences are easily visible due to the Mach Band effect. Non-Lambertian projection surfaces lead to an angle dependent brightness, resulting in brightness differences when viewed from an angle.

A drawback of blending by attenuating the light via the video signal is that projectors have a black level, i.e. they cannot show completely black. These black levels add up and

may be visible, especially at corners where four projector images meet. Physical masks close to the projector lens can also be used for blending to avoid this. While they result in the qualitatively best image, additional hardware is needed and usually the calibration is done manually.

The other commonly used method is software blending. This is usually done with a linear gradient instead of a hard edge. The goal is to get a visually smooth mask image that is multiplied with the final image to appropriately attenuate the light. However, commonly used methods lead to artifacts at regions where more than two projectors overlap, i.e. in the inner corners. We first explain these artifacts in more detail, then present and analyze existing and new methods to reduce such artifacts. Finally we list some implementation details.

In case of overlapping images, a geometric calibration is also necessary to correctly place the content.

2 Related Work

Raskar et al. showed automatic geometric calibration using a camera per projector [RBY⁺99]. They use soft edge blending and use the following equation to compute the attenuation value of projector m at each pixel position (u, v)

$$A_m(u, v) = \frac{\alpha_m(m, u, v)}{\sum_i \alpha_i(m, u, v)} \quad (1)$$

with a weight function

$$\alpha_m(m, u, v) = w_i(m, u, v) * d_i(m, u, v) \quad (2)$$

where $d_i(m, u, v)$ is the distance to the closest edge and $w_i(m, u, v)$ is defined as 1 inside and as 0 outside the projected area of projector m . Finally, they use a gamma lookup table to correct for non-linear projector brightness curves. This method leads to the already mentioned artifacts in the corners of the blend masks that are discussed in more detail in the next section.

[YGH⁺01] extend this formulation to take different pixel densities into account. This is useful for the special case when high resolution insets are used.

More work for automatic calibration of tiled displays was done in the last decade. Majumder et al. achieve photometric uniformity by a luminance attenuation map [MS02], [MS03]. Furthermore, they use perceptual thresholds to not sacrifice too much of brightness/dynamic range. They observe, that non-Lambertian surfaces, e.g. viewed from the side may lead to brightness differences. Harville et al. present geometric and photometric calibration for a curved screen with special attention on its use in practise [HCS⁺06]. They also address the problem of computing a smooth blend mask in a very similar way to one of our methods. Their idea is to use a weighting function with a continuous first-order derivative inside the projected region. However, they overlook the discontinuity at the region border. Details are given in the next section.

A survey of camera based calibration techniques for photometric and geometric calibration is presented by Brown et al. in [BMY05]. Zhou et al. show continuously self calibrating projectors, using one rigidly attached camera for each projector [ZWAY08]. Grundhöfer et al. show real-time radiometric compensation [GB08]. A comprehensive overview of projector-camera systems is given by Bimber et al. in [BIWG08].

However, many of these approaches above rely on special hardware. As we rarely need to recalibrate our display wall, the effort of setting up a camera and implementing automatic calibration algorithms is not justified. Instead, we perform the geometric calibration manually, as described in [Ras00] and [LOU⁺06].

The HEyeWall uses a special hard edge blending, with the blends located close to the screen [KRK03]. However, color non-uniformity is more visible with such very short transitions.

The larger the overlap of the projectors, the less visible is the transition. But a larger overlap also results in less bright and lower resolution projection. To address this problem, a super resolution projection approach exists where all projector images are superimposed [DVC07]. This method needs very accurate information about the pixel position of each projector. Small mechanical misalignment requires a new calibration.

3 Corner Artifacts and Improved Algorithms

Implementing the soft edge blending approach by Raskar et al. [RBY⁺99], we observe artifacts in the corners of the blend masks (see Figure 1 on the left).

To address this problem we modify equation (2). The resulting intensities will still sum up to 1 by construction of the equation (1).

In practice, the blending area for our setup is considerably smaller than 50% of the image. Thus, it is sufficient to only consider the two closest edges e_1 and e_2 , with respective distances $d_1(m, u, v)$ and $d_2(m, u, v)$. Also, we do not consider edges that should not contribute to the blending, i.e. distances to edges on the outside of the screen.

We follow the notation of [RBY⁺99] and present the following variants, replacing the edge distance related term d_i of the weighting function:

$$d_i^1(m, u, v) = \min(d_1(m, u, v), d_2(m, u, v)) \quad (3)$$

$$d_i^2(m, u, v) = \left(\frac{1}{\frac{1}{d_1(m, u, v)} + \frac{1}{d_2(m, u, v)}} \right)^p \quad (4)$$

$$d_i^3(m, u, v) = d_1(m, u, v) * d_2(m, u, v) \quad (5)$$

$$d_i^4(m, u, v) = (d_1(m, u, v) * d_2(m, u, v))^p \quad (6)$$

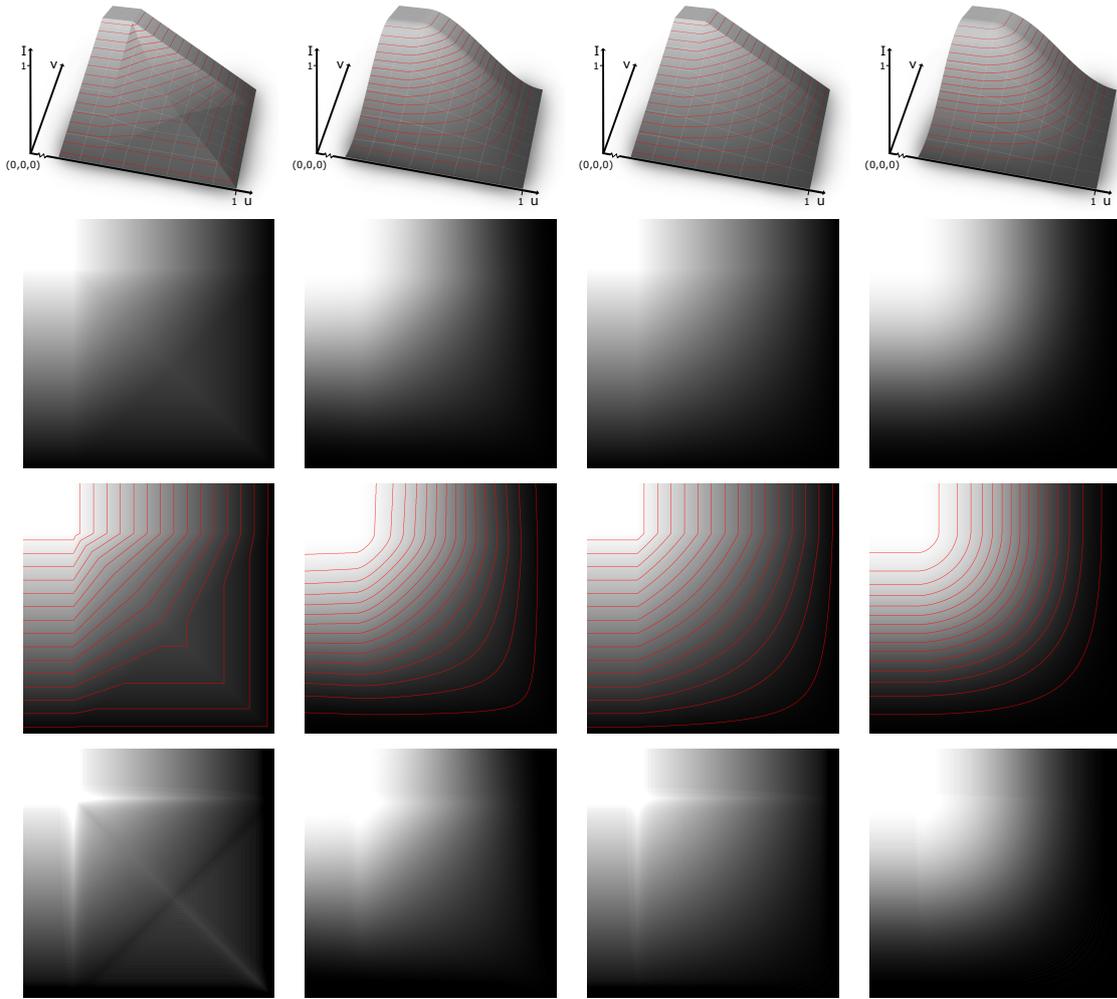


Figure 2: Comparison of different weighting functions. Only a crop of the bottom right corner of the mask of the top left projector is shown. The inner corners of other projectors are rotational symmetric. From left to right: d^1 by Raskar et al., d^2 with $p = 2.0$, d^3 by Harville et al., d^4 with $p = 1.5$. From top to bottom: 3D intensity graphs as in Figure 1, attenuation map coded in gray values, attenuation map with contour lines of equal brightness, attenuation map with an unsharp mask (sharpening) filter applied to highlight the type and amount of artifacts.

d^1 is the original function by Raskar et al. By using the minimum, there are two regions where only the smaller value is used while the other one is completely ignored. To avoid this reason for a visible artifact, we introduce a new function d^2 with a double fraction, always respecting both values. For a different, non-linear distribution of the weights, we also introduce an exponent p .

Later, we also tried to reproduce the attenuation pattern of physical blends in a corner with d^3 . The resulting function is the same as described in [HCS⁺06]. In order to simplify the code slightly, the authors change this to use the product of all four distances, i.e. to each edge. Again we apply the idea of adding an exponent p and get our new function d^4 . With

$p > 1$, the first order derivation of the weighting function also gets continuous at the image border.

Figure 2 shows a detailed comparison of the aforementioned weighting functions.

Another observation is, that when only adding the intensities of two neighboring projector images, when using d^1 , the corner is still visible (see Figure 3). Ideally the transition from one projector to the other should not be visible at all, as it is the case in d^2 , d^3 and d^4 . This case is relevant in practice e.g. when viewing a rear projection wall from the side, where the brightness varies with the projection angle.

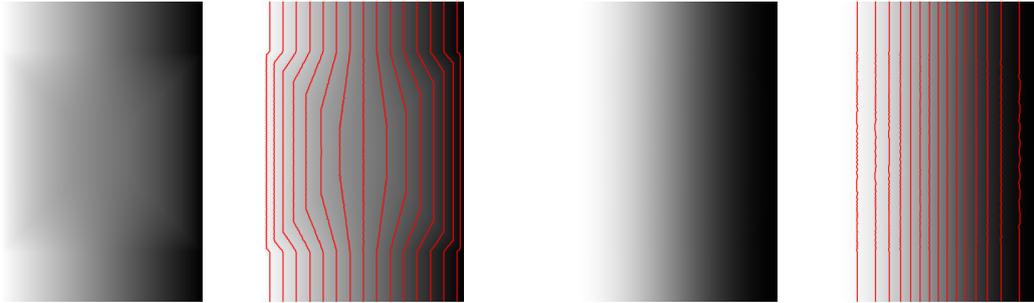


Figure 3: Comparison of the resulting light intensity when only using two neighboring projectors. From left to right: d^1 by Raskar et al. (left), the same showing iso-lines of equal intensity, d^4 with $p = 1.5$ and the same with iso-lines. This case is important e.g. for non-Lambertian projection surfaces like a rear projection screen, when viewed from an angle. On the left side, the corner is still visible.

Looking at the two bottom rows of Figure 2, both d^2 and d^4 lead to the least artifacts due to their first order continuity inside and at the borders with $p > 1$. This is well visible as the iso-intensity curves in the 3rd row are smooth, whereas the curves in the case of previous work (d^1 and d^3) suffer from sharp bends. As both of our new functions d^2 and d^4 seem to produce similar results, we further analyze their differences as well as the influence of p , shown in Figures 4 and 5. Increasing p leads to smoother iso-intensity curves but also narrows the transition width. However, a large transition width is desired. By experiment, we determine a good trade off for p for both functions. Finally, we choose d^4 with $p = 1.5$ because of its slightly visually smoother appearance at a similarly broad transition compared to d^2 .

4 Implementation Details

To compute the blend mask for each projector, the input is given as geometrical calibration file in an ASCII format with 16 float values for the 4×4 matrix. The photometric calibration file contains only a single gamma value for now. The same files are used by our frameworks for display. The mask generation additionally needs the following input parameters: output file name, native width and height of the projector, and a flag for each of the four edges if they should be considered for blending, i.e. whether the edge is on the inside of the screen.

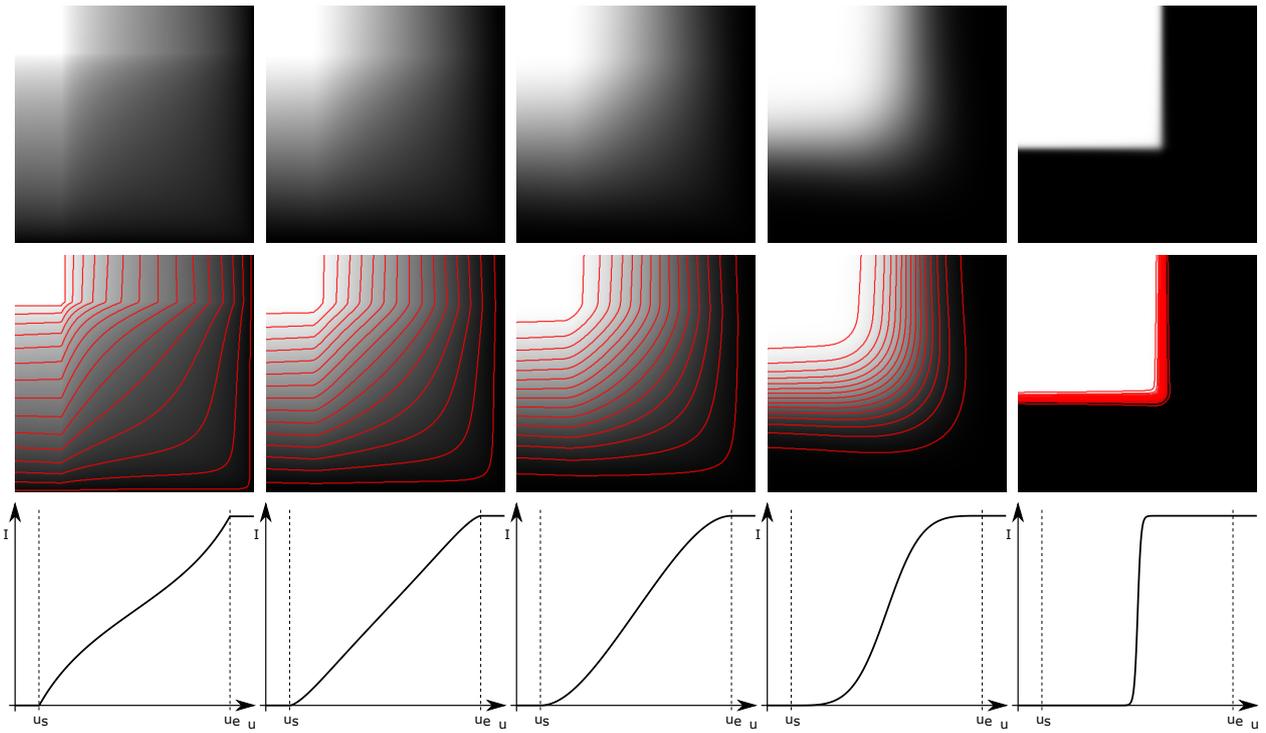


Figure 4: d^2 with varying exponents. From left to right: $p = 1.0$, $p = 1.5$, $p = 2.0$, $p = 4.0$, $p = 40.0$. The graph in the last row shows the edge transition between two projectors. By experiment, we found that $p = 2.0$ leads to good results (center column).

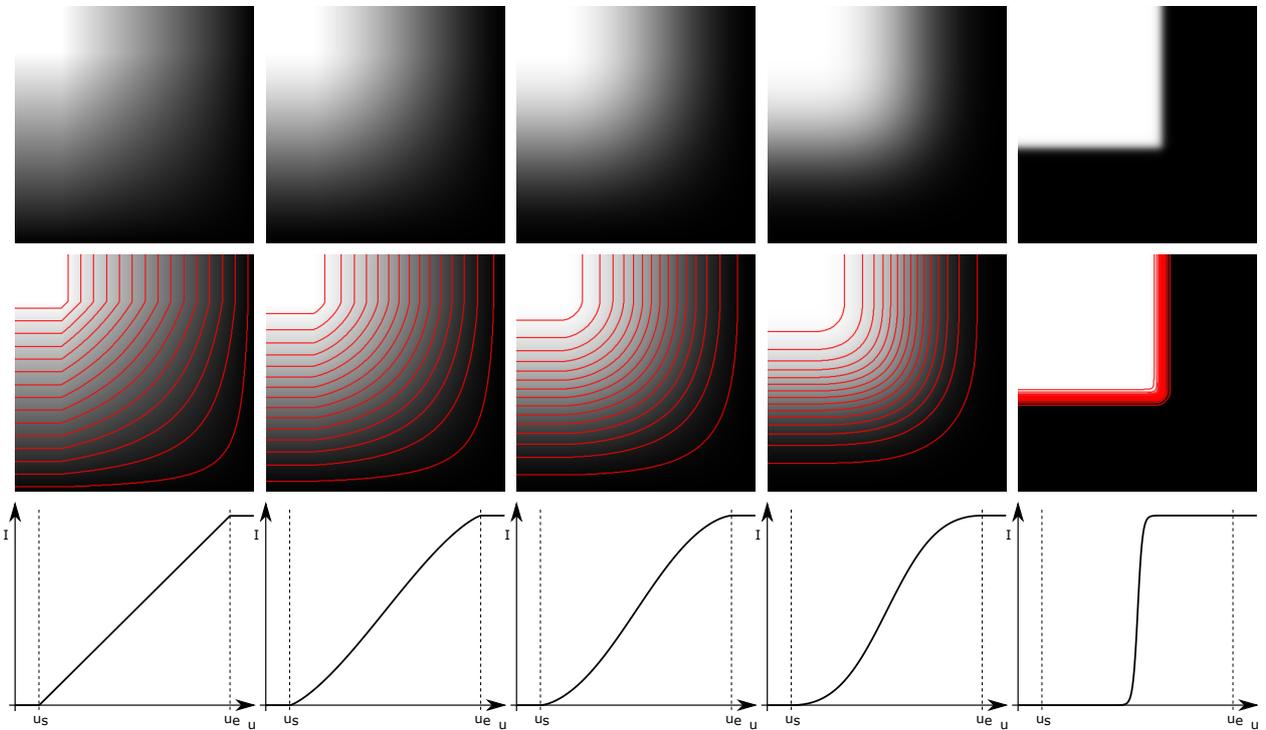


Figure 5: d^4 with varying exponents. From left to right: $p = 1.0$, $p = 1.25$, $p = 1.5$, $p = 2.0$, $p = 20.0$. By experiment, we found that $p = 1.5$ leads to good results (center column).

To get a pixel precise value, instead of forward warping the mask, we compute the inverse transformation for each mask pixel into the screen space.

To be compatible with all frameworks we use, we write out both just the mask as a pgm file as well as a png file with black rgb values and the respective alpha value.

5 Results

We have not yet implemented a photometric calibration by measuring the projector response curve. Instead, we use a manually chosen single gamma value for all projectors, which is a very rough approximation that we will address in future.

Figures 6 and 7 show a white image on our rear projection setup consisting of six DLP projectors with 1400×1050 pixels each, resulting in an effective image with about eight megapixels. Note the too dark corner due to the incorrectly approximated projector response curve. Also note that the white image is a worst case scenario, i.e. in a natural image with high frequencies, the transitions are much less visible.

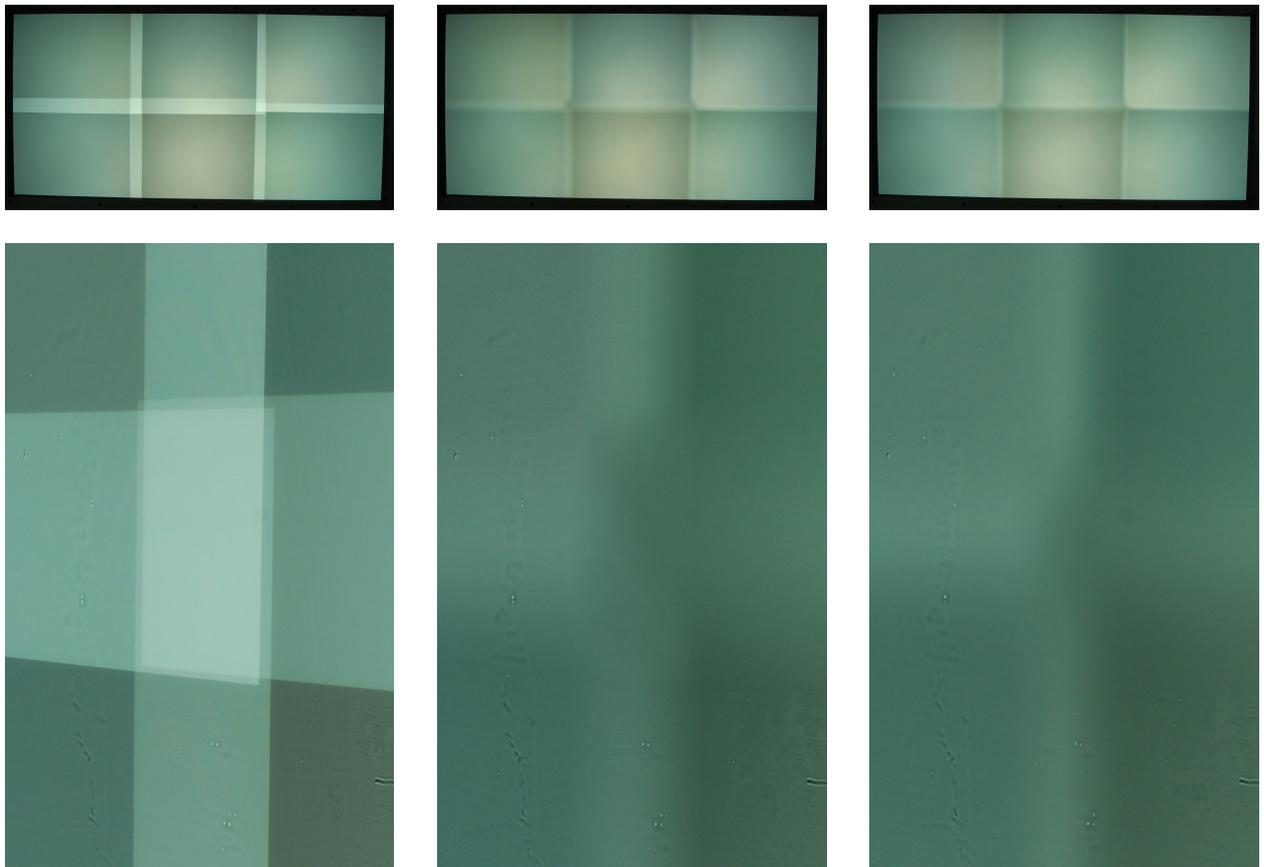


Figure 6: Comparison of no blending, d^1 by Raskar et al. and our d^4 with $p = 1.5$ for a tiled rear projection (from left to right). Here, the photometric calibration is very roughly approximated by just a single gamma value for all projectors. The second row shows a close up view with four overlapping images.



Figure 7: Another close up view of a different setup with less exact mechanically aligned projectors. No blending, d^1 by Raskar et al. and d^4 with $p = 1.5$ (from left to right).

6 Conclusion

While we get inferior results than sophisticated camera based methods for full photometric calibration, we avoid their complexity and effort with a simple and easy to implement approach. Even with fully calibrated projector arrays, when using non-Lambertian display surfaces, the calibration is only perfect for the camera point of view. Soft edge blending helps to hide the transitions.

Compared to the commonly used soft edge blending, we obtain blend masks with considerably smoother results at corners. Thus, in practice we achieve a slight reduction in perceived corner artifacts. Furthermore, only a small modification of the algorithm or program code is necessary. While only adding a small change to a similar solution suggested by Harville et al., we present in depth analyses and evaluation of the methods.

We thank the reviewers for the helpful comments.

References

- [BIWG08] Oliver Bimber, Daisuke Iwai, Gordon Wetzstein, and Anselm Grundhöfer. The visual computing of projector-camera systems. In *ACM SIGGRAPH 2008 classes*, SIGGRAPH '08, pages 84:1–84:25, New York, NY, USA, 2008. ACM.
- [BMY05] Michael Brown, Aditi Majumder, and Ruigang Yang. Camera-based calibration techniques for seamless multiprojector displays. *IEEE Transactions on Visualization and Computer Graphics*, 11:193–206, March 2005.
- [DVC07] N. Damera-Venkata and N.L. Chang. Realizing super-resolution with superimposed projection. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8, june 2007.

- [GB08] Anselm Grundhöfer and Oliver Bimber. Real-time adaptive radiometric compensation. *IEEE Transactions on Visualization and Computer Graphics*, 14:97–108, January 2008.
- [HCS⁺06] Michael Harville, Bruce Culbertson, Irwin Sobel, Dan Gelb, Andrew Fitzhugh, and Donald Tanguay. Practical methods for geometric and photometric correction of tiled projector. In *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop, CVPRW '06*, pages 5–, Washington, DC, USA, 2006. IEEE Computer Society.
- [KRK03] W. Kresse, D. Reiners, and C. Knöpfle. Color consistency for digital multi-projector stereo display systems: the heyewall and the digital cave. In *Proceedings of the workshop on Virtual environments 2003, EGVE '03*, pages 271–279, New York, NY, USA, 2003. ACM.
- [LOU⁺06] Marcel Lancelle, Lars Offen, Torsten Ullrich, Torsten Techmann, and Dieter W. Fellner. Minimally invasive projector calibration for 3d applications, 2006.
- [MS02] Aditi Majumder and Rick Stevens. Lam: luminance attenuation map for photometric uniformity in projection based displays. In *Proceedings of the ACM symposium on Virtual reality software and technology, VRST '02*, pages 147–154, New York, NY, USA, 2002. ACM.
- [MS03] Aditi Majumder and Rick Stevens. Color nonuniformity in projection-based displays: Analysis and solutions, 2003.
- [Ras00] Ramesh Raskar. Immersive planar display using roughly aligned projectors. In *Proceedings of the IEEE Virtual Reality 2000 Conference, VR '00*, pages 109–, Washington, DC, USA, 2000. IEEE Computer Society.
- [RBY⁺99] Ramesh Raskar, Michael S. Brown, Ruigang Yang, Wei-Chao Chen, Greg Welch, Herman Towles, Brent Seales, and Henry Fuchs. Multi-projector displays using camera-based registration. In *Proceedings of the conference on Visualization '99: celebrating ten years, VIS '99*, pages 161–168, Los Alamitos, CA, USA, 1999. IEEE Computer Society Press.
- [YGH⁺01] Ruigang Yang, David Gotz, Justin Hensley, Herman Towles, and Michael S. Brown. Pixelflex: a reconfigurable multi-projector display system. In *Proceedings of the conference on Visualization '01, VIS '01*, pages 167–174, Washington, DC, USA, 2001. IEEE Computer Society.
- [ZWAY08] Jin Zhou, Liang Wang, Amir Akbarzadeh, and Ruigang Yang. Multi-projector display with continuous self-calibration. In *Proceedings of the 5th ACM/IEEE International Workshop on Projector camera systems, PROCAMS '08*, pages 3:1–3:7, New York, NY, USA, 2008. ACM.