Optimized 3D Scene Rendering on Projection-Based 3D Displays

Oleksii Doronin
Unit of Computing Sciences
ITC Faculty
Tampere University
Tampere, Finland
oleksii.doronin@tuni.fi

Robert Bregovic
Unit of Computing Sciences
ITC Faculty
Tampere University
Tampere, Finland
robert.bregovic@tuni.fi

Atanas Gotchev
Unit of Computing Sciences
ITC Faculty
Tampere University
Tampere, Finland
atanas.gotchev@tuni.fi

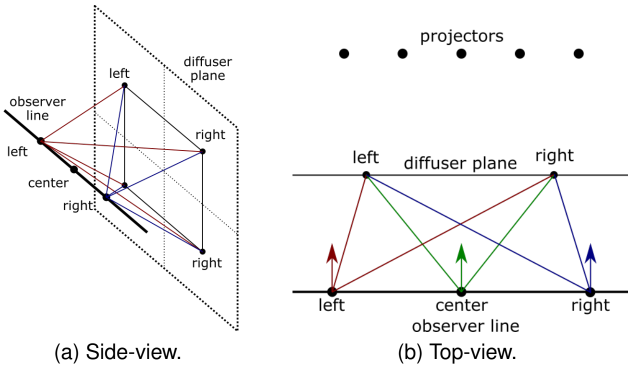
***Abstract*—We address the problem of 3D scene rendering on projection-based light field displays and optimizing the input display images to obtain the best possible visual output. We discuss a display model comprising a set of projectors, an anisotropic diffuser and a viewing manifold. Based on this model, we render an initial set of projector images to be further optimized for the best perception at a specified set of viewing positions. We propose a least squares method, which minimizes the channel-wise color difference between the generated images for different viewer positions, and their ground-true counterparts. We formulate a constrained optimization problem and solve it iteratively by the descent method.**

***Index Terms*—light field, optimization, 3d display.**

I. Introduction

Passive Light Field (LF) 3D display is a device that allows users to see a dynamically changed content relative to their current position in space, without the need of wearing special glasses or enabling eye tracking. Such displays maintain continuous parallax and to some extend generate focus visual cues. This is achieved mainly by generating a dense set of multi-perspective rays. In recent years, a variety of passive LF 3D displays has been introduced [1], [2]. There are two types of such displays which are of particular interest: first is *integral imaging* (InIm) display [3]; second is *projection-based* display [4], [5], [6]. This classification is not strict, since there are devices that utilize both technologies [7], or can be configured to represent either first or second class [8].

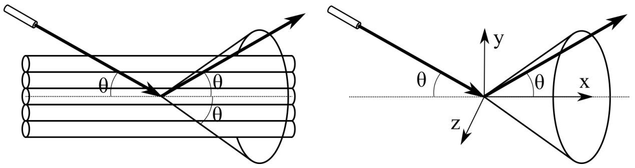
The basic configuration of projection-based display is described in [4]. It consists of a set of projectors placed along a horizontal row, and transparent anisotropic diffuser in front of them (see Figure 1). The role of the diffuser is to merge the output of all projectors with different weighting coefficients that depend on the viewing direction. Compared to an InIm setting, this configuration provides significantly wider field of view, better spatial-angular resolution, and scalability for a higher computational cost and data usage. An inter-projector cross-talk during the visualization might also occur. The basic display model described in [4] can be easily extended for other cases, including multi-row projector system [5], reflective diffuser surface, or 360 spinning mirror display [6].



(a) Side-view.

(b) Top-view.

Fig. 1. Display model.



(a) Microcylinder level.

(b) Scattering cone.

Fig. 2. Optical model of diffuser.

In this paper, we discuss the problem of optimal generation of the set of projector images for the case of single-row projection-based display model [4]. For this display model, only rasterization-based [4] and raytracing-based [9] rendering techniques have been previously introduced. Similar rendering approach has been utilized in [6] for the case of 360 spinning mirror display with reflective diffuser. Another problem of optimal projector arrangement has been discussed in [5] for a particular case of multi-row 300-projector system.

This paper is structured as follows. In Section II, we review the projection-based display model. In Section III, we formulate the optimization problem and solve it by using a least squares method. Then, we propose a cost-efficient iterative procedure which can be applied in real-life system. We summarize the results of our simulations in Section IV.

The research in this paper was done as a part of and was funded from the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant agreement No 676401, ETN-FPI.

II. Projection-based display model

In our model, we assume that all projectors are placed uniformly within an interval on a horizontal line (*projector line*) behind the screen surface. The screen surface is assumed to be flat and parallel to the projector line. In front of the screen surface, there is an imaginary *observer line* which corresponds to the expected viewer positions (see Figure 1).

A. Diffuser model

The screen surface is represented by an anisotropic diffuser, which scatters the incoming light differently in different directions. The exact diffusion model depends on a particular manufacturer, and it is often kept confidential. However, a viable typical diffuser model has been utilized in [6] for the case of 360 spinning mirror display with reflective diffuser. The model consists of a three-layers structure, similar to [10]: an upper layer corresponds to the local curvature of the screen surface visible by a naked eye, a middle layer represents the internal optical structure modeled by aligned glass microcylinders (see Figure 2), and a lower level accounts for the internal irregularities of a particular microcylinder. According to [11], the microcylinder layer scatters the light in a conical pattern, where the cone main axis corresponds to the microcylinder direction (a.k.a. diffuser *axis of anisotropy*), the cone vertex is the intersection point of light ray with the diffuser surface, and the cone radius is determined by the rule that the incoming light ray should lie on the cone surface.

We apply the aforementioned conical light scattering model to the case of transparent anisotropic diffuser. For horizontal-parallax display, the diffuser axis of anisotropy must be strictly horizontal. Consider a particular light ray intersecting the diffuser, and assume the local coordinate system with origin at the intersection point, and x, y, and z axes aligned with horizontal and vertical directions, and the direction towards the observer line correspondingly (see Figure 2b). In such parameterization, the scattering cone equation takes the form

$$x^2 = \rho^2 \cdot (y^2 + z^2), \tag{1}$$

where ρ is the inverse cone radius (can be negative). An incoming ray direction (x, y, z) can be uniquely identified by values ρ and η as follows:

$$\rho := x/\sqrt{y^2 + z^2}, \tag{2a}$$

$$\eta := y/z. \tag{2b}$$

Note that ρ and η can be understood as the tangent values of the horizontal and vertical scattering angles respectively.

For real-life diffusers, the incoming light ray with $\rho = \rho_0$ and $\eta = \eta_0$ is refracted in all possible directions. The refracted directions (ρ, η) will have the strongest light intensity at $(\rho, \eta) = (\rho_0, \eta_0)$. This intensity attenuates rapidly for the increase of the absolute value of $\Delta\rho := \rho - \rho_0$, and slowly for $\Delta\eta := \eta - \eta_0$. The attenuation model depends on the particular diffuser. Here, we model it by a Gaussian depending on the angular difference:

$$I(\Delta\rho, \Delta\eta) = \exp\Big[-\frac{\mathrm{atan}(\Delta\rho)^2}{\sigma_\rho^2} - \frac{\mathrm{atan}(\Delta\eta)^2}{\sigma_\eta^2} \Big], \tag{3}$$

where atan is the inverse tangent function, σ_ρ^2 and σ_η^2 are horizontal and vertical scattering powers respectively. It is naturally assumed that $\sigma_\rho^2 >> \sigma_\eta^2$.

B. Projector image

We adapt the raytracing-based method from [9] that allows us to render the set of projector images for a given synthetic scene. In this method, a particular pixel on the virtual image plane of a projector image corresponds to a unique light ray coming from the projector center through this pixel. Once this ray hits the screen, it is being scattered by the conical law described in Section II-A. Then, we need to find the intersection between the scattering cone and observer line. Finally, we spawn the camera ray from the found intersection towards the corresponding screen position.

C. Perceived image

Let us consider a point on the screen surface. This point takes the contribution c_i from each projector in the system, $i = 1, .., M$, where M is the total number of projectors. The color of this point varies depending on the viewer position X on the observer line. It can be calculated as the channel-wise total sum of all projector contributions:

$$v(X) := \sum_{i=1}^{M} w_i(X) \cdot c_i, \tag{4}$$

where c_i is the light intensity from i-th projector, $w_i(X)$ is the weighting coefficient calculated as $\alpha \cdot I(\Delta\rho, \Delta\eta)$, where $I(\Delta\rho, \Delta\eta)$ is defined in (3), and α is a normalization multiplier, defined pixel-wise as the reciprocal sum of $I(\Delta\rho, \Delta\eta)$ for all projectors. Equation (4) can be expressed in a vector-form:

$$v(X) := w(X)^\tau c, \tag{5}$$

where $c := (c_i) \in \mathbb{R}^M$ is the set of all same-channel projector image values for the given screen position, $w(X) := (w_i(X))$ is a vector-function from $\mathfrak{X}$ to $\mathbb{R}^M$, and $\mathfrak{X}$ is the set of all valid observer positions (*observer space*).

III. Optimization problem

Let us assume that we know the channel-wise ground-true value $v_{gt}(X)$ of the user-perceived image for all possible positions X from the observer space $\mathfrak{X}$. Then, we formalize the problem of finding the optimal set of projector contributions $c \in \mathbb{R}^M$ as a least squares minimization problem:

$$\mathcal{L}(c) := \int_{\mathfrak{X}} \big[w(X)^\tau c - v_{gt}(X) \big]^2 dX \to \min, \tag{6}$$

where the term $w(X)^\tau c$ represents the expected user-perceived value from position X, according to (5).

Under some generic mathematical assumptions, the Lagrangian $\mathcal{L}$ in the minimization problem (6) can be rewritten as

$$\mathcal{L}(c) = c^\tau Bc - 2\beta^\tau c + \bar{v}_{gt}^2 \to \min, \tag{7}$$

where B is M-by-M matrix, β is M-dimensional vector, and $\bar{v}_{gt}^2$ is a scalar defined as

$$B := \int_{\mathfrak{X}} w(X)w(X)^\tau dX, \tag{8a}$$

$$\beta := \int_{\mathfrak{X}} w(X)v_{gt}(X)dX, \tag{8b}$$

$$\bar{v}_{gt}^2 := \int_{\mathfrak{X}} v_{gt}(X)^2 dX. \tag{8c}$$

A. Unconstrained solution

Note that $\mathcal{L}(c)$ in (7) is in fact a quadratic form. Matrix B in this form is symmetric and positive semi-definite. Therefore, $\mathcal{L}(c)$ reaches the global minimum at some point c_*, for which $\mathcal{L}'_c(c_*) = 0$, or, equivalently, $Bc_* = \beta$. If the matrix B is non-singular, then there is a unique solution $c_* = B^{-1}\beta$. If B is singular, there are multiple solutions, and one of them can be expressed as $c_* = B^+\beta$, where B^+ is the Moore-Penrouse pseudoinverse matrix to B.

B. Constrained solution

In real life, the color values in projector images are limited within a certain range:

$$c_{\min} \le c_i \le c_{\max}, \quad i = 1, .., M, \tag{9}$$

and the problem (7) has to be solved under the constrains (9). The standard way to do it is to apply the Lagrange multipliers method with Karush-Kuhn-Tucker conditions. In case of constrains (9), this will increase the dimensionality of the linear equations system from M to $3M$.

Such approach is hardly suitable for practical needs, since the number of projectors M in real-life display systems may count up to few hundreds [4], [5]. This introduces several numerical errors while solving the system of $3M$ linear equations for each independent position on the screen. Therefore, we further introduce an iterative approach.

C. Iterative solution

We solve the problem (7) under constrains (9) iteratively:

$$c^{(t+1)} := c^{(t)} - \lambda^{(t)}\varphi^{(t)}, \tag{10}$$

where $c^{(t)} \in \mathbb{R}^M$ is the t-th iteration of the solution, $\varphi^{(t)} \in \mathbb{R}^M$ is the descent direction, and $\lambda^{(t)} \in \mathbb{R}$ is the step size.

Descent direction $\varphi^{(t)}$ is calculated as the element-wise solution of one-dimensional optimization problem:

$$\varphi_i^{(t)} := \begin{cases} 0 & \text{, if } \tilde{\varphi}_i^{(t)} > 0 \text{ and } c_i^{(t)} \le c_{min}, \\ 0 & \text{, if } \tilde{\varphi}_i^{(t)} < 0 \text{ and } c_i^{(t)} \ge c_{max}, \\ \tilde{\varphi}_i^{(t)} & \text{, otherwise,} \end{cases} \tag{11}$$

where $\tilde{\varphi}_i^{(t)}$ is the descent direction for the unconstrained problem:

$$\tilde{\varphi}_i^{(t)} := \begin{cases} \big([Bc^{(t)}]_i - \beta_i\big)/B_{ii} & \text{, if } B_{ii} > 0, \\ 0 & \text{, otherwise.} \end{cases} \tag{12}$$

The step size $\lambda^{(t)}$ is chosen as the solution of one-dimensional minimization problem $\mathcal{L}\big(c^{(t)} - \lambda\varphi^{(t)}\big) \to \min$ over $\lambda \in \mathbb{R}$:

$$\lambda^{(t)} := \frac{[\varphi^{(t)}]^\tau g^{(t)}}{[\varphi^{(t)}]^\tau B\varphi^{(t)}}, \tag{13}$$

where $g^{(t)}$ is the gradient value for the t-th iteration:

$$g^{(t)} := Bc^{(t)} - \beta. \tag{14}$$

We prove the mathematical correctness of our approach with the following statement.

Statement 1. *The following statements hold true:*
(i) $\tilde{\varphi}_i^{(t)}$ *in* (12) *gives solution for one-dimensional unconstrained optimization problem* $\mathcal{L}(c) \to \min$ *over* $c_i \in \mathbb{R}$.
(ii) $\lambda^{(t)}$ *in* (13) *gives solution of one-dimensional minimization problem* $\mathcal{L}\big(c^{(t)} - \lambda\varphi^{(t)}\big) \to \min$ *over* $\lambda \in \mathbb{R}$.
(iii) *Iterations* $c^{(t)}$ *in* (10) *converge to the solution of problem* (7) *under constrains* (9).

Proof. **(i)** The problem $\mathcal{L}(c) \to \min$ over $c_i \in \mathbb{R}$ has Lagrangian derivative $\mathcal{L}'_i(c) = 2([Bc]_i - \beta_i)$. From equation $\mathcal{L}'_i(c) = 0$ we get solution $c_i^* = \big(\beta_i - \sum_{j \neq i} B_{ij}c_j\big)/B_{ii}$, which is equivalent to $c_i^* = c_i - \lambda\varphi_i$ for $\lambda = 1$ and $\varphi_i = \big([Bc]_i - \beta_i\big)/B_{ii}$.

(ii) For an arbitrary $\varphi \in \mathbb{R}^M$, the problem $\mathcal{L}(c - \lambda\varphi) \to \min$ over $\lambda \in \mathbb{R}$ has Lagrangian derivative $L'_\lambda := -\varphi^\tau \mathcal{L}'(c - \lambda\varphi)$. Equation $L'_\lambda = 0$ can be expanded as $\varphi^\tau g - \lambda\varphi^\tau B\varphi = 0$, where $g := Bc - \beta$. Solution of this equation is obtained at $\lambda_* := (\varphi^\tau g)/(\varphi^\tau B\varphi)$.

(iii) Let us consider the unconstrained case. Proof for constrained case is similar, since (9) defines a convex subset of $\mathbb{R}^M$. First, the sequence $\mathcal{L}(c^{(t)})$ is monotone: $\mathcal{L}(c^{(t+1)}) \le \mathcal{L}(c^{(t)})$ for all $t \ge 0$. This follows from **(ii)**. Second, since $\mathcal{L}(c) \ge 0$ for all possible $c \in \mathbb{R}^M$, we conclude that there is a limit value of $\mathcal{L}(c^{(t)})$ for $t \to \infty$. Third, by the construction of $c^{(t)}$, we get $\mathcal{L}'(c^{(t)}) \to 0$ for $t \to \infty$. Since $\mathcal{L}(c)$ is a quadratic form with semi-positive matrix B, there is only one point of its extremum, which is obtained at c^* such that $\mathcal{L}'(c^*) = 0$. In case of non-singular B, such c^* is unique, and $c^{(t)} \to c^*$ for $t \to \infty$. □

IV. Results

We implemented the proposed approach in a virtual environment, following the display model as described in Section II. All numbers hereafter are given in world units. Screen size: 400-by-300, distance to projector line: 800, distance to observer line: 400. The number of projectors M is 41, and they are uniformly placed on the projector line interval $[-1000, 1000]$, with inter-projector distance of 50 units. The observer space $\mathfrak{X}$ is discrete, with 101 specified viewer positions in total, which are uniformly placed over the observer line interval $[-500, 500]$ with step 10. Each of them has the same weight in $\mathfrak{X}$ such that $\int_{\mathfrak{X}} 1dX = 1$.

We extended the PBRT-v3 renderer [12] in order to obtain all necessary source images. To get repeatable results, we used Whitted integrator and Halton sampler with 8 samples per pixel.

The test scene consists of three kangaroo models of red, green, and blue colors (see Figure 3). Each model is placed on different depth relative to the screen surface: the green one is at zero depth, the red one is at 200 units towards projectors, and the blue one is at 200 units towards the viewer. Images from the recentered pinhole cameras at discrete positions from $\mathfrak{X}$ are used as the ground-true ones (see "Ground-true" row in Figure 3). Initial projector images are obtained as described in Section II-B. After this, they were visualized by our simulation according to Section II-C (see "Iteration 0" row of Figure 3). Next, we applied the constrained iterative approach from Section III-C, and visualized the result as well (see rows "Iteration N", $N = 1, 10, 100$ in Figure 3).

Finally, we measured channel-wise MSE and SSIM metrics for all pairs of visualized vs ground-true images. The obtained values were then averaged for all 101 views. Table I shows the result for iterations 0, 1, 2, 3, 5, 10, 20, 50, 100.

TABLE I
Average values of MSE$\times 10^3$ (left) and SSIM$\times 10^2$ (right).

	Red	Green	Blue	Red	Green	Blue
Iter 0	3.77	4.56	6.16	92.42	92.02	92.10
Iter 1	3.08	3.72	5.13	92.75	92.38	92.50
Iter 2	2.86	3.46	4.80	92.95	92.58	92.70
Iter 3	2.76	3.36	4.66	93.03	92.67	92.79
Iter 5	2.66	3.25	4.51	93.13	92.77	92.89
Iter 10	2.55	3.12	4.35	93.25	92.91	93.02
Iter 20	2.47	3.03	4.24	93.36	93.01	93.12
Iter 50	2.41	2.96	4.17	93.45	93.11	93.19
Iter 100	2.39	2.94	4.15	93.49	93.14	93.22

Both Table I and Figure 3 show the gradual increase of perceived image quality during the iterative procedure. The MSE values of 100-th iteration are around 1.5 times lower than the values of the initial projector image set, which effectively means 50% improvement rate. Notably, the very first iteration already gives 20% of them. The SSIM values also show a monotone improving during the iterations. Thus we conclude that the proposed iterative procedure is able to yield up to 50% of visual improvement in terms of L_2-norm, if compared to the straightforward ray tracing method from [9].

If we compare visually the expected output from the iterative procedure (see Figure 3) to the ground-true images (the upper row) and a straightforward ray tracing ("Iteration 0"), we can conclude that the proposed method significantly reduces projector cross-talk. Such effect, however, may be less significant for the real-life devices, where projector placement and calibration is done more optimally. However, even our simplistic simulation shows that there is a room for improvement of the conventional projector image generation methods. For example, one can improve the ray tracing for projector images by modifying the initial ray distribution, which should take into account the influence of adjacent projectors. We leave this idea for the future work.

V. Conclusion

The proposed method significantly reduces the cross-talk effect on a projection-based 3D display. The experiments show fast convergence rate of the iterative procedure (results after iteration 10 and 100 are almost indistinguishable), even if the initial pixel value is not correct. Additionally, the proposed method can be used to optimize the set of projector images for the desired probability distribution of the viewer position on the observer line. It can be easily extended for the cases of non-planar screen, multi-row, 360-degrees, and other variations of projection-based 3D displays. The source code of our simulation is publicly available at GitHub repository [13].

References

[1] Masahiro Yamaguchi, "Light-field and holographic three-dimensional displays," *Journal of the Optical Society of America (JOSA)*, vol. 33, no. 12, pp. 2348–2364, 2016.

[2] Péter Tamás Kovács and Tibor Balogh, "3D visual experience," in *High-Quality Visual Experience*, pp. 391–410. Springer, 2010.

[3] Xiao Xiao, Bahram Javidi, Manuel Martinez-Corral, and Adrian Stern, "Advances in three-dimensional integral imaging: sensing, display, and applications," *Applied optics*, vol. 52, no. 4, pp. 546–560, 2013.

[4] Tibor Balogh, Péter T. Kovács, and Zoltán Megyesi, "HoloVizio 3D Display System," in *Proceedings of the First International Conference on Immersive Telecommunications*, ICST, Brussels, Belgium, Belgium, 2007, ImmersCom '07, pp. 19:1–19:5, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[5] Jin-Ho Lee, Juyong Park, Dongkyung Nam, Seo Young Choi, Du-Sik Park, and Chang Yeong Kim, "Optimal projector configuration design for 300-Mpixel multi-projection 3D display," *Optics express*, vol. 21, no. 22, pp. 26820–26835, 2013.

[6] Andrew Jones, Ian McDowall, Hideshi Yamada, Mark Bolas, and Paul Debevec, "Rendering for an interactive 360 light field display," *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3, pp. 40, 2007.

[7] Yasuhiro Takaki and Nichiyo Nago, "Multi-projection of lenticular displays to construct a 256-view super multi-view display," *Optics express*, vol. 18, no. 9, pp. 8824–8835, 2010.

[8] Jong-Young Hong, Soon-Gi Park, Chang-Kun Lee, Seokil Moon, Sun-Je Kim, Jisoo Hong, Youngmin Kim, and Byoungho Lee, "See-through multi-projection three-dimensional display using transparent anisotropic diffuser," *Optics express*, vol. 24, no. 13, pp. 14138–14151, 2016.

[9] Oleksii Doronin, Attila Barsi, Peter A. Kara, and Maria G. Martini, "Ray Tracing for HoloVizio Light Field Displays," in *3D Immersion (IC3D) 2017*. 2017, pp. 1–8, IEEE.

[10] Pierre Poulin and Alain Fournier, "A model for anisotropic reflection," in *ACM SIGGRAPH Computer Graphics*. ACM, 1990, vol. 24, pp. 273–282.

[11] James T Kajiya and Timothy L Kay, "Rendering fur with three dimensional textures," in *ACM Siggraph Computer Graphics*. ACM, 1989, vol. 23, pp. 271–280.

[12] Matt Pharr, Wenzel Jakob, and Greg Humphreys, *Physically based rendering: From theory to implementation*, Morgan Kaufmann, 2016.

[13] "LFDisplay project," https://github.com/LeksiDor/LFDisplay, (retrieved July 2020).

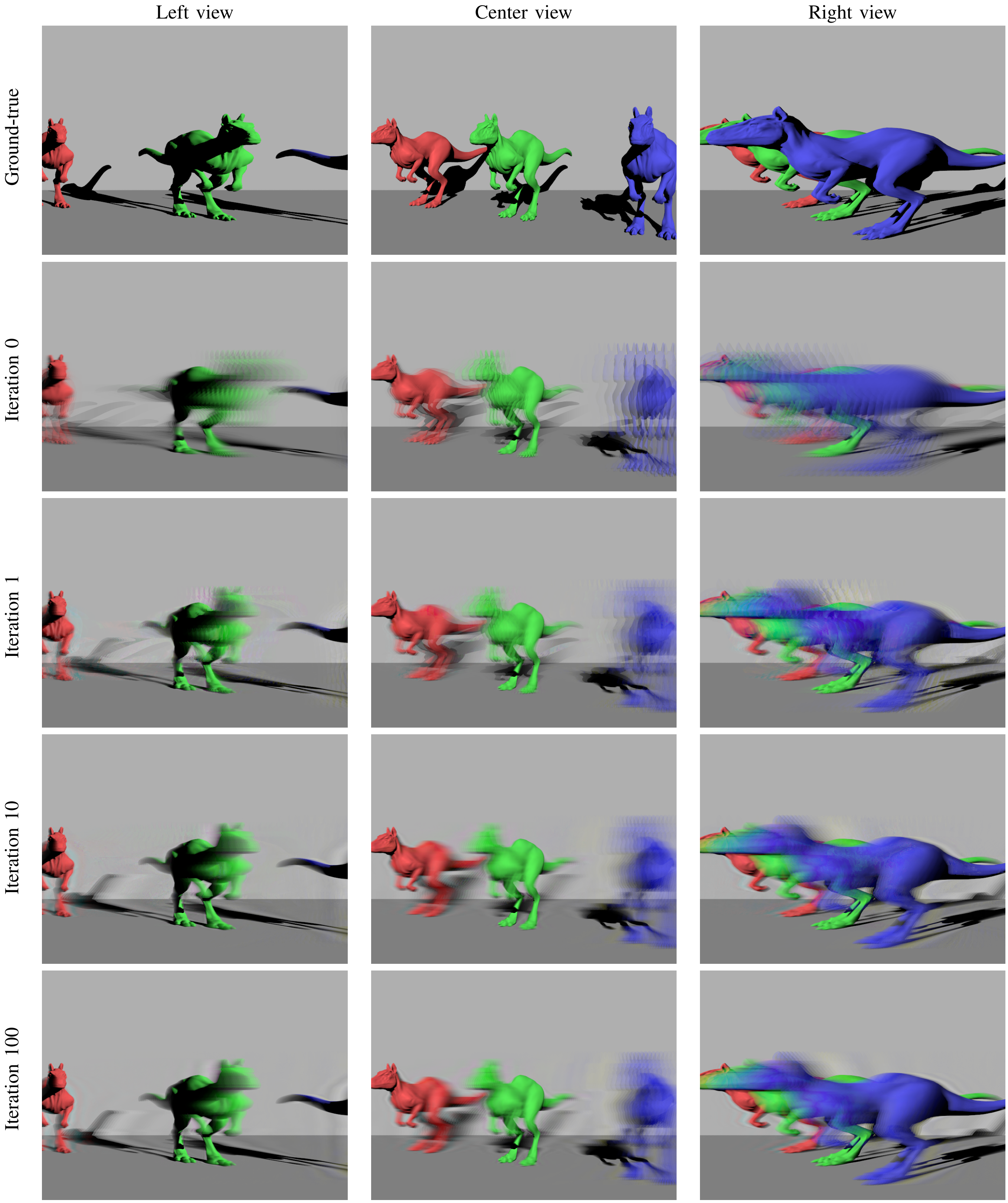


Fig. 3. Rendered images at three viewing positions.