Compression and Denoising of Time-Resolved Light Transport

YUN LIANG[1], MINGQIN CHEN[1], ZESHENG HUANG[1], DIEGO GUTIERREZ[2], ADOLFO MUÑOZ[2], AND JULIO MARCO[2,*]

[1] South China Agricultural University
[2] Universidad de Zaragoza, I3A
[*] Corresponding author: juliom@unizar.es

Exploiting temporal information of light propagation captured at ultra-fast frame rates has enabled applications such as reconstruction of complex hidden geometry, or vision through scattering media. However, these applications require high-dimensional and high-resolution transport data which introduces significant performance and storage constraints. Additionally, due to different sources of noise in both captured and synthesized data, the signal becomes significantly degraded over time, compromising the quality of the results. In this work we tackle these issues by proposing a method that extracts meaningful sets of features to accurately represent time-resolved light transport data. Our method reduces the size of time-resolved transport data up to a factor of 32, while significantly mitigating variance in both temporal and spatial dimensions.

http://dx.doi.org/10.1364/ao.XX.XXXXXX

Transient imaging methods [1] typically exploit time-resolved data in the order of nano- [2] to femto-seconds [3], involving spatio-temporal data structures to represent light propagation. Related applications such as reconstruction of hidden geometry [4, 5] require exhaustive scans of the scene at multiple camera and light locations, resulting in 5-dimensional data. Monte Carlo methods for transient rendering [6, 7] allow to accurately simulate time-resolved light transport. As such, they have become a helpful instrument for analysis, benchmarking, or as a data source for machine learning approaches [8, 9]. This increased dimensionality and high temporal resolution yield massive discretized representations of light transport that hamper the efficiency on practical applications. While methods to increase computational performance exist [10], memory and bandwidth are still limiting constraints. Moreover, this sort of time-resolved signals are degraded either by the attenuation of captured light, or due to variance in Monte Carlo simulations. Therefore noise removal and reconstruction algorithms become key to develop robust imaging methods. Feature extraction and representation in alternative domains have been extensively used for reconstruction and compression of different types of signals. There exist a wide variety of encoding and fast decoding methods for low-dynamic-range image and video data, where exploiting frequency characteristics predominates in most widespread compression algorithms [11]. Closer to our domain of application, representing time-resolved light transport by a combination of Gaussians and exponential functions has been proved useful for applications such as illumination decomposition [12] or imaging in scattering media [13].

However, while compression and denoising methods have been extensively researched for steady-state images and video, time-resolved light transport has distinctive properties that we exploit in this paper. First, light propagation is heavily structured in both time and space: the magnitude and frequency of the signal decrease over time due to multiple convolutions and attenuations of scattered light (see Figure 1, right); moreover temporal propagation is strongly correlated to spatial features of the scene, since light time-of-flight depends in part on the optical paths through the scene. Second, due to temporal delays in light propagation, similar temporal patterns can occur at different times. In Figure 1 (blue, red, yellow) we can see how the temporal delay of the initial peak is directly proportional to the depth at different points of the scene. Finally, time-resolved transport is particularly prone to noise, either due to signal attenuation in captured data, or to slow convergence rates in simulation (see Figure 1, right). These characteristics pose several challenges when finding alternative representations of time-resolved light transport. We take into account all these aspects to design a method for compressing and recovering transient light transport data based on encoder-decoder neural networks. We leverage existing databases [8] to learn sets of spatio-temporal features, and build lightweight representations of discretized time-resolved transport up to 32 times smaller than the original signal. This work is a formalization and continuation of our preliminary results [14].

Let $L_{\vec{\omega}}(t), t \in [0, \infty)$ be a function that represents time-resolved radiance in a scene from a viewing direction $\vec{\omega}$. While $L_{\vec{\omega}}(t)$ is continuous, this function does not have closed-form solutions for general scenes. As a consequence, in practice $L_{\vec{\omega}}(t)$ is represented by a discrete set of radiance values $L_{ij}[0, 1, ..., T-1] \in \mathbb{R}^T$—either measured or computed—uniformly-distributed over time. Each $L_{ij}[k]$ represents the integrated radiance over a time interval Δt centered at a time t_k, at pixel $\mathcal{H}[i, j, k]$ of a transient image $\mathcal{H}_{M \times N \times T}$ (Figure 1, middle). For simplicity we will use $L_{ij}(t)$ to refer to these discretized

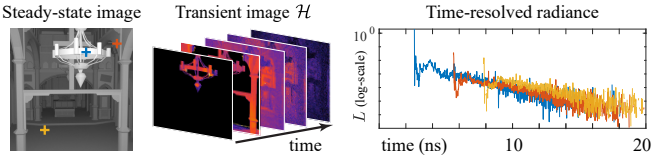

Fig. 1. Left: Simulated steady-state render of the *Altar* scene. Middle: Transient image of the scene. Right: Time-resolved radiance at marked points of the scene. Transient light transport is often characterized by arbitrary propagation delays, exponential decay, and temporal frequency that decreases over time.

radiance profiles at positions $\{i, j\}$ of a transient image $\mathcal{H}$.

In order to obtain accurate but small representations of time-resolved pixels $L_{ij}(t) \in \mathbb{R}^T$, we analyze and exploit the aforementioned properties of transient light transport to introduce a compression and denoising method. Recent works [8, 15, 16] explicitly described the strong spatio-temporal correlation and convolutional nature of light transport. Inspired by this, we propose to use convolutional encoder-decoders to learn two mappings. First, learning an encoding function $\mathbf{E}(\cdot)$ to extract a set of features f_L from some discretized input data X,

$$\mathbf{E}(X') = f_L, \quad \text{where} \quad X' = g(X). \tag{1}$$

The function $g(\cdot)$ represents a transformation function applied to the input X. Second, learning a decoding function $\mathbf{D}(\cdot)$ such that

$$\mathbf{D}(f_L) = Y', \quad \text{where} \quad g^{-1}(Y') = \hat{L}_{ij}(t), \tag{2}$$

which estimates the target time-resolved radiance $L_{ij}(t) \approx \hat{L}_{ij}(t)$ based on the feature vector f_L.

The resulting f_L of the encoding function will be the compressed representation of the signal $L_{ij}(t)$. The choice of X is key to ensure that the encoding function $\mathbf{E}$ has enough information to obtain a feature vector f_L representative enough for the decoder $\mathbf{D}$ to accurately estimate $L_{ij}(t)$. Functions g, $\mathbf{E}$, and $\mathbf{D}$ must account for the aforementioned challenges of time-resolved radiance: 1) it decays exponentially and its frequency is reduced over time; 2) it can have arbitrary propagation delays; 3) it can suffer from signal noise. Finally, since the data can have arbitrary temporal resolution, it is desirable to handle temporal profiles of arbitrary length with the same compression ratio. We thus introduce several design choices on the input data X, the transformation function g, and the encoder and decoder operations $\mathbf{E}$, $\mathbf{D}$.

Input data To leverage the local spatio-temporal coherence of light transport, we propose to use a time-resolved spatial neighborhood $X \equiv \langle L_{ij} \rangle$ centered at L_{ij} as input for the feature extraction step (Equation 1). Time-resolved signal has a high dynamic range with exponential decay over time due to recursive light bounces. To prevent the encoding step from ignoring low-valued radiance features, we define a logarithmic transformation g over the input data as

$$g(X) = \begin{cases} \log_{10}(X) - \log_{10}(\varepsilon) & X \geq \varepsilon \\ 0 & X < \varepsilon \end{cases}. \tag{3}$$

The threshold ε and offset $\log_{10}(\varepsilon)$ ensure all resulting values are above zero, and prevent input values close to zero going to infinity. In our experiments, not applying a logarithmic transformation made our optimization fall into local minima with zero-valued outputs for any input. We set a threshold of $\varepsilon = 1\text{e}{-7}$ based on radiance values distributions of our training and validation datasets. In practice, we found that a neighborhood of size 9×9 allows to find enough spatio-temporal features, while significantly mitigating noise in the recovered signal.

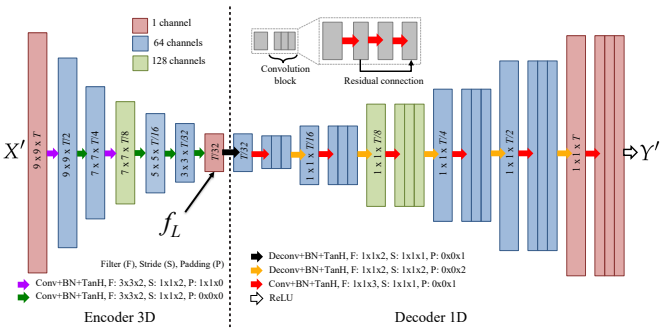

Fig. 2. Our proposed architecture. The encoder extracts a total of $T/32$ features f_L from a $9 \times 9 \times T$ spatial neighborhood in logarithmic space $X' = g(\langle L_{ij} \rangle)$, centered at the time-resolved pixel L_{ij} to compress. The decoding step uses these features to recover the time-resolved pixel $\hat{L}_{ij} = g^{-1}(Y')$ with a set of deconvolutions and residual convolution blocks.

Encoding step To extract a set of representative features from the spatial neighborhood $\langle L_{ij} \rangle$, we design a fully-convolutional learnable encoding function $\mathbf{E}$ (Equation 1). The function is composed of 3D convolutional filters (see Figure 2, left) that operate over both spatial and temporal dimensions. These filters exploit spatio-temporal structures of light transport, while simultaneously discarding noise in the signal. The fully-convolutional nature of this function allows us to keep a constant compression ratios over arbitrary temporal resolutions. To enable this, the filters simultaneously perform the following operations: a) progressively reduce the size of the spatial dimensions to 1×1 in the innermost layer (i.e. the compressed signal) by controlling the padding over the fixed-size spatial neighborhood $\langle L_{ij} \rangle$; b) sequentially apply strides of size 2 in the temporal dimension. Each layer of this function works similarly to a downsampling operation. However since the filters are optimized based on a minimized loss between the estimated and the reference signals, the encoding learns to extract the most representative features. Each element of the resulting vector f_L encodes features from a bounded time interval of the input $\langle L_{ij} \rangle$ (see Figure 3, left). Note that while our encoding function is computationally expensive due to 3D convolution operations, it needs to be run only once per each time-resolved pixel when compressing our signal. We design this function with five convolutional layers that generate a feature vector f_L 32 times smaller than the original signal $L_{ij}(t)$ to be compressed. This compression ratio can be varied by retraining with different number of convolutional layers, but in practice we found that this number provides a good trade-off between size reduction, denoising, and preservation of features.

Decoding step Given a set of features f_L, we aim to learn a decoding function $\mathbf{D}$ (Equation 2) that estimates the target uncompressed signal L_{ij}. Note that we do not want to estimate the whole input $\langle L_{ij} \rangle$, but just the central time-resolved pixel L_{ij}. We design the function $\mathbf{D}$ to perform a set of 1D temporal deconvolutions and convolutions that operate over the features

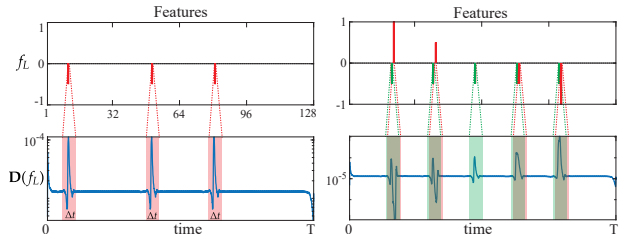

Fig. 3. Our encoder output features map to bounded time intervals of the decoder output, resulting in equally-shaped radiance patterns at arbitrary times (bottom-left). Our decoding function combines these features over the whole temporal domain in an overlapped manner (bottom-right), where a time instant is affected by multiple features.

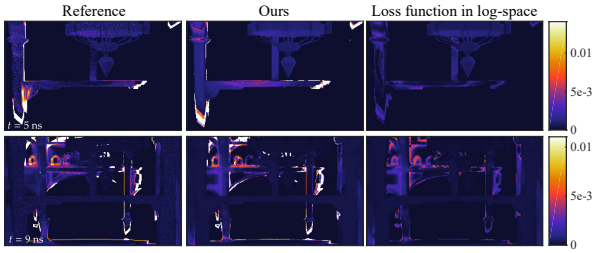

Fig. 4. Results of the *Altar* scene (see Visualization 1), with reference frames (left). Training with our exponential transform MSE loss (center, Equation 4) is able to recover strong direct peaks, while a simple MSE loss applied directly over the logarithmic-space of the decoder output (right) fails to recover these features.

f_L extracted by the encoding step (Equation 1). This step works as an upsampling operation with learnable 1D filters. Following previous works on deep residual nets [17], we apply residual connections between deconvolution blocks (see Figure 2). The key aspect of our decoding function is that, by construction, it learns a non-linear mapping between every feature and a corresponding time interval Δt over the recovered signal. This ensures that our method can handle arbitrary propagation delays that yield similar radiance patterns placed over the temporal dimension. In Figure 3, left, we illustrate this by changing the value of a single feature at different positions of f_L, resulting in equivalent temporal profiles over the corresponding time intervals. More importantly, the convolutional blocks in our decoder (see Figure 2, right) ensure each time instant t is covered by multiple features, and therefore its radiance value $L(t)$ is the sum of multiple non-linear mappings of the features that cover that time instant, allowing for increased complexity in the recovered signal. This is illustrated in Figure 3, right, where adjacent features map to overlapping time intervals in the decoded radiance.

Training and loss function As in classic encoding-decoding architectures, we perform simultaneous training of $\mathbf{E}$ and $\mathbf{D}$ parameters. We optimize these by minimizing an error function $\mathcal{L}$ between the reference L_{ij} and the decompressed time-resolved radiance $\hat{L}_{ij}$. Since our encoding function operates over a logarithmic transformation of radiance (Equation 3), the features f_L handled to the decoder $\mathbf{D}$ and in consequence the resulting output $Y' = \mathbf{D}(f_L)$ (Equation 2) are also in logarithmic space of radiance. To keep a good trade-off between estimating peak direct illumination and indirect illumination, we apply an exponential transformation over both the decoding output $\mathbf{D}(f_L)$ and the log-space central pixel $g(L_{ij}(t))$, and minimize the mean squared error over these, having

$$\mathcal{L} = \frac{1}{T}\sum_{t=0}^{T-1}\left(b^{g(L_{ij}(t))} - b^{\mathbf{D}(f_L)(t)}\right)^2, \qquad (4)$$

where b is the base of the exponential function. In practice, we found that choosing $b = 2$ provides good results for successfully decompressing both direct illumination peaks and smooth indirect bounces (see Figure 4).

Dataset For training and validation, we rely on the publicly available *Zaragoza-DeepToF* transient dataset [8], which contains a sufficiently large number of complex scenarios to prevent overfitting in our approach. It contains 1050 time-resolved simulations for a wide variety of architectural scenarios, with a spatial resolution of 300 × 300 and a temporal resolution of 4096 pixels at 16.6 picoseconds/pixel. For training, we randomly select a total of 860 000 pixel neighborhoods of size 9×9 from 145 scenes. For validation, we select a total of 370 000 inputs from 37 completely different scenes. While global illumination introduces correlation between patches, our validation set is uncorrelated with the training set, since the patches come from different scenarios. Our training is unsupervised, where our target L_{ij} is the central pixel of the input neighborhood $\langle L_{ij} \rangle$. Although the simulations in the dataset are not completely noise-free, our method based on 3D convolutions is capable of extracting spatio-temporal features while simultaneously removing high-frequency variance from noisy data.

Scene	HDF5	OpenEXR	RGBE	**Ours**
Altar (Fig. 4)	9.5 %	8.8 %	8.9 %	3.1 %
Balcony (Fig. 5, left)	12.7 %	9.8 %	13.4 %	
Building (Fig. 5, right)	17.1 %	12.9 %	16.6 %	
Room (Fig. 6)	28.4 %	20.1 %	24.9 %	

Table 1. Reduction ratios for the validation scenes illustrated in this article for standard libraries supporting HDR compression.

Figure 6 shows reference frames of the *Room* scene from the validation set (top row), and the resulting frames after compressing each reference time-resolved pixel to 128 features and decompressing them back to 4096 pixels (second row). Bottom row shows the full time-resolved signal at the marked location, with the reference (blue) and our recovered radiance (green), and the timestamps of the frames. Our trained decoder successfully recovers most radiance features of the scene using a compressed representation of the radiance 32 times smaller than the original. Table 1 compares compression ratios for three standard HDR compression libraries—RGBE, OpenEXR using wavelet/Huffman compression, and HDF5 with gzip— for all the validation scenes shown in this article, showing that our method yields smaller representations (3.1% of the original signal) than other approaches (8.8% to 28.4%). Please refer to Visualization 1 for the entire frame sequences. One of the pathological problems in transient light transport data is the presence of different types of noise in the signal. In particular, Monte-Carlo-based transient rendering methods suffer from high variance due to uneven distributions of samples over time [6]. Our fully-convolutional encoder is capable of extracting the most significant features by performing 3D spatio-temporal convolutions. In Figure 5 we can observe the results of the denoising in two extreme cases with higher-order indirect illumination in the *Building* and *Balcony* validation scenes. Our approach does not force the compressed features (shown in red) to retain light transport properties. However, while the samples at the tar-

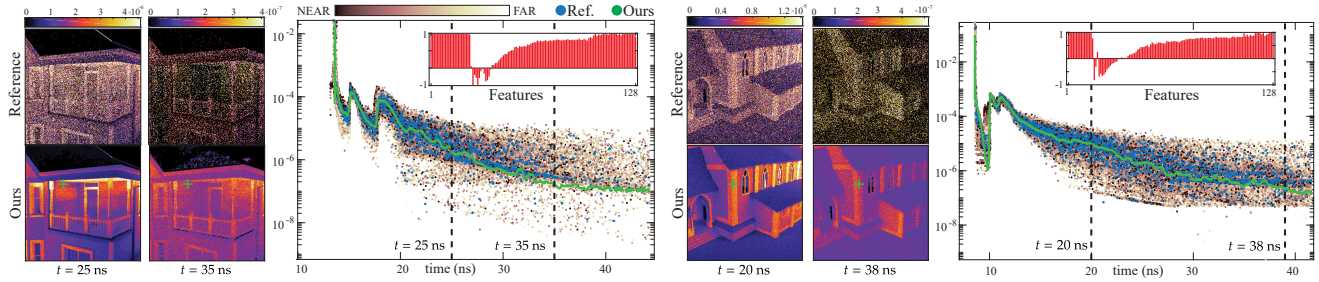

Fig. 5. Higher-order indirect illumination results in the *Balcony* (left) and *Building* (right) scenes from the validation set. Images show selected reference frames (top) and our denoised frames (bottom) after encoding and decoding each time-resolved pixel. Plots show our time-resolved profiles at marked pixels (green), reference samples of that pixel (blue), compressed features (red), and all the spatio-temporal input samples analyzed by our encoder, color-coded by the distance to the center of the neighborhood $\langle L_{ij} \rangle$. See Visualization 1 for entire video.

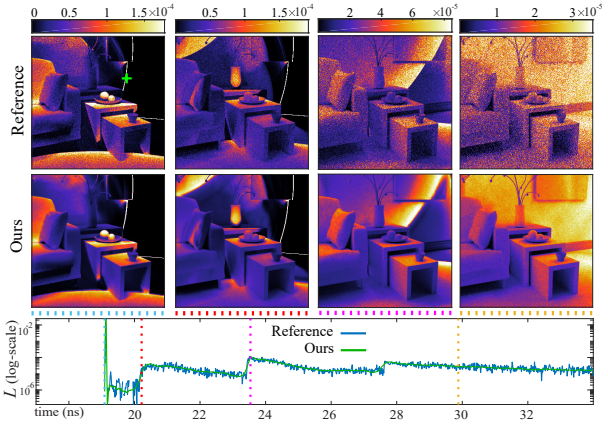

Fig. 6. *Room* scene (see Visualization 1) after encoding and decoding steps, showing high decompression accuracy, recovering both high- and low-frequency features in the temporal domain. Top: Reference input frames. Center: Our resulting frames after decompressing all time-resolved pixels. Bottom: Time-resolved transport at the marked location in the top-left frame.

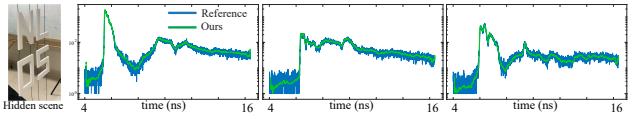

Fig. 7. Results for real data (blue) captured on a non-line-of-sight setup (left) [5]. The plots show our results (green) at different points of the captured grid.

get time-resolved pixel L_{ij} (blue) present a lot of variance, the spatio-temporal neighboring samples (brown color scale) contain relevant information which our encoder uses to extract the most significant features to decode our reconstructed signal. Finally, Figure 7 shows how our method generalizes to real data captured in non-line-of-sight configurations (e.g. [4, 5]), where the temporal profiles present much smoother features due to direct illumination being scattered by an auxiliary capture wall.

In conclusion, we have presented a new method for compressing and denoising transient light transport data. By observing the characteristics of light transport in the temporal domain, we have demonstrated how spatio-temporal 3D convolutions are capable of extracting most meaningful features even in extremely noisy conditions. This leads to a compressed signal, from which the original can be recovered with significantly less variance by means of a convolutional decoder. Transient imaging methods and hardware present critical trade-offs between capture time and signal noise. Our method can mitigate this, while reducing the computational time required to post-process the data. We believe that our pipeline can be further applied to large captured datasets, once acquisition processes become faster.

Funding European Research Council (ref. 682080). DARPA (HR0011-16-C-0025). National Natural Science Fund of China (No:61772209). Graduate Student Overseas Study Program from South China Agricultural University (No: 2019GWFX028). The Science and Technology Planning Project of Guangdong Province (No:2019A050510034, 2019B020219001).

Disclosures The authors declare no conflicts of interest.

REFERENCES

1. A. Jarabo, B. Masia, J. Marco, and D. Gutierrez, Vis. Informatics **1** (2017).
2. F. Heide, M. Hullin, J. Gregson, and W. Heidrich, ACM Trans. Graph. **32** (2013).
3. A. Velten, D. Wu, A. Jarabo, B. Masia, C. Barsi, C. Joshi, E. Lawson, M. Bawendi, D. Gutierrez, and R. Raskar, ACM Trans. Graph. **32** (2013).
4. A. Velten, T. Willwacher, O. Gupta, A. Veeraraghavan, M. G. Bawendi, and R. Raskar, Nat. Commun. (2012).
5. X. Liu, I. Guillén, M. La Manna, J. H. Nam, S. A. Reza, T. H. Le, A. Jarabo, D. Gutierrez, and A. Velten, Nature. (2019).
6. A. Jarabo, J. Marco, A. Muñoz, R. Buisan, W. Jarosz, and D. Gutierrez, ACM Trans. Graph. **33** (2014).
7. J. Marco, I. Guillén, W. Jarosz, D. Gutierrez, and A. Jarabo, Comput. Graph. Forum **38** (2019).
8. J. Marco, Q. Hernandez, A. Muñoz, Y. Dong, A. Jarabo, M. Kim, X. Tong, and D. Gutierrez, ACM Trans. on Graph. **36** (2017).
9. Q. Guo, I. Frosio, O. Gallo, T. Zickler, and J. Kautz, "Tackling 3d tof artifacts through learning and the flat dataset," in *The European Conference on Computer Vision (ECCV),* (2018).
10. V. Arellano, D. Gutierrez, and A. Jarabo, Opt. express **25**, 11574 (2017).
11. G. K. Wallace, IEEE Trans. on consumer electronics **38**, xviii (1992).
12. D. Wu, A. Velten, M. O'Toole, B. Masia, A. Agrawal, Q. Dai, and R. Raskar, Int. J. Comput. Vis. **107** (2014).
13. F. Heide, L. Xiao, A. Kolb, M. B. Hullin, and W. Heidrich, Opt. Express **22** (2014).
14. Y. Liang, M. Chen, Z. Huang, D. Gutierrez, A. Muñoz, and J. Marco, "A data-driven compression method for transient rendering," in *ACM SIGGRAPH 2019 Posters,* (ACM, 2019), p. 33.
15. M. O'Toole, F. Heide, L. Xiao, M. B. Hullin, W. Heidrich, and K. N. Kutulakos, ACM Trans. Graph. **33** (2014).
16. R. Ng, R. Ramamoorthi, and P. Hanrahan, ACM Trans. Graph. **22** (2003).
17. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition,* (2016), pp. 770–778.

FULL REFERENCES

1. A. Jarabo, B. Masia, J. Marco, and D. Gutierrez, "Recent advances in transient imaging: A computer graphics and vision perspective," Vis. Informatics **1** (2017).
2. F. Heide, M. Hullin, J. Gregson, and W. Heidrich, "Low-budget transient imaging using photonic mixer devices," ACM Trans. Graph. **32** (2013).
3. A. Velten, D. Wu, A. Jarabo, B. Masia, C. Barsi, C. Joshi, E. Lawson, M. Bawendi, D. Gutierrez, and R. Raskar, "Femto-photography: Capturing and visualizing the propagation of light," ACM Trans. Graph. **32** (2013).
4. A. Velten, T. Willwacher, O. Gupta, A. Veeraraghavan, M. G. Bawendi, and R. Raskar, "Recovering three-dimensional shape around a corner using ultrafast time-of-flight imaging," Nat. Commun. (2012).
5. X. Liu, I. Guillén, M. La Manna, J. H. Nam, S. A. Reza, T. H. Le, A. Jarabo, D. Gutierrez, and A. Velten, "Non-line-of-sight imaging using phasor-field virtual wave optics," Nature. (2019).
6. A. Jarabo, J. Marco, A. Muñoz, R. Buisan, W. Jarosz, and D. Gutierrez, "A framework for transient rendering," ACM Trans. Graph. **33** (2014).
7. J. Marco, I. Guillén, W. Jarosz, D. Gutierrez, and A. Jarabo, "Progressive transient photon beams," Comput. Graph. Forum **38** (2019).
8. J. Marco, Q. Hernandez, A. Muñoz, Y. Dong, A. Jarabo, M. Kim, X. Tong, and D. Gutierrez, "Deeptof: Off-the-shelf real-time correction of multipath interference in time-of-flight imaging," ACM Trans. on Graph. **36** (2017).
9. Q. Guo, I. Frosio, O. Gallo, T. Zickler, and J. Kautz, "Tackling 3d tof artifacts through learning and the flat dataset," in *The European Conference on Computer Vision (ECCV),* (2018).
10. V. Arellano, D. Gutierrez, and A. Jarabo, "Fast back-projection for non-line of sight reconstruction," Opt. express **25**, 11574–11583 (2017).
11. G. K. Wallace, "The jpeg still picture compression standard," IEEE Trans. on consumer electronics **38**, xviii–xxxiv (1992).
12. D. Wu, A. Velten, M. O'Toole, B. Masia, A. Agrawal, Q. Dai, and R. Raskar, "Decomposing global light transport using time of flight imaging," Int. J. Comput. Vis. **107** (2014).
13. F. Heide, L. Xiao, A. Kolb, M. B. Hullin, and W. Heidrich, "Imaging in scattering media using correlation image sensors and sparse convolutional coding," Opt. Express **22** (2014).
14. Y. Liang, M. Chen, Z. Huang, D. Gutierrez, A. Muñoz, and J. Marco, "A data-driven compression method for transient rendering," in *ACM SIGGRAPH 2019 Posters,* (ACM, 2019), p. 33.
15. M. O'Toole, F. Heide, L. Xiao, M. B. Hullin, W. Heidrich, and K. N. Kutulakos, "Temporal frequency probing for 5D transient analysis of global light transport," ACM Trans. Graph. **33** (2014).
16. R. Ng, R. Ramamoorthi, and P. Hanrahan, "All-frequency shadows using non-linear wavelet lighting approximation," ACM Trans. Graph. **22** (2003).
17. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition,* (2016), pp. 770–778.