Shape from Blur: Recovering Textured 3D Shape and Sub-frame Pose of Motion-Blurred Objects

Anonymous Author(s)
Affiliation
Address
email

Abstract

We address the novel task of jointly reconstructing the 3D shape, pose , texture and motion of an object from a single motion-blurred image including the decomposition of objects and background. While previous approaches address the deblurring problem only in the 2D image domain, our proposed rigorous modeling of all object entities in the 3D domain enables the correct description of arbitrary object motion leads to significantly better color correspondence estimates and sharper deblurring results. We assume the observed appearance of a motion-blurred object to be a combination of the background and a 3D object with constant speed and rotation. Our method minimizes a loss on reconstructing the input image via differentiable rendering with suitable regularizers. This enables estimating the textured 3D mesh of the blurred object with high fidelity . Our method outperforms competing approaches on several benchmarks of fast moving objects deblurring. Qualitative results show that the reconstructed 3D mesh is able to generate high-quality temporal super-resolution and novel views of the deblurred object.

1 Introduction

Motion blur is a common cause of degraded image quality. It can originate from camera motion, fast object motion, long exposure times due to low light settings, or a combination of these effects. In this paper we focus on deblurring fast moving objects (FMOs), which move over a distance larger than their size for the duration of the camera exposure of a single image. Since FMOs appear as a mixture of motion-blurred object texture and the background, our goal is to decompose the combined appearance of the background and the one of the FMO while recovering a sharp, deblurred texture of the object. Please see Fig. 1 for examplary FMO deblurring results of our method. The accurate reconstruction and motion estimation of FMOs enables to generate videos with temporal super-resolution, as well as FMO motion analysis and prediction. This is useful for a variety of applications like sports analysis (soccer, tennis, baseball, badminton, etc.) or fast meteorites detection in astrophysics. Our work could eventually make FMO tracking accessible to everyone using regular cameras rather than expensive high-speed cameras or further push their capabilities.

The majority of deblurring methods aim for the generic task of removing any kind of blur in images [1, 2, 3, 4] or videos [5, 6, 7, 8]. However, they perform poorly on deblurring FMOs as shown in comparisons to specialized methods [9, 10]. All these methods try to tackle the deblurring task in the 2D image domain. This limits their ability to describe object rotations around any axis that does not correspond to the camera viewing direction, or any object motion along the viewing direction.

Our key idea is to model the object motion, rotation and texture in the 3D domain to better explain the input image as the projection and temporal aggregation of a continuously moving object. Although we are solving a more general and more challenging problem than previous works, our method achieves

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

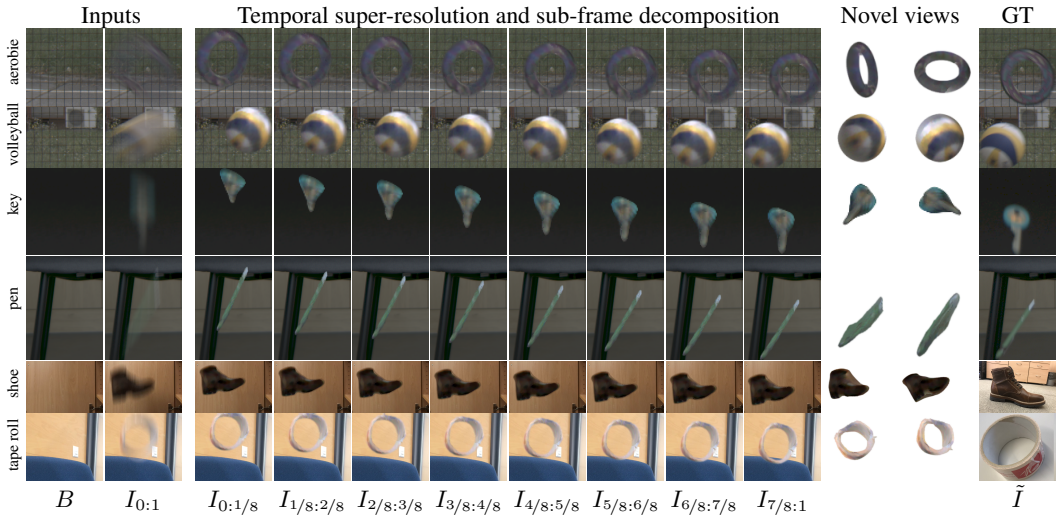


Figure 1: **Temporal super-resolution and deblurring with 3D reconstruction of motion-blurred objects from a single frame.** Solely from the inputs on the left (clean background B, blurred image $I_{0:1}$), the proposed Shape-from-Blur (SfB) method reconstructs a high-quality textured 3D shape by minimizing the input image reconstruction loss with suitable regularizers, which allows to generate a mini-video of subframes $I_{0:1/8}, \ldots, I_{7/8:1}$ from a single image. Moreover, novel views of the reconstructed shape can be generated. The ground truth (GT) on the right shows the corresponding high-speed camera footage or a picture of the static object.

substantially better results due to the rigorous joint modeling of object shape, pose, appearance and motion in 3D.

In summary, we make the following **contributions: (1)** We propose a novel generalized fast moving object deblurring method which for the first time jointly estimates the 3D shape, pose, appearance of an object, as well as its translational and rotational velocities from a single input image. **(2)** We describe an effective end-to-end trainable pipeline which uses a deep neural network for FMO detection and pure test-time optimization for the recovery of the object 3D shape and sharp appearance. To constrain the optimization, we propose several suitable regularizers.

2 Related work

Constant 2D object appearance. The image formation with motion blur of a fast moving object (FMO) [9, 11] was introduced as

$$I = H * F + (1 - H * M)\, B \ , \tag{1}$$

where the input image I is formed as a composition of two terms. The first term represents the blurred object as a convolution of the sharp object appearance F and the blur kernel H. The second term encodes the influence of the background B, which depends on the object silhouette/mask M. This model assumes that the object appearance is constant throughout the whole exposure time. Tracking by Deblatting (TbD) [12], from deblurring and matting, was proposed to estimate the unknown sharp object model (F, M) and its motion represented by H. To achieve this, Kotera et al. [12] applied ADMM variable splitting with appearance total variation and blur kernel sparsity priors. Follow-up methods improve the precision of the 2D trajectory estimation [13, 14]. A special case of planar object with planar rotation is studied in [15].

Piece-wise constant 2D object appearance. The constant appearance model (1) was extended to piece-wise constant [16] that can deal with some amount of rotation or appearance changes over time, defined as

$$I = \sum_i H_i * F_i + \big(1 - \sum_i H_i * M_i\big)\, B \ , \tag{2}$$

where the exposure time is split into several uniform time intervals i. Each time interval has its own dedicated sub-frame appearance model (F_i, M_i) and blur kernel H_i, which are constant within the interval. TbD-3D [16] proposed to estimate these piece-wise appearances by energy optimization with several priors, such as rotational symmetry, appearance total variation, and appearance consistency across neighboring time intervals. The blur kernel is still estimated by deblatting [12] assuming the simple constant appearance model (1), and the sub-frame appearances are estimated in a second pass by splitting the trajectory into the desired number of pieces. Thus, TbD-3D performs well only with very small appearance changes and spherical objects . For perfectly spherical objects, TbD-3D can also estimate 3D rotation by minimizing the sphere reprojection error. Then, the 3D location is computed from 2D mask radius. However, the method is not stable and cannot generalize to other shapes.

2D projections of any object. DeFMO [10] was the first learning-based method for FMO deblurring. The image formation model is generalized further as

$$I = \int_0^1 F_t M_t \, \mathrm{d}t + \Big(1 - \int_0^1 M_t \, \mathrm{d}t\Big) B \ , \tag{3}$$

where the sharp object appearance (F_t, M_t) is defined for any real-valued time stamp t as an output of their encoder-decoder network architecture. In (3), the 2D object motion is directly encoded in the masks M_t. The network is trained end-to-end on a synthetic dataset with various moving and rotation 3D objects . DeFMO generalizes well to real-world data, but the outputs are still just 2D projections of the moving object. There is no guarantee that they are consistent in 3D , and there is no straightforward solution to estimate the 3D shape from 2D DeFMO outputs. The 3D location and 3D rotation is not estimated either.

3D shape from sharp image. Estimating 3D shape from a single sharp image is a hot topic in computer vision. Recent methods are based on differentiable or neural rendering [17, 18, 19, 20] and Neural Radiance Fields (NERF) [21, 22, 23]. Many of those methods heavily depend on shape priors and/or large datasets of training images (of an object class [19, 20, 23], sometimes even multiple views of the same object [21]) . Over the last 30 years, a variety of methods have been proposed for monocular 3D shape estimation from other clues such as Shape from Shading [24], Shape from Texture [25], Shape from Focus [26], Shape from Defocus [27], Shape from Shadow [28], Shape from Specularities [29], and others. We augment the family of "Shape from X" methods with a new Shape from Blur (SfB). To some degree, SfB can be related to Shape from Motion (SfM) [30] and Shape from Silhouettes [31], but in our case, the 2D image sequence is collapsed and averaged into a single image by motion blur.

In contrast to all previous methods , we explicitly estimate textured 3D shape of any object with significant motion blur from a single image.

3 Method

We assume that a triangular mesh is represented by its parameters Θ: vertices, faces, a texture map, and fixed texture mapping. The camera is assumed to be fixed. Then, we define rendering functions $\mathcal{R}_F$ and $\mathcal{R}_M$ that take a mesh Θ as input and output the 2D projected appearance $F = \mathcal{R}_F(\Theta)$ and mask/silhouette $M = \mathcal{R}_M(\Theta)$. Differentiable Interpolation-based Renderer (DIB-R) [17] is used to approximate functions $\mathcal{R}_F$ and $\mathcal{R}_M$ to make the rendering step differentiable.

Furthermore, we define a mesh motion function $\mathcal{M}$ that rotates mesh vertices around their center of mass by a 3D rotation ϕ and translates them by a 3D vector δ. The result is a new mesh $\Theta^1 = \mathcal{M}(\Theta, \phi, \delta)$. Then, the generalized image formation model with a fast moving object is defined as

$$I_{t_0:t_1} = \int_{t_0}^{t_1} \mathcal{R}_F\big(\mathcal{M}(\Theta, \phi_0 + t\phi_1, \delta_0 + t\delta_1)\big) \, \mathrm{d}t + \Big(1 - \int_{t_0}^{t_1} \mathcal{R}_M\big(\mathcal{M}(\Theta, \phi_0 + t\phi_1, \delta_0 + t\delta_1)\big) \, \mathrm{d}t\Big) B \ , \tag{4}$$

which can be seen as a generalization of all previous formation models (1), (2), (3). In comparison to all previous models, the 3D shape is explicitly modelled by (4).

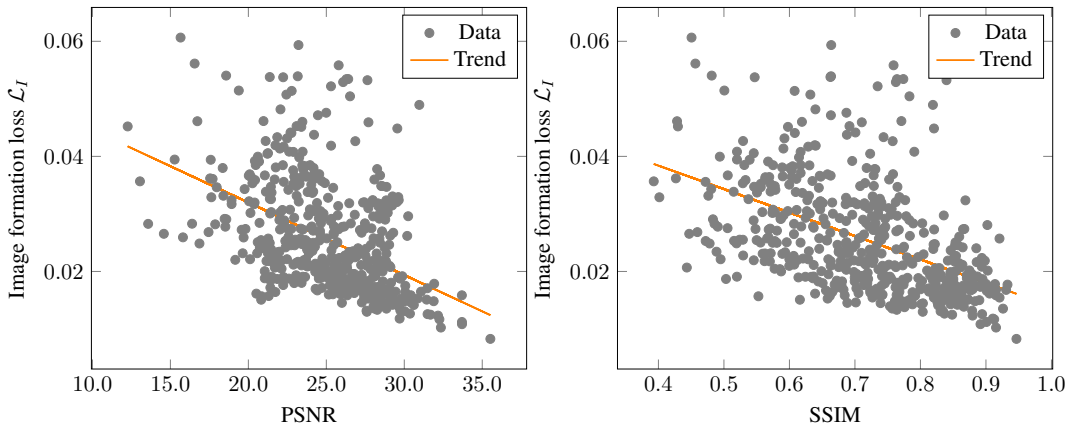


Figure 2: **Correlation between the image formation loss and the testing ground truth scores (PSNR/SSIM),** averaged on all test datasets. The scores are computed after optimizing the image formation loss $\mathcal{L}_I$ (5) with additional regularizers (7) for 500 iterations. Linear regression of the measured data shows the trend: the lower the image formation loss, the better the ground truth scores. Thus, it can be viewed as a surrogate for the ground truth scores.

3.1 Optimization

The main component of the proposed method is the image formation loss that measures the input image reconstruction according to (4). It is defined as the difference between the observed input image and its reconstruction by the generalized formation model:

$$\mathcal{L}_I(\Theta, \phi_0, \phi_1, \delta_0, \delta_1; I, B) = \|I - I_{0:1}(\Theta, \phi_0, \phi_1, \delta_0, \delta_1; B)\|_1 \ . \tag{5}$$

We observed that the image formation loss is directly related to the quality of reconstruction and deblurring measured by the ground truth scores (Fig. 2).

However, directly minimizing the image formation loss (5) is a complex optimization problem since it is highly ill-posed, under-constrained, and with many local minima. Therefore, additional priors and regularizers are vital.

Silhouette consistency. The most important regularization is the approximate location of the object silhouette for a good initial shape estimation. We use sub-frame masks M_t estimated by DeFMO [10] as the regularization, which especially help in the initial optimization steps. The silhouette consistency is defined as

$$\mathcal{L}_S(\Theta, \phi_0, \phi_1, \delta_0, \delta_1) = 1 - \int_0^1 \mathrm{IoU}\Big(M_t, \mathcal{R}_M\big(\mathcal{M}(\Theta, \phi_0 + t\phi_1, \delta_0 + t\delta_1)\big)\Big)\, \mathrm{d}t \ , \tag{6}$$

where the Intersection over Union (IoU) between the estimated mesh silhouettes and the DeFMO masks is computed.

Laplacian regularization. We assume that the mesh geometry (the set of faces) is fixed. Only the vertices offsets are estimated, where the initial vertices are given by some prototype shape. Since the underlying optimization problem is highly under-constrained, many meshes would generate the same input image, especially in the unobserved mesh areas. In order to favor smooth meshes, we add a commonly used Laplacian shape regularization $\mathcal{L}_L(\Theta)$ as in [17, 19, 18].

Texture smoothness. Smooth textures are favored by adding a total variation loss $\mathcal{L}_{TV}(\Theta)$ on the predicted texture map. Thanks to this term, the texture in the unobserved parts of the mesh is a smooth transition between the edges of the observed parts. Without $\mathcal{L}_{TV}(\Theta)$, the unobserved parts contain random noise pixels that degrade the quality of novel view synthesis.

Joint loss. We minimize the joint loss that consists of the image formation loss, the intersection-over-union between the predicted and DeFMO silhouettes, Laplacian shape regularization, and total

Table 1: **Evaluation on FMO benchmark datasets.** The best-performing method is highlighted. Classical deblurring methods [1, 3] do not estimate the object trajectory, thus their TIoU is not defined (N / A). The datasets are sorted from the most challenging one [15] to the easiest one [12] from left to right.

Method	Falling Objects [15]			TbD-3D Dataset [16]			TbD Dataset [12]		
	TIoU↑	PSNR↑	SSIM↑	TIoU↑	PSNR↑	SSIM↑	TIoU↑	PSNR↑	SSIM↑
Jin et al. [3]	N / A	23.54	0.575	N / A	24.52	0.590	N / A	24.90	0.530
DeblurGAN [1]	N / A	23.36	0.588	N / A	23.58	0.603	N / A	24.27	0.537
TbD [12]	0.539	20.53	0.591	0.598	18.84	0.504	0.542	23.22	0.605
TbD-3D [16]	0.539	23.42	0.671	0.598	23.13	0.651	0.542	25.21	**0.674**
DeFMO [10]	0.684	26.83	0.753	0.879	26.23	0.699	0.550	25.57	0.602
SfB (ours)	**0.701**	**27.18**	**0.760**	**0.921**	**26.54**	**0.722**	**0.565**	**25.66**	0.609

variation of the mesh texture. The RGB texture of the mesh comes solely from the image formation loss, and only the shape is approximately guided by the DeFMO masks. The joint loss is defined as a weighted sum

$$\mathcal{L} = \mathcal{L}_I(\Theta, \phi_0, \phi_1, \delta_0, \delta_1; I, B) + \mathcal{L}_S(\Theta, \phi_0, \phi_1, \delta_0, \delta_1) + \mathcal{L}_{TV}(\Theta) + \lambda_L \mathcal{L}_L(\Theta) \ . \tag{7}$$

The learned parameters are the motion representation $\phi_0, \phi_1, \delta_0, \delta_1$ and mesh-related parameters Θ, which are the vertices offsets and the texture map.

Setting. The hyper-parameter for Laplacian shape regularization is set to $\lambda_L = 1000$. The smoothness term for soft mask rasterization in DIB-R [17] is set to 7000^{-1}. The joint loss (7) is optimized end-to-end using ADAM [32] optimizer with the fixed learning rate 0.1 for 500 iterations in PyTorch [33] framework with Kaolin [34] implementation of DIB-R [17]. Optimization for 500 iterations of a mesh with 1212 vertices, 2420 faces, 100×100 texture map, and input image of 240×320 pixels takes around 60 seconds on a single 8 Gb Nvidia GTX 1080 GPU.

Normalization. Texture RGB values are learnable parameters followed by Sigmoid activation for normalization to unit values. The mesh vertices are normalized to zero center of mass and unit variance. Translation vectors δ_0, δ_1 are learnable parameters followed by hyperbolic tangent (tanh) activation and normalization such that the 2D object projection is always visible in the image, i.e.$(-1, -1)$ is bottom left corner of the image, $(1, 1)$ is top right corner of the image, and z coordinate of -1 is an object that covers the whole field-of-view (FOV) and $z = 1$ indicates the object that covers 5% of FOV. Rotations ϕ_0, ϕ_1 are represented by quaternions that are normalized to at most 120 degrees total rotation.

Initialization. The mesh is initialized with uniform grey texture, zero vertices offsets, zero rotation, zero translation, and placed in the middle of the image.

4 Experiments

The 3D mesh geometry, 3D rotation, and 3D translation are evaluated on the synthetic dataset since there are no real-world datasets with blurred objects and their 3D models. The quality of deblurred texture and 2D projection of the estimated motion are evaluated on the real-world datasets, where both the ground truth high-speed camera footage and the low-speed blurry footage for evaluation are available with ground truth annotation. Metric used for the evaluation of 2D motion projection is measured by Trajectory-IoU (TIoU) [12] that averages the standard IoU over the predicted and the ground truth 2D trajectory. The deblurred texture quality is evaluated by Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) metrics on the 2D projection of the mesh, compared to the ground truth sharp frames from a high-speed camera.

4.1 FMO benchmark datasets

The FMO deblurring benchmark [10] includes three datasets. The easiest one is the TbD dataset [12] with uniformly colored mostly spherical objects. TbD-3D dataset [16] is more difficult and contains

Table 2: **Ablation study**. Scores are computed on the Falling Objects [15] dataset and synthetic 3D dataset [10]. The best scores in each block are highlighted. The default setting uses DeFMO [10] masks M_t, all regularization terms, 1 prototype (sphere deformation), and 1×8 sub-frame steps.

		Falling Objects dataset [15]				Synthetic 3D dataset [10]			
	Version	$\mathcal{L}_I \downarrow$	TIoU↑	PSNR↑	SSIM↑	$\epsilon_\delta \downarrow$	$\epsilon_\phi \downarrow$	$\epsilon_\Theta \downarrow$	Time [s]↓
Silhouettes	SfB (no M_t)	0.0371	0.341	17.94	0.594	1.608	23.9°	0.039	58
	SfB (diff. image)	0.0264	0.591	20.71	0.680	2.372	36.0°	0.030	58
	SfB (M_t)	**0.0234**	**0.693**	**26.30**	**0.745**	**0.305**	**17.1°**	**0.023**	63
	SfB (M_t and F_t)	0.0302	0.674	25.64	0.734	0.553	20.2°	0.024	63
Prototypes	SfB (1 prototype)	0.0234	0.693	26.30	0.745	0.305	17.1°	0.023	63
	SfB (2 prototypes)	0.0220	0.695	26.43	0.744	0.304	13.2°	0.023	124
	SfB (3 prototypes)	0.0219	**0.701**	26.57	**0.748**	**0.194**	**11.1°**	0.023	241
	SfB (+ 4 distractors)	**0.0215**	0.696	**26.65**	**0.748**	0.250	15.7°	**0.022**	421
Terms	SfB (no Laplacian)	0.0269	0.676	25.84	0.735	0.462	**14.2°**	**0.023**	61
	SfB (no TV)	0.0276	0.690	24.98	**0.751**	**0.286**	17.5°	**0.023**	62
	SfB (all terms)	**0.0234**	**0.693**	**26.30**	0.745	0.305	17.1°	**0.023**	63
Time steps	SfB (1×8 steps)	0.0234	0.693	26.30	**0.745**	0.305	17.1°	0.023	63
	SfB (3×8 steps)	0.0212	0.690	26.36	0.738	0.320	19.5°	**0.021**	158
	SfB (5×8 steps)	**0.0209**	**0.698**	**26.39**	0.736	**0.293**	**17.0°**	0.023	253
	SfB (10×8 steps)	0.0210	**0.698**	26.25	0.734	0.329	18.1°	0.022	502
	SfB (best of all)	**0.0191**	**0.701**	**27.18**	**0.760**	**0.194**	**11.0°**	**0.024**	1969

objects with complex texture and significant 3D rotation. However, the objects are still mostly spherical. The most recent Falling Objects [15] dataset is the most challenging one with objects of complex 3D shape (e.g. key, pen, marker) and significant rotation.

The compared methods include DeblurGAN-v2 [1] and Jin et al. [3], designed for generating a video sequence from a single image. We also compare to methods specifically designed for FMO deblurring, which are the TbD method [12], TbD-3D method [16], and the most recent DeFMO [10]. Notice that there are no methods that could estimate 3D shape from a blurred image. Hence, the quality of the 3D reconstruction is evaluated only in the ablation study.

According to the ablation study (Table 2), we use the best performing version of SfB with DeFMO masks, all regularization terms, 7 prototypes (including 4 distractors), and 5 times more steps than the desired temporal super-resolution. The proposed SfB method outperforms all other methods on all metrics on two most challenging datasets (Table 1). On the easiest TbD dataset, SfB achieves the highest TIoU and PSNR, but is slightly worse than TbD-3D on the SSIM metrics since the TbD-3D method is designed for uniformly colored spherical objects. Qualitative results on the TbD-3D dataset are shown in Fig. 3 and on the Falling Objects dataset in Fig. 4.

4.2 Ablation study

We evaluate the ground truth scores of all ablated versions on the real-world Falling Objects dataset [15, 10]. The mesh geometry and the accuracy of the 3D translation and 3D rotation prediction are evaluated on the synthetic dataset [10] with moving 3D ShapeNet [35] objects. Since the canonical mesh representation of the ShapeNet objects is usually different from the one estimated by the proposed method, we cannot directly compare absolute values of 3D translation and 3D rotation. Instead, we only compare the error in predicting the translation change δ_1 and rotation change ϕ_1. The translation error is measured as $\epsilon_\delta = \|\delta_1 - \delta_1^{\text{GT}}\|_2$. The rotation error ϵ_ϕ is the angle between ϕ_1 and ϕ_1^{GT}. Here, the ground truth rotation and translation changes are denoted by ϕ_1^{GT} and δ_1^{GT}. The predicted and ground truth ShapeNet meshes are aligned, and the average bidirectional distance is computed as

$$\epsilon_\Theta = {}^1\!/_2(\text{D}(\Theta, \Theta^{\text{GT}}) + \text{D}(\Theta^{\text{GT}}, \Theta)) \ , \tag{8}$$

where $\text{D}(\Theta_1, \Theta_2)$ is the average orthogonal distance of each vertex of mesh Θ_1 to the closest face of mesh Θ_2.

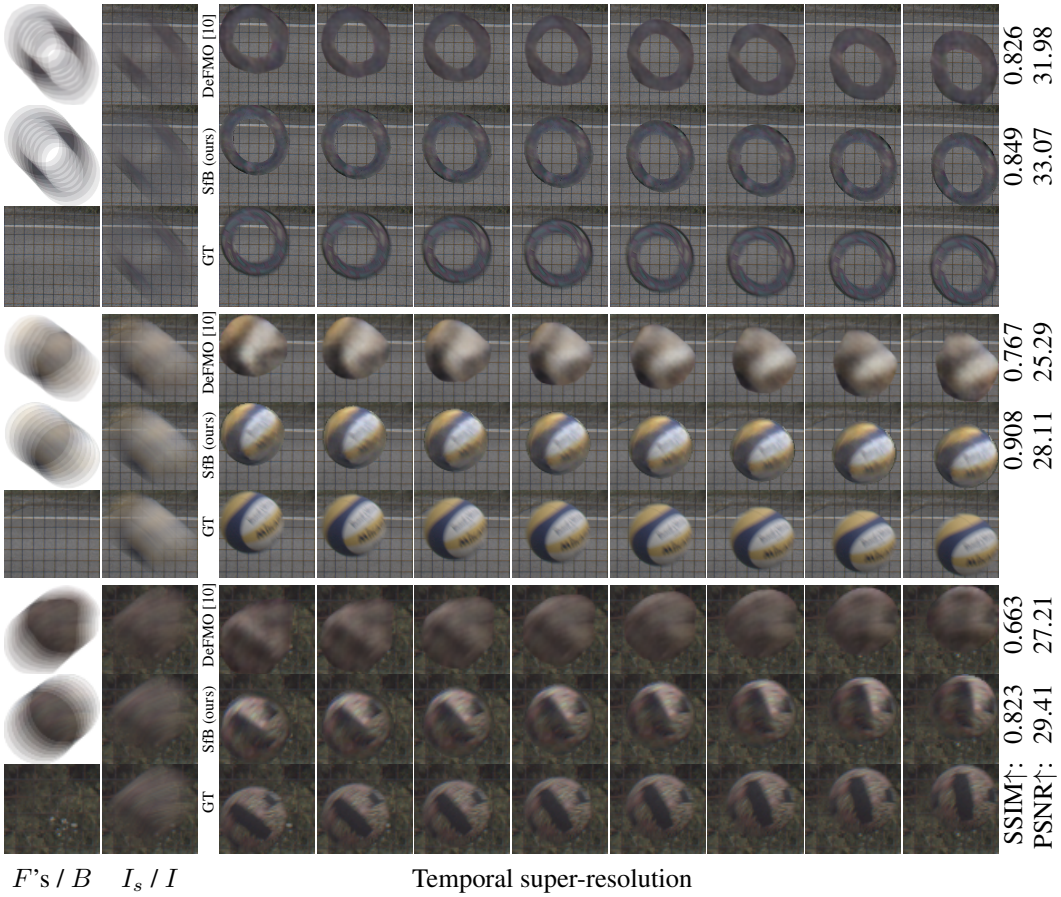


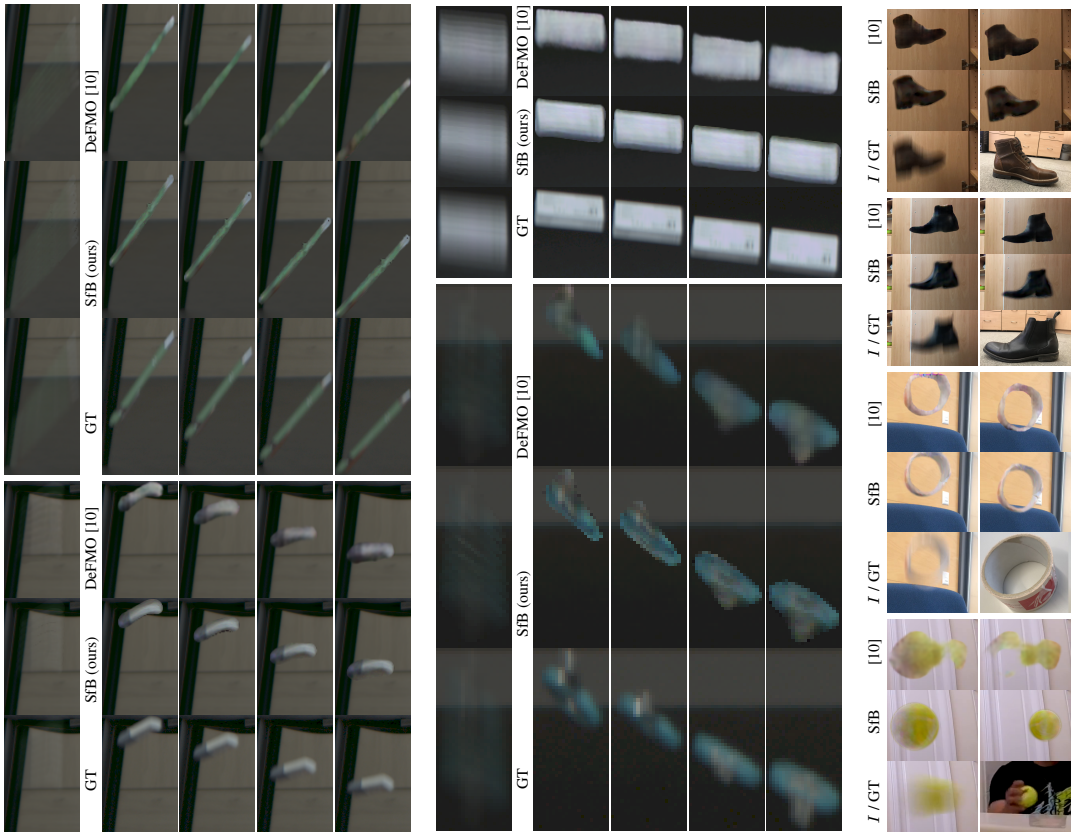
Figure 3: **Results on the TbD-3D [16] dataset.** Examples include from top to bottom: aerobie, volley ball, soccer ball. We compare the proposed Shape-from-Blur (SfB) method with the previous state-of-the-art DeFMO [10] and the ground truth from a high-speed camera. SfB is the first method to reconstruct textured 3D shapes from a single blurred image. The reconstructed input image I_s from sharp appearances F's and background B is almost indistinguishable from the real input image I.

We generate 10 random objects from ShapeNet classes moving in front of random VOT [36] backgrounds. The average ground truth translation change is around 3 object sizes, and the rotation change is around $100°$. The meshes are rescaled to tightly fit a unit cube, therefore one unit of ϵ_Θ and ϵ_δ is equal to object size (diameter).

The first block in the ablation study (Table 2) shows that omitting the DeFMO [10] silhouette consistency degrades the scores by a wide margin. On the other hand, adding another L2 term to force the similarity to DeFMO RGB appearance reconstruction hurts the performance. Another ablated solution uses the difference image between the input image and the background as an approximate location of the object, which improves the results compared to completely omitting the silhouette consistency, but still is far from the performance with DeFMO masks.

The second block shows that the Laplacian shape regularization and texture total variation (TV) bring further improvement. However, since the shapes and textures in the synthetic dataset are significantly more complex (e.g. guitar, jar, skateboard, sofa, bottle, pillow), the Laplacian and TV increase the 3D errors by a small margin on this synthetic dataset.

The third blocks demonstrates that the more prototypes are used, the better are the results. The first prototype is sphere with 1212 vertices, 2420 faces, and simple spherical projection texture mapping. The second prototype is a bigger sphere with 1538 vertices, 3072 faces, and with more sophisticated Voronoi atlas mapping [37] that builds an atlased parametrization based on geodesic Voronoi partitioning of the surface and parametrizes each region using Harmonic Mapping. The third



I_s / I Temporal super-resolution I_s / I Temporal super-resolution FMOs in the wild

Figure 4: **Results on the Falling Objects [11] dataset and FMOs in the wild.** Left part contains results on the pen (top) and eraser (bottom) examples, and the middle part shows the box (top) and key (bottom) examples. The right part contains reconstructions of thrown objects captured by a mobile device camera or of object found on YouTube.

prototype is a torus with a similar number of vertices and faces and the same Voronoi partitioning. The distractor shapes include a teapot, a teddy, a bunny, and a cow shape. Adding them changes the scores only marginally since they are rarely chosen.

The last block evaluates the influence of the number of steps used for discretizing the integral in the generalized formation model (4). Since for Falling Objects dataset, the ground truth high-speed camera footage is available at 8 times higher frame rate than the input video, the temporal super-resolution is generated as a sequence $I_{0:1/8}, \ldots, I_{7/8:1}$ following the same generalized formation model (4).

Applications. The main application of Shape from Blur is temporal super-resolution generation. Given a video stream that contains a significantly motion-blurred object, SfB can generate a video of arbitrary higher frame rates with correctly modeling the object 3D motion (Fig. 3). Since the output of the proposed method is a textured 3D shape of the object in motion, we can also generate novel views of the given object (Fig. 5). SfB also correctly recovers the 3D shape and deblurred texture of various moving objects captured by a mobile device (Fig. 4, shoes, tape) or found on YouTube (Fig. 4, tennis).

Estimation of 3D object rotation has application in sports analysis, e.g. in professional tennis games. If the intrinsic camera parameters are known, object speed and rotation in real-world units can also be estimated.

Limitations and future work. Among limitations of the proposed method are estimation inconsistencies of unobserved parts of the object. Since the shape estimation starts from deforming a

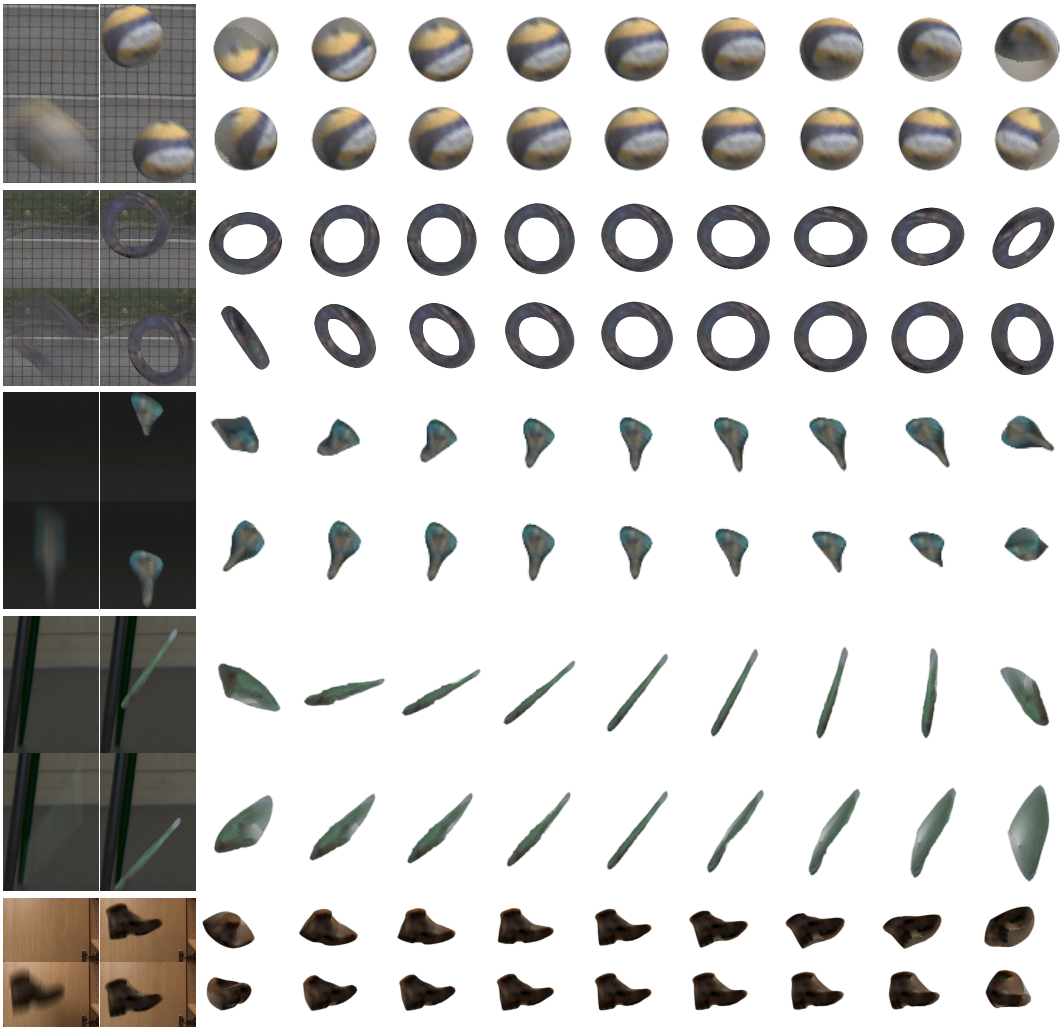


Figure 5: **Novel view synthesis,** shown in x and y canonical axis and rotated from $-60°$ to $60°$. Given the inputs on the left, the proposed method generates textured 3D shape and estimates sub-frame object location and rotation, which can be used to re-generate the input image as a mini-video (temporal super-resolution, TSR) or to generate novel views of the object.

prototype shape (mostly spherical), the shape occasionally stays spherical in the unobserved direction (Fig. 5, pen).

To address some of these shortcomings, the future work includes adding shape priors, such as a symmetry prior or by learning from data. Since the whole optimization is end-to-end differentiable, SfB can be incorporated into a deep learning pipeline to benefit from large training datasets.

The proposed method can be generalized to include multiple frame optimization. For instance, when optimizing the textured 3D shape for several consecutive frames, the object becomes visible from more angles, which gives more constraints and less ambiguities in the unobserved parts.

5 Conclusion

We introduced a novel task to estimate textured 3D shapes of motion-blurred objects from a single image with a known background. The proposed method is the first solution to the stated problem. The introduced Shape-from-Blur method states a new state of the art on fast moving object deblurring datasets.

References

[1] O. Kupyn, T. Martyniuk, J. Wu, and Z. Wang, "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better," in *ICCV*, Oct 2019.

[2] M. Jin, Z. Hu, and P. Favaro, "Learning to extract flawless slow motion from blurry videos," in *CVPR*, June 2019.

[3] M. Jin, G. Meishvili, and P. Favaro, "Learning to extract a video sequence from a single motion-blurred image," in *CVPR*, June 2018.

[4] L. Pan, R. Hartley, M. Liu, and Y. Dai, "Phase-only image based kernel estimation for single image blind deblurring," in *CVPR*, June 2019.

[5] J. Pan, H. Bai, and J. Tang, "Cascaded deep video deblurring using temporal sharpness prior," in *CVPR*, June 2020.

[6] Z. Zhong, Y. Gao, Y. Zheng, and B. Zheng, "Efficient spatio-temporal recurrent neural network for video deblurring," in *ECCV*, pp. 191–207, Springer, 2020.

[7] W. Shen, W. Bao, G. Zhai, L. Chen, X. Min, and Z. Gao, "Blurry video frame interpolation," in *CVPR*, June 2020.

[8] S. Zhou, J. Zhang, J. Pan, H. Xie, W. Zuo, and J. Ren, "Spatio-temporal filter adaptive network for video deblurring," in *ICCV*, 2019.

[9] D. Rozumnyi, J. Kotera, F. Šroubek, L. Novotný, and J. Matas, "The world of fast moving objects," in *CVPR*, pp. 4838–4846, July 2017.

[10] D. Rozumnyi, M. R. Oswald, V. Ferrari, J. Matas, and M. Pollefeys, "Defmo: Deblurring and shape recovery of fast moving objects," in *CVPR*, (Nashville, Tennessee, USA), Jun 2021.

[11] J. Kotera and F. Šroubek, "Motion estimation and deblurring of fast moving objects," in *ICIP*, pp. 2860–2864, Oct 2018.

[12] J. Kotera, D. Rozumnyi, F. Šroubek, and J. Matas, "Intra-frame object tracking by deblatting," in *ICCVW*, Oct 2019.

[13] D. Rozumnyi, J. Kotera, F. Šroubek, and J. Matas, "Non-causal tracking by deblatting," in *GCPR* (G. A. Fink, S. Frintrop, and X. Jiang, eds.), (Cham), pp. 122–135, Springer International Publishing, 2019.

[14] F. Šroubek and J. Kotera, "Motion blur prior," in *ICIP*, pp. 928–932, 2020.

[15] J. Kotera, J. Matas, and F. Šroubek, "Restoration of fast moving objects," *IEEE TIP*, vol. 29, pp. 8577–8589, 2020.

[16] D. Rozumnyi, J. Kotera, F. Šroubek, and J. Matas, "Sub-frame appearance and 6d pose estimation of fast moving objects," in *CVPR*, pp. 6777–6785, 2020.

[17] W. Chen, J. Gao, H. Ling, E. Smith, J. Lehtinen, A. Jacobson, and S. Fidler, "Learning to predict 3d objects with an interpolation-based differentiable renderer," in *NeurIPS*, 2019.

[18] S. Liu, T. Li, W. Chen, and H. Li, "Soft rasterizer: A differentiable renderer for image-based 3d reasoning," *ICCV*, Oct 2019.

[19] N. Wang, Y. Zhang, Z. Li, Y. Fu, W. Liu, and Y.-G. Jiang, "Pixel2mesh: Generating 3d mesh models from single rgb images," in *ECCV*, 2018.

[20] S. Popov, P. Bauszat, and V. Ferrari, "Corenet: Coherent 3d scene reconstruction from a single rgb image," in *ECCV* (A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, eds.), (Cham), pp. 366–383, Springer International Publishing, 2020.

[21] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020.

[22] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," 2020.

[23] K. Rematas, R. Martin-Brualla, and V. Ferrari, "Sharf: Shape-conditioned radiance fields from a single view," *CoRR*, vol. abs/2102.08860, 2021.

[24] E. Prados and O. Faugeras, *Shape From Shading*, pp. 375–388. Boston, MA: Springer US, 2006.

[25] K. ichi Kanatani and T.-C. Chou, "Shape from texture: General principle," *Artificial Intelligence*, vol. 38, no. 1, pp. 1–48, 1989.

[26] S. Nayar and Y. Nakagawa, "Shape from focus," *IEEE TPAMI*, vol. 16, no. 8, pp. 824–831, 1994.

[27] P. Favaro and S. Soatto, "A geometric approach to shape from defocus," *IEEE TPAMI*, vol. 27, no. 3, pp. 406–417, 2005.

[28] S. Savarese, M. Andreetto, H. Rushmeier, F. Bernardini, and P. Perona, "3d reconstruction by shadow carving: Theory and practical evaluation," *IJCV*, vol. 71, no. 3, 2007.

[29] Y. Adato, Y. Vasilyev, O. Ben-Shahar, and T. Zickler, "Toward a theory of shape from specular flow," in *ICCV*, pp. 1–8, 2007.

[30] W. Skarbek, "Shape from motion revisited," in *Active Media Technology* (D. Ślezak, G. Schaefer, S. T. Vuong, and Y.-S. Kim, eds.), (Cham), pp. 383–394, Springer International Publishing, 2014.

[31] S. N. Sinha and M. Pollefeys, "Multi-view reconstruction using photo-consistency and exact silhouette constraints: A maximum-flow formulation," in *ICCV*, ICCV '05, (USA), p. 349–356, IEEE Computer Society, 2005.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR* (Y. Bengio and Y. LeCun, eds.), 2015.

[33] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32* (H. Wallach, H. Larochelle, A. Beygelzimer, F. d Alché-Buc, E. Fox, and R. Garnett, eds.), pp. 8024–8035, Curran Associates, Inc., 2019.

[34] K. M. Jatavallabhula, E. Smith, J.-F. Lafleche, C. F. Tsang, A. Rozantsev, W. Chen, T. Xiang, R. Lebaredian, and S. Fidler, "Kaolin: A pytorch library for accelerating 3d deep learning research," *arXiv:1911.05063*, 2019.

[35] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, and F. Yu, "ShapeNet: An Information-Rich 3D Model Repository," Tech. Rep. arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[36] M. Kristan, J. Matas, A. Leonardis, T. Vojir, R. Pflugfelder, G. Fernandez, G. Nebehay, F. Porikli, and L. Čehovin, "A novel performance evaluation methodology for single-target trackers," *IEEE TPAMI*, vol. 38, pp. 2137–2155, Nov 2016.

[37] U. Pinkall and K. Polthier, "Computing discrete minimal surfaces and their conjugates," *Experimental Mathematics*, vol. 2, no. 1, pp. 15 – 36, 1993.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Limitations are discussed in Section 4.2.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] We believe there are no direct negative societal impacts of our work.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A] We do not include theoretical results.
   (b) Did you include complete proofs of all theoretical results? [N/A] We do not include theoretical results.

3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We include the code to reproduce the main experimental results in the supplemental material. The code contains a script that takes as input an image of a blurred object and the background and produces temporal super-resolution.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] The training details are provided in Section 3.1.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] We do not report error bars since the proposed method is deterministic and is not based on random number generator. The initialization of the proposed optimization is always the same as reported in Section 3.1.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] The total amount of compute and used resources are reported in Section 3.1.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We cited the creators of DeFMO [10], which we use directly in the proposed method. Also, the creators of all used datasets are cited [12, 16, 15]. The other compared methods are also cited [3, 1]. The used datasets are available at `http://cmp.felk.cvut.cz/fmo/`. The FMO benchmark implementation is taken from `https://github.com/rozumden/fmo-deblurring-benchmark`. DeFMO implementation is taken from `https://github.com/rozumden/DeFMO`.
   (b) Did you mention the license of the assets? [Yes] The mentioned assets are open-sourced.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] Implementation of the proposed method is included in the supplemental material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] The used data is open-sourced.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] The used data does not contain personally identifiable information or offensive content.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]