2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)

Sub-frame Appearance and 6D Pose Estimation of Fast Moving Objects

Denys Rozumnyi[3,1] denysr@inf.ethz.ch

Jan Kotera[2] kotera@utia.cas.cz

Filip Šroubek[2] sroubekf@utia.cas.cz

Jiří Matas[1] matas@fel.cvut.cz

[1]Visual Recognition Group, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic

[2]Czech Academy of Sciences, UTIA, Prague, Czech Republic [3]Department of Computer Science, ETH Zurich

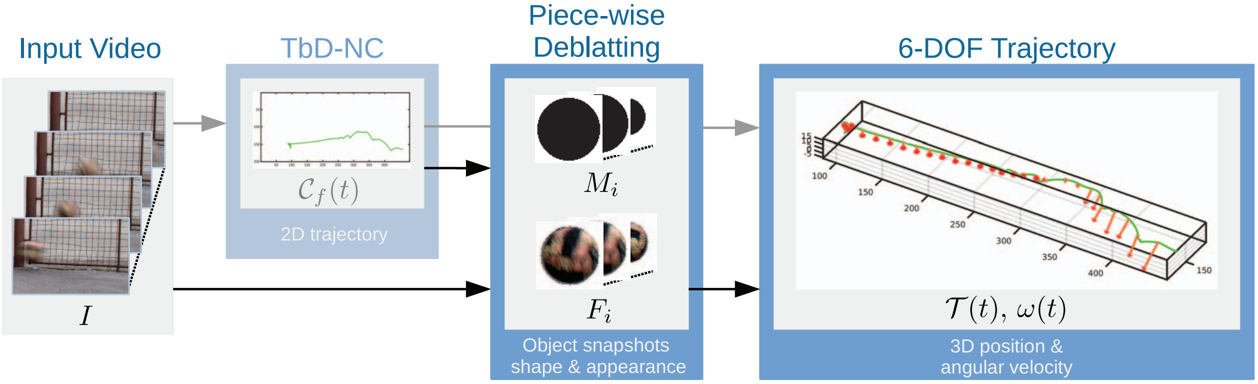


Figure 1: Estimation of appearance, shape and 6D pose (3D position and rotation) of fast moving objects. The input video and 2D trajectories estimated by Non-Causal Tracking by Deblatting, TbD-NC [14], are processed by the proposed piece-wise deblatting that generates, with sub-frame temporal resolution, the object appearance and shape (snapshots), from which the complete 6-DOF trajectory is estimated.

Abstract

We propose a novel method that tracks fast moving objects, mainly non-uniform spherical, in full 6 degrees of freedom, estimating simultaneously their 3D motion trajectory, 3D pose and object appearance changes with a time step that is a fraction of the video frame exposure time. The sub-frame object localization and appearance estimation allows realistic temporal super-resolution and precise shape estimation. The method, called TbD-3D (Tracking by Deblatting in 3D) relies on a novel reconstruction algorithm which solves a piece-wise deblurring and matting problem. The 3D rotation is estimated by minimizing the reprojection error. As a second contribution, we present a new challenging dataset with fast moving objects that change their appearance and distance to the camera. High-speed camera recordings with zero lag between frame exposures were used to generate videos with different frame rates annotated with ground-truth trajectory and pose.

1. Introduction

Visual tracking encompasses a broad class of problems that have received significant interest [7, 8]. Current state-of-the-art methods employ a range of techniques, such as deep Siamese networks [9, 18] and discriminative correlation filters [20, 12]. The standard output of tracking methods is a 2D bounding box, either axis aligned or rotated. Video segmentation methods output precise segmentation masks [22, 21].

Recently, fast moving objects (FMOs) have been introduced as one of the problems in tracking [15]. Such objects are recorded as blurred streaks. They are common in sport videos and many other scenarios, such as videos of falling objects, hailstorm and flying insects, or more specialized ones, *e.g.* visual navigation of microrobots in a magnetic field. To avoid FMOs and the related phenomena, one can use high-speed cameras operating at high frame rates, *e.g.* 240 fps or more. However, this brings additional requirements on resources, such as data transfer bandwidth and storage. When capturing such objects, camera settings have to be considered a priori before video acquisition.

The blurred trace of an object encodes information about its velocity, shape and appearance. Estimating these quantities should be thus in principle possible even from more affordable cameras with 30 fps, but it is a challenging task as the problem is heavily ill-posed. As shown in [15], standard tracking methods do not perform well on FMOs.

2575-7075/20/$31.00 ©2020 IEEE
DOI 10.1109/CVPR42600.2020.00681

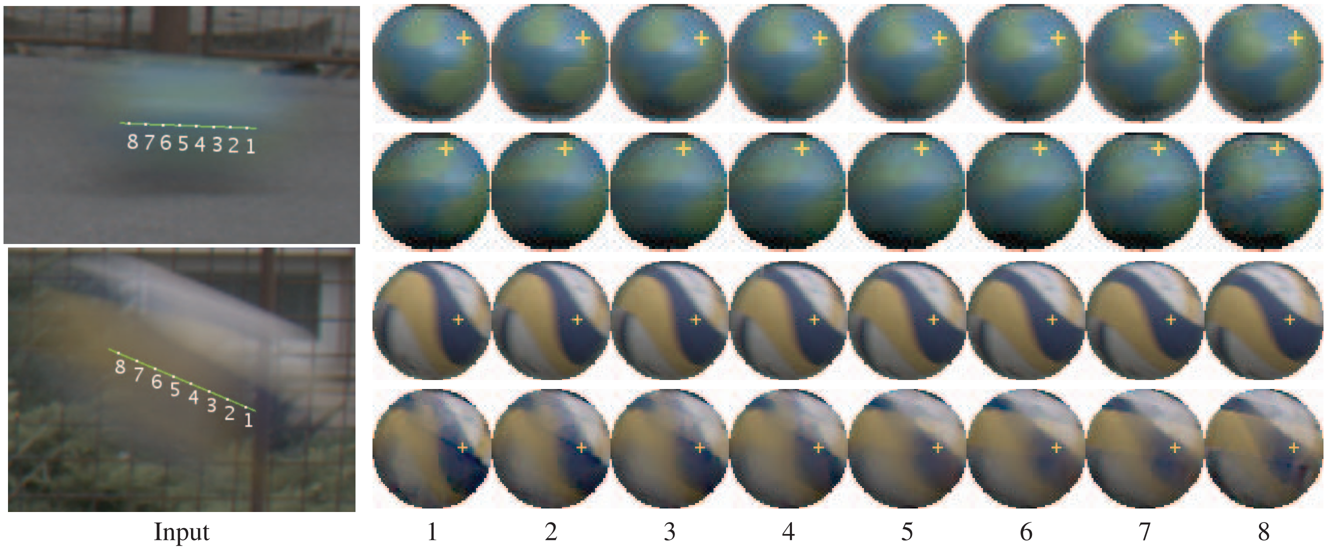


Figure 2: Sub-frame appearance estimation of fast moving objects. Left: 30 fps input images with overlaid 2D projections of recovered 3D trajectories in green. White points correspond to time instants in the middle of high-speed camera frames. Right: cropped objects from a high-speed camera (top) and output of the proposed TbD-3D (bottom). 3D rotation is estimated by minimizing the reprojection error, assuming a spherical object. The estimated rotation axis is visualized by a yellow cross.

For a fast moving object, a bounding box or a segmentation mask is not an adequate representation of its trajectory, as it travels a non-negligible path in a single frame. Such object may be localized more precisely, with a sub-frame accuracy.

Tracking by Deblatting (TbD) [5] was the first method to track fast moving objects by solving a joint *debl*urring and m*atting* (*deblatting*) problem. These techniques are closer to blind deconvolution than to visual tracking methods. Non-causal post-processing proposed in [14] gives more precise and complete trajectories. The output of both above-mentioned methods is only a 2D trajectory. They assume a 2D appearance and mask of an object to stay unchanged over the duration of a frame. This is equivalent to ignoring the 3D rotation of the object, the change of its distance to camera and of appearance due to the non-uniform light field, reflections, shadows or deformations. Such simplifications are only adequate for objects with almost uniform texture and moving in a plane parallel to the camera projection plane. To date, the full nature of 3D object motion and appearance has not been considered, nor object location in 3D nor angular velocity in 3D.

In this paper we are the first to estimate continuous-time sub-frame changes in appearance of the object. While solving for the shape and appearance, we recover the 3D rotation of the object and distance to the camera (currently we are able to handle only close to spherical objects). The output of the proposed method is a continuous object pose with 6 degrees of freedom. The reconstruction pipeline is summarized in Figure 1.

We make the following **contributions**:

- We propose TbD-3D (Tracking by Deblatting in 3D) – the first method to reconstruct the appearance and the shape of blurred moving objects with sub-frame temporal resolution using piece-wise deblatting. We call these reconstructions snapshots (as in Figure 2).
- The method estimates continuous-time pose with 6 degrees of freedom (3D location and rotation) for non-uniform spherical objects. The rotation is estimated by a new method which minimizes the reprojection error.
- We collect and make available a new challenging dataset with fast moving objects that change their appearance and distance to the camera. High-speed camera with zero lag between frame exposures is used to generate videos with different low frame rates annotated with ground-truth trajectory and pose data.
- Sub-frame reconstruction accuracy on object deformations that occur during contact with other objects is demonstrated.

2. Related Work

Detection and tracking of fast moving objects was introduced by Rozumnyi *et al*. [15]. Their work was limited by several assumptions on object trajectory and appearance, such as linear trajectory parallel to the camera projection plane, uniform 2D appearance of the object, high contrast to the background and no contact of the moving object with other objects. Some of these assumptions were relaxed in a method called Tracking by Deblatting (TbD) [5], which tracks fast moving objects by solving a deblurring problem in every frame and processing the video in a causal manner. TbD outperforms the previous approach by a wide margin, yet trajectories estimated at adjacent frames are independent and the final trajectory for the whole sequence is a set of segments. These limitations are addressed by a follow-up method – non-causal Tracking by Deblatting (TbD-NC) [14]. TbD-NC takes the output of TbD and estimates the final trajectory which is continuous over the duration of the whole sequence.

All these methods assume that the object trajectory is parallel to the camera plane and that the object appearance is static within one frame (no rotation). The appearance can change between frames, but in arbitrary fashion as a long-term appearance template is learned online. The only exception is the work of Kotera and Šroubek [6] that estimates object rotation, yet only 2D in-plane rotation is considered and the object shape is assumed to be known. The method is thus applicable only to nearly flat objects moving on a flat surface.

Deep learning has been applied to motion deblurring of videos [19, 17] and to the generation of intermediate short-exposure frames [4]. Their proposed convolutional neural networks are trained only on small blurs; blur parameters are not available as they are not directly estimated. Tracking methods that consider blurred objects in [13, 10] assume object motion that is approximately linear and relatively small compared to the object size. Alpha blending of the tracked object with the background is ignored and their output per frame is only a bounding box, which is insufficient for fast moving objects.

The tracking methods [15, 5, 14] for fast moving objects use an image formation model that is defined as

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

for a single color video frame I. The formation model is a mixture of two components. The first component is the object appearance F (after projection to the image plane) blurred by motion along the object trajectory, which is represented as a blur kernel H. The second part is the background B attenuated by object occlusion, where M, equivalent to the indicator function of F, is the object shape after projection to the image plane. Following [5], the blur is simplified to a 2D convolution. The background B is estimated as a median of the last 5 frames. They assume either an almost static camera or a stabilized sequence.

The output of TbD-NC [14] is a 2D object trajectory $\mathcal{C}_f(t)$: $[0, N] \to \mathbb{R}^2$ in an analytical form where N is the number of frames in the video sequence. This output is then used as an input to the proposed TbD-3D method.

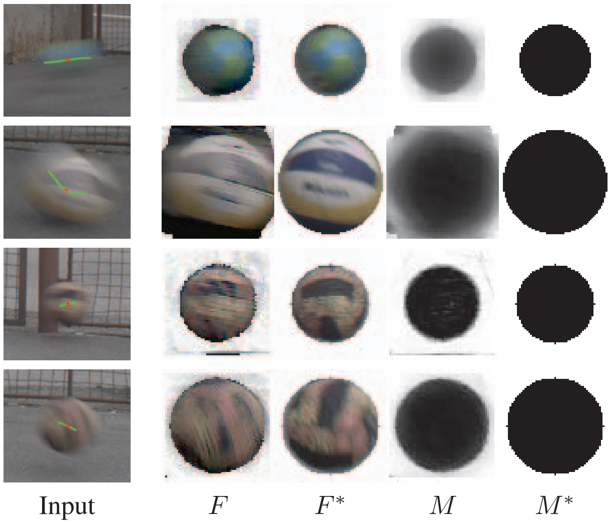


Figure 3: TbD-3D applied to 30 fps videos compared to images from a high-speed camera at 240 fps (marked with $*$). F: snapshots of object appearance estimates of fast moving objects. Each row corresponds to one sub-frame instant (red dot on a green trajectory) of the input frame on the left. For visualization purposes, the masks M are inverted.

3. Method

We propose the following pipeline to reconstruct a 6DoF pose of a fast moving object:

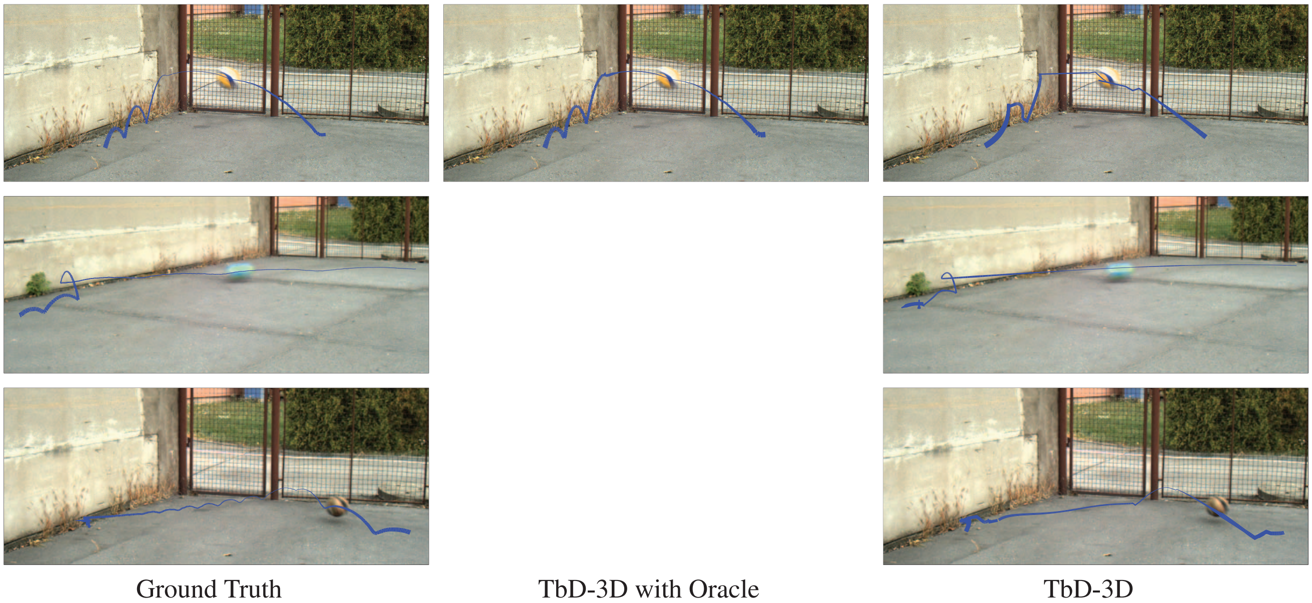
1. From a given 2D trajectory, in our case computed by the TbD-NC algorithm [14], reconstruct sub-frame blur-free snapshots of the object by piece-wise deblatting (Section 3.1).

2. Estimate the relative distance from the object to the camera from the estimated shape mask (Section 3.2).

3. Using the assumption of a spherical object with locally constant rotation find the rotation axis and velocity by minimizing the reprojection error (Section 3.3).

An alternative method to estimate the 3D rotation of FMOs from their snapshots is a classical 3D reconstruction pipeline. We tried several reconstruction and structure-from-motion pipelines [16, 11, 2, 3] and none of them were able to deal with small objects containing few features. They do not perform well even on snapshots from a high-speed camera sequence, where the motion blur is negligible.

Tracking by Deblatting in 3D thus extends TbD and TbD-NC by using trajectories estimated by these methods to infer more attributes about the object and its motion. The core of TbD consists of two alternating optimization steps. The first step updates the object shape and appearance (F, M) while the trajectory H is fixed, and the second one updates the trajectory H while the object (F, M)


Ground Truth

TbD-3D with Oracle

TbD-3D

Figure 4: 3D trajectory estimation for selected sequences from the proposed TbD-3D dataset. Curve thickness codes distance from the object to the camera (thicker curve means that the object is closer to the camera). TbD-3D with Oracle means that the 2D trajectory is estimated from the original high-speed footage and only the third dimension is estimated. Otherwise, the output of TbD-NC [14] is used as the 2D trajectory. Sequences correspond to 30 fps.

is fixed. Both steps are formulated as convex optimization problems with non-smooth terms and constraints and solved using the ADMM method [1]. While processing the video sequence, TbD maintains a long-term appearance model $\hat{F}$ used to regularize the estimation of F in the new frame.

We have made two modifications to the TbD core steps. First, we added a new regularization term to the shape-and-appearance (F, M) estimation step to facilitate shape estimation in cases when the tracked object is a ball and its shape is thus circularly symmetric. The modified optimization problem is

$$\min_{F,M} \frac{1}{2} \|H * F + (1 - H * M)B - I\|_2^2 \\ + \frac{\lambda}{2}\|F - \hat{F}\|_2^2 + \alpha_F \|\nabla F\|_1 + \frac{\lambda_R}{2}\|\mathbf{R}M - M\|_2^2, \quad (2)$$

s.t. $0 \leq F \leq M \leq 1$, where matrix inequalities are considered element-wise. The first term is the data likelihood given by the image formation model (1). The second term constrains the solution to be close to the template $\hat{F}$ and the third term is Total Variation that enforces piece-wise smooth object appearance. In the last, λ_R-weighted term, $\mathbf{R}$ is a linear operator that performs circular averaging, i.e. the shape mask M is forced to be rotationally symmetric.

Second, in the estimation of H we replaced the L^1 regularization of H by the constraint $\sum_i H[i] = 1$, which is free of weighting parameters that have to be tuned for different sequences. The modified optimization problem is then

$$\min_{H} \frac{1}{2} \|H * F + (1 - H * M)B - I\|_2^2 , \\ \text{s.t. } H \geq 0, \sum_i H[i] = 1. \quad (3)$$

3.1. Piece-wise Deblatting

TbD assumes that the object appearance and shape is constant during single frame exposure. In reality, the appearance changes even within a single video frame due to the object rotation and camera projection. We propose to approximately model this gradual change as a sequence of constant snapshots which we estimate. The snapshots can be used for temporal super-resolution and also to determine the intra-frame object rotation.

Suppose that the object trajectory in the form of a parametric curve $\mathcal{C} : [0, 1] \to \mathbb{R}^2$ has been estimated for a given video frame. We partition this curve to multiple contiguous segments $\mathcal{C}_i$ with their corresponding blurs denoted H_i and estimate the appearance and shape (F_i, M_i) of the object at the time interval corresponding to $\mathcal{C}_i$. From the piece-wise constant appearance assumption, we get the formation model of the video frame I as

$$I = \sum_i H_i * F_i + \Big(1 - \sum\nolimits_i H_i * M_i\Big) \cdot B. \quad (4)$$

The optimization problem (2) for joint estimation of

Sequence	#	TIoU-3D				Radius Error [pixels]			Axis Error [°]				Angle Error [°]			
		TbD	-NC	-3D	-3D-O	-NC	-3D	-3D-O	TbD-3D		TbD-3D-O		TbD-3D		TbD-3D-O	
		Mean	Mean	Mean	Mean	Mean	Mean	Mean	Mean	Std	Mean	Std	Mean	SE3	Mean	SE3
depthf1	46	0.550	0.579	0.625	**0.937**	3.348	1.333	**1.035**	**59.796**	14	60.881	21	**0.124**	3.8	2.269	5.5
depthf2	50	0.475	0.528	0.599	**0.911**	6.424	3.209	**1.678**	**19.966**	19	22.125	36	1.733	23.5	**0.097**	19.9
depthf3	37	0.317	0.363	0.452	**0.763**	10.986	6.004	**5.397**	21.185	24	**11.932**	25	1.336	1.4	**0.556**	1.8
depth2	48	0.448	0.590	0.626	**0.906**	4.213	2.549	**1.894**	**71.085**	7	85.816	6	**6.715**	6.0	8.242	9.3
depthb2	81	0.366	0.444	0.388	**0.949**	2.101	7.080	**0.850**	**68.061**	3	69.126	3	9.760	13.0	**7.838**	13.3
out1	57	0.465	0.495	0.562	**0.964**	6.865	2.286	**0.705**	47.329	7	**13.308**	26	0.673	3.0	**0.308**	1.6
out2	50	0.503	0.533	0.561	**0.981**	4.251	1.354	**0.369**	**18.259**	3	45.009	14	**0.152**	2.3	0.236	1.5
outb1	41	0.350	0.384	0.431	**0.939**	4.932	3.361	**0.885**	18.856	32	**13.819**	29	1.692	7.2	**0.658**	5.9
outf1	60	0.551	0.587	0.611	**0.968**	3.297	0.924	**0.614**	25.174	13	**12.743**	14	**0.015**	4.1	0.041	2.1
Average	52	0.447	0.500	0.539	**0.924**	5.157	3.122	**1.492**	38.857	13	**37.195**	19	2.467	7.2	**2.250**	6.8

Table 1: TbD-3D dataset – comparison of TbD [5], TbD-NC [14] and the proposed TbD-3D. For each sequence, we report: TIoU-3D (15) to measure the accuracy of 3D object location, radius error, axis error as the average angle between the estimated axis and the ground truth axis, and the angle error in degrees. For each sequence and each score, we highlight the best performing method in bold. TbD-3D-O means TbD-3D with oracle: the 2D object location is known from the ground truth. For axis and angle estimation, the difference between TbD-3D and TbD-3D-O is statistically insignificant and the p-values in both cases are around 0.89. SE3: standard deviation times 10^3. The TbD-3D dataset corresponds to 30 fps frame rate, 8 times lower than the ground truth data from the high-speed camera. Results for other frame rates are shown in Figure 5.

(F_i, M_i) on segments $\mathcal{C}_i$ becomes

$$\begin{aligned} &\min_{F_i, M_i} \frac{1}{2} \Big\| \textstyle\sum_i H_i * F_i + (1 - \sum_i H_i * M_i) B - I \Big\|_2^2 \\ &+ \frac{\lambda}{2} \|F_i - \hat{F}_i\|_2^2 + \alpha_F \|\nabla F\|_1 + \frac{\lambda_R}{2} \sum_i \|\mathbf{R} M_i - M_i\|_2^2 \\ &+ \gamma_F \sum_i \|F_i - F_{i+1}\|_1 + \gamma_M \sum_i \|M_i - M_{i+1}\|_1, \quad (5) \end{aligned}$$

s.t. $0 \leq F_i \leq M_i \leq 1$.

The last two terms, weighted by γ_F and γ_M, are new regularization terms enforcing similarity of both appearance and shape of the object in neighboring time intervals. $\hat{F}_i$ is a sub-frame extension of the appearance template used in TbD, regularizing the appearance estimation in corresponding segments.

The piecewise appearance estimation is implemented in a hierarchical manner. First, we split $\mathcal{C}$ into two segments $\mathcal{C}_1^1$ and $\mathcal{C}_2^1$ (superscript denotes the hierarchy level) and solve (5) for F_1^1, F_2^1 with both templates $\hat{F}_1^1 = \hat{F}_2^1 := F^0$ where F^0 is the initial result of TbD. On the next level, we do another binary splitting of $\mathcal{C}_1^1$ to $\mathcal{C}_1^2, \mathcal{C}_2^2$ and $\mathcal{C}_2^1$ to $\mathcal{C}_3^2, \mathcal{C}_4^2$ and again solve (5) with templates set to results from the previous level, $\hat{F}_1^2 = \hat{F}_2^2 := F_1^1$ and $\hat{F}_3^2 = \hat{F}_4^2 := F_2^1$. This process continues until the desirable number of splitting of $\mathcal{C}$ is achieved. Results of this estimation process are illustrated in Figure 2.

3.2. 3D Trajectory

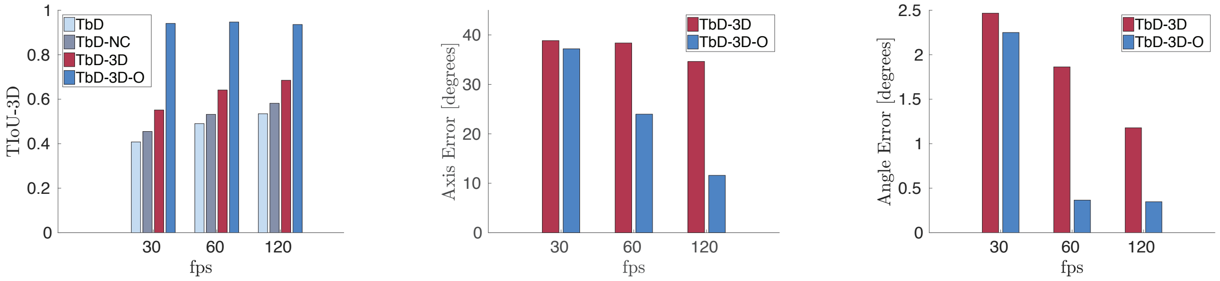
TbD-NC [14] provides a 2D part of the estimated trajectory by fitting piece-wise polynomial curves. We extend this approach to fitting piece-wise polynomial curve in 3D, where the third dimension is the object distance to the camera. We assume that the object is approximately spherical with radius r, *i.e.* the area of mask defined as sum of all pixel values is $\text{area}(M) := \sum_i M[i] = \pi r^2$. The distance d is inversely proportional to the perceived object radius r and is given by

$$d \propto \sqrt{\frac{\pi}{\text{area}(M)}}\,. \quad (6)$$

Note that the absolute distance can be calculated if we know camera parameters and the actual object radius. The estimated relative distances in sub-frame precision are expressed analytically by piece-wise continuous curve fitting. First, bounces are found as initially estimated in 2D trajectory and then additional bounces which are only visible in 3D are added, *e.g.* during motion perpendicular to the camera plane. The bounces separate the trajectory into segments and in each segment we fit a polynomial of degree up to 6. The final trajectory is a function $\mathcal{T}(t)$: $[0, N] \subset \mathbb{R} \to \mathbb{R}^3$, N is the number of frames, defined as

$$\mathcal{T}(t) = \sum_{k=0}^{p_s} \bar{c}_{s,k} t^k \quad t \in [t_{s-1}, t_s], s = 1..S, \quad (7)$$

with S polynomials, where polynomial with index s has degree p_s and it is represented by its coefficient matrix



Location (3DoF) Rotation Axis (2DoF) Rotation Angle (1DoF)

Figure 5: Evaluation of the proposed TbD-3D method on the TbD-3D dataset with different frame rates. We report scores for three settings: 30, 60 and 120 fps. From left to right: TIoU-3D (15) of TbD-3D compared to the TbD [5] and TbD-NC [14] methods, error of rotation axis estimation, error of rotation angle estimation. The errors of rotation axis and angle are measured by a mean average angle between the estimate and the ground truth from the high-speed footage at 240 fps. Oracle with known 2D trajectory from ground truth is marked by "-O". The TbD-3D method performs better in the task of 3D location estimation and provides meaningful results for 3D rotation estimation w.r.t. the ground truth.

$\bar{c}_s \in \mathbb{R}^{3,p_s+1}$. Time stamps t_s split the whole sequence into intervals between 0 and N, such that $0 = t_0 < t_1 < \ldots < t_{S-1} < t_S = N$. The degree of the polynomial depends on the number of frames it is fitted to; the interpolation scheme is similar to [14].

3.3. Angular velocity

In the case of spherical objects, we are able to estimate their angular velocity $\omega \in \mathbb{R}^3$. Following the standard definition in physics, ω is a 3D vector of the rotation axis orientation whose magnitude represents the rotation angle along the axis per time unit. Let $\mathbf{R}_\omega$ be an operator transforming a 2D image of a ball by performing 3D rotation given by ω of a virtual 3D representation of the ball. More specifically, if $F_2 = \mathbf{R}_\omega\, F_1$, then F_2 is given by mapping the 2D image F_1 to a virtual 3D sphere, rotating the sphere by ω and projecting back on the 2D image. The error of the transformation between the two images is defined as

$$E(F_1, F_2|\,\omega) = \|\,\mathbf{R}_\omega\, F_1 - F_2\|_1. \tag{8}$$

Since different parts of the ball are visible before and after rotation, the sum in eq. (8) is carried out only in some predefined region visible in both images after arbitrary considered rotation, so that errors corresponding to different rotations are mutually comparable.

Having recovered the object appearance F_1 and F_2 at two different video sequence timestamps t_1 and t_2, we can find the average angular velocity ω between t_1 and t_2 as the minimizer of the transformation error $E(F_1, F_2|\,(t_2 - t_1)\omega)$. Velocity estimation from just two restored images at close timestamps is prone to errors, especially if either of the images is estimated with artifacts. We therefore state an assumption that angular velocity is locally constant in small time interval of the motion (which is physically nearly correct even in the long term if the ball is in free flight) and estimate angular velocity more robustly in a sliding-window manner from several restored images belonging to the corresponding time-window.

Let $F_1, \ldots, F_n$ be a set of estimated ball appearances at timestamps $t_1, \ldots, t_n$; a short time-window of the whole sequence. We estimate a single average angular velocity ω at this time-window as follows. Let ω_{ij} be the minimizer of the transformation error from F_i to F_j and S_{ij} inverse of the attained error ('score'):

$$\omega_{ij} = \operatorname*{argmin}_{\omega} E(F_i, F_j|\,(t_j - t_i)\omega), \tag{9}$$

$$S_{ij} = \frac{1}{E(F_i, F_j|\,(t_j - t_i)\omega_{ij}) + \varepsilon}. \tag{10}$$

In other words, ω_{ij} is the vote of the corresponding image pair for the true ω and S_{ij} is the confidence of such vote. We minimize (9) by searching the discretized space of feasible angular velocities.

Simply averaging ω_{ij} results in non-robust estimate that is sensitive to outliers. Instead we proceed with RANSAC-like approach. Based on the discretization step used in the minimization of (9), an inlier threshold ρ is defined as the maximum acceptable error in determining ω. We treat ω_{ij}'s as hypotheses for the final estimate ω and for each hypothesis calculate its consensus set C_{ij} as

$$C_{ij} = \{(k,l)\colon \|\omega_{kl} - \omega_{ij}\| \leq \rho\}\,. \tag{11}$$

The winning candidate ω_{pq} is the one with the best total score of its consensus set,

$$(p,q) = \operatorname*{argmax}_{(i,j)} \sum_{(k,l)\in C_{ij}} S_{kl}. \tag{12}$$

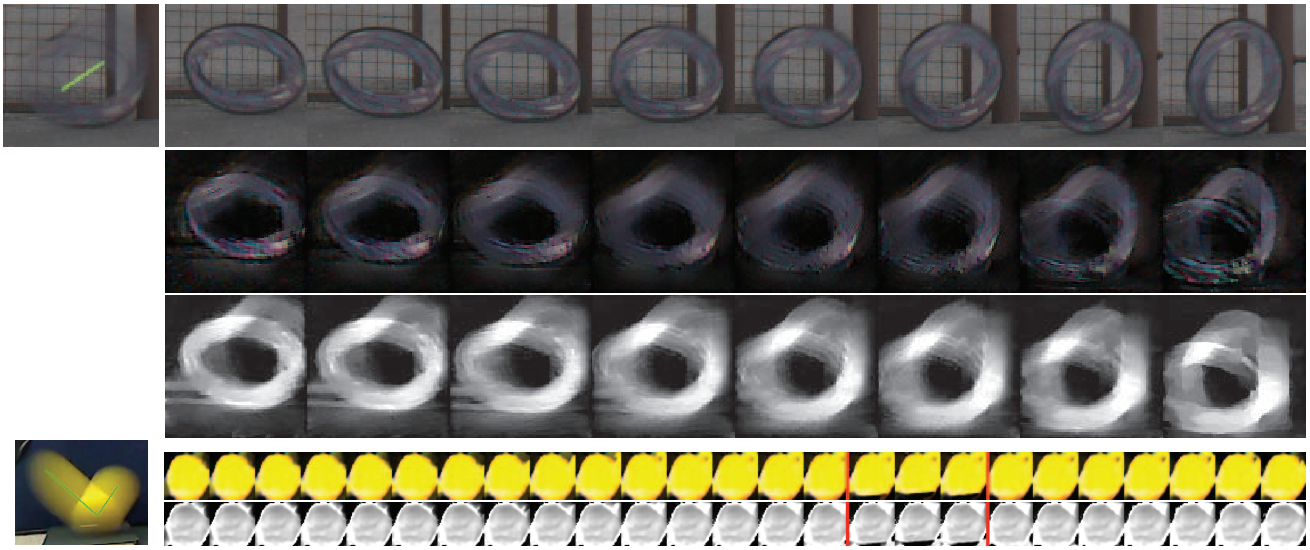


Figure 6: Deformations found using the TbD-3D method. They are not modeled explicitly, but are visible during contact with other objects in the scene. Left: input images with trajectories overlaid in green. Right: crops from high-speed camera footage (top), object appearance F and mask M reconstructions by the proposed TbD-3D method with the uniform split of trajectories. For this experiment, we set the term on rotational symmetry λ_R in eq. (5) to zero. We estimate sub-frame snapshots using only the input frame on the left and the background. The trajectory is split into 8 (top row) or 25 (bottom row) segments. Deformation during a soft ball bounce is visible between the two red bars in the bottom row.

The final estimate is then the weighted average of the votes in the consensus set of the winning candidate

$$\omega = \frac{\sum_{kl} S_{kl}\omega_{kl}}{\sum_{kl} S_{kl}}, \quad (k,l) \in C_{pq}. \tag{13}$$

4. Experiments

Kotera *et al*. [5] introduced Trajectory Intersection over Union (TIoU) to measure the accuracy of estimated trajectories, which is defined as

$$\mathrm{TIoU}(\mathcal{C}, \mathcal{C}^*) = \int_t \mathrm{IoU}\left(M^*_{\mathcal{C}(t)},\, M^*_{\mathcal{C}^*(t)}\right) \mathrm{d}t, \tag{14}$$

where $M^*_{\mathcal{C}(t)}$ denotes the ground truth object mask M^* placed at a 2D point on either the estimated trajectory $\mathcal{C}$ or the ground truth trajectory $\mathcal{C}^*$. Integral is approximated by sum, sampling time at 8 evenly-spaced instants. We extend this measure to 3D trajectories and define TIoU-3D as

$$\text{TIoU-3D}(\mathcal{T}, \mathcal{T}^*) = \int_t \mathrm{IoU}\left(S^*_{\mathcal{T}(t)},\, S^*_{\mathcal{T}^*(t)}\right) \mathrm{d}t, \tag{15}$$

where $S^*_{\mathcal{T}(t)}$ is a ball corresponding to the ground truth radius and located at $\mathcal{T}(t)$, a 3D point along trajectory $\mathcal{T}$ at time-stamp t. Similarly, $\mathcal{T}^*$ is the ground truth trajectory.

4.1. TbD-3D Dataset

We created a new annotated dataset containing fast moving objects. All previous datasets with FMOs, such as FMO dataset [15] and TbD dataset [5], included only objects moving in a 2D plane parallel to the camera plane and their appearance was close to static. Ground truth 2D object location was provided, but no angular velocity.

The introduced dataset is the first dataset with non-negligible 3D object motion and with changing appearance of non-uniform fast moving objects. Objects are from a set of three balls with complex texture. The dataset is called TbD-3D and it contains nine sequences with annotated object location, pose, and size from a high-speed camera. In contrast to previous datasets, the perceived size of objects in TbD-3D dataset varies throughout the whole sequence due to depth of the scene, as shown in Figure 3.

Videos were recorded in raw format using a high-speed camera at 240 fps with exposure time 1/240s (so called 360° shutter angle – negligible lag between two frames). The dataset sequences were generated by averaging 2, 4 and 8 frames, which corresponds to real videos captured at 30, 60, 120 fps, respectively. Indoor-scene movies with 30 fps and $\frac{1}{30}$s shutter speed are not rare. Ground truth annotation was done on the original raw footage at 240 fps. 3D object location (2D position and radius) was annotated manually and 3D object rotation was estimated using the proposed method in Section 3.3; see Section 4.2 for details about ground-truth annotation of the object rotation.

The proposed method is evaluated on the TbD-3D dataset for all three frame-rate settings. Figure 5 shows accuracy of the estimated 6DoF object pose: 3D location

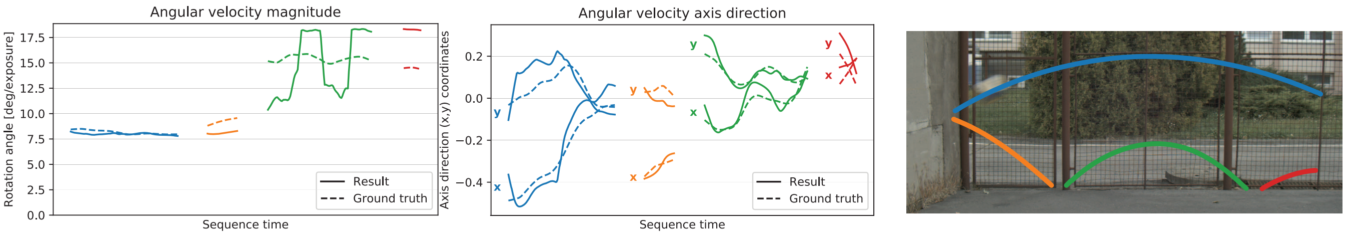


Figure 7: Rotation velocity magnitude and direction in different parts of the sequence (color coded). TbD-3D results – solid lines, ground truth – dashed. Rotation is estimated only between bounces.

error measured by TIoU-3D (left), 2D rotation axis error measured as a mean average deviation from the GT axis in degrees (middle) and mean average error of 1D rotation angle (right). We use TbD [5] and TbD-NC [14] as baselines, which only estimate 2D trajectory. These methods ignore depth changes and assume one object size for the whole sequence. To show the performance of TbD-3D when the input 2D trajectory has no errors, we also provide scores of TbD-3D with oracle (TbD-3D-O) where we use 2D trajectory from the annotated 240-fps videos. TbD-3D-O estimates only additional 4DoF of object pose and compared to TbD-3D it performs better in average. The performance drop of TbD-3D can be thus attributed to errors in 2D trajectories estimated by TbD-NC. Table 1 provides more detailed comparison on every sequence at the lowest frame rate of 30 fps. Estimation of the rotation axis is known to be highly unstable, yet for objects of size 50×50 pixels the reported error of 37 degrees is in average only 8 pixels in the image. Three examples of 3D trajectory reconstruction on sequences 'depth2', 'depthf1' and 'depthb2' are shown in Figure 4 and one example of angular velocity estimation on sequence 'out2' is in Figure 7.

4.2. Rotation Estimation

Calculating ground truth rotation of FMOs, even when the high-speed camera footage is available, is a challenging task. To evaluate the accuracy of the proposed method for rotation estimation (Section 3.3) when applied on high-speed footage, we captured sequences of a ball rolling on the ground along a straight trajectory of known length. The ground truth angular velocity is derived from physical properties of the rolling ball as we know its actual circumference and the distance it traverses. The estimation of rotation axis was 100x less accurate than the estimation of rotation angle. The average error between the GT and estimated rotation axis using the proposed method was 4.052 degrees, while the average error between the GT and estimated rotation angle was only 0.037 degrees, which corresponds to 1.2 % relative error.

A special case appears during contact with another object in the scene. The object is deformed and modeling the object there is out of the scope of this paper. However, we can still detect such deformations as shown in Figure 6.

4.3. Applications

Temporal super-resolution is among the most interesting applications of the proposed method. First, a video free of FMOs is produced by replacing blurred objects with the estimated background. Second, a higher frame rate video is created by linear interpolation. Last, the trajectory is split into the desired number of segments and the object is blended into the sequence with its 6DoF appearance at desired snapshot time-scale, following the image formation model (1). Compared to the previous methods, which use the same appearance for all frames among one low rate trajectory, we synthesize the object at much higher temporal resolution. Videos generated using temporal super-resolution are provided in the supplementary material.

Other applications and future work include rotation estimation for tennis serves, full 3D reconstruction of the blurred object, or handling unknown non-spherical shapes.

5. Conclusion

We proposed a method for estimating up to 6DoF trajectory of fast moving objects which are severely blurred by object motion. The assumption of a non-uniform spherical object is needed, otherwise only a 3D object location is estimated. The proposed TbD-3D method achieves good results on a newly created dataset of non-uniform FMOs with significant changes of appearance and distance to the camera within the sequence or even a frame. Sub-frame appearance estimation enables us to see deformations which last shorter than the exposure duration. We showed a more precise temporal super-resolution compared to the previous methods. The dataset and implementation will be made publicly available.

Acknowledgements. This work was supported by Czech Science Foundation grant GA18-05360S, by the Praemium Academiae of the Czech Academy of Sciences and by Google Focused Research Award.

References

[1] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, Jan. 2011.

[2] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York, NY, USA, 2 edition, 2003.

[3] Michal Jancosek and Tomas Pajdla. Multi-view reconstruction preserving weakly-supported surfaces. In *CVPR 2011*, pages 3121–3128, June 2011.

[4] Meiguang Jin et al. Learning to extract a video sequence from a single motion-blurred image. In *IEEE CVPR*, pages 6334–6342, June 2018.

[5] Jan Kotera, Denys Rozumnyi, Filip Šroubek, and Jiří Matas. Intra-frame Object Tracking by Deblatting. In *International Conference on Computer Vision (ICCV) Workshops*, October 2019.

[6] Jan Kotera and Filip Šroubek. Motion estimation and deblurring of fast moving objects. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2860–2864, Oct 2018.

[7] Matej Kristan, Aleš Leonardis, Jiří Matas, Michael Felsberg, Roman Pflugfelder, Luka Čehovin Zajc, Tomáš Vojíř, Goutam Bhat, Alan Lukežič, Abdelrahman Eldesokey, et al. The sixth visual object tracking VOT2018 challenge results. In Laura Leal-Taixé and Stefan Roth, editors, *ECCV 2018 Workshops*, pages 3–53, Cham, 2019. Springer International Publishing.

[8] Matej Kristan, Jiri Matas, Ales Leonardis, Michael Felsberg, Roman Pflugfelder, Joni-Kristian Kamarainen, Luka Cehovin Zajc, Ondrej Drbohlav, Alan Lukezic, et al. The seventh visual object tracking vot2019 challenge results. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2019.

[9] Bo Li, Junjie Yan, Wei Wu, Zheng Zhu, and Xiaolin Hu. High performance visual tracking with siamese region proposal network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[10] Xu Lingyun, Haibo Luo, Hui Bin, and Zhenxing Chang. Real-time robust tracking for motion blur and fast motion via correlation filters. *Sensors*, 16:1443, 09 2016.

[11] Manolis I. A. Lourakis and Antonis A. Argyros. Sba: A software package for generic sparse bundle adjustment. *ACM Trans. Math. Softw.*, 36(1):2:1–2:30, Mar. 2009.

[12] Alan Lukežič, Tomas Vojíř, Luka C. Zajc, Jiri Matas, and Matej Kristan. Discriminative correlation filter with channel and spatial reliability. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4847–4856, July 2017.

[13] Youngmin Park, Vincent Lepetit, and Woontack Woo. Esmblur: Handling rendering blur in 3d tracking and augmentation. In *IEEE International Symposium on Mixed and Augmented Reality*, pages 163–166, Oct 2009.

[14] Denys Rozumnyi, Jan Kotera, Filip Šroubek, and Jiří Matas. Non-Causal Tracking by Deblatting. In *German Conference on Pattern Recognition (GCPR)*, September 2019.

[15] Denys Rozumnyi, Jan Kotera, Filip Šroubek, Lukas Novotný, and Jiri Matas. The world of fast moving objects. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4838–4846, July 2017.

[16] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-Motion Revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[17] Shuochen Su et al. Deep video deblurring for hand-held cameras. In *IEEE CVPR*, pages 237–246, July 2017.

[18] Jack Valmadre, Luca Bertinetto, Joao Henriques, Andrea Vedaldi, and Philip H. S. Torr. End-to-end representation learning for correlation filter based tracking. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[19] Patrick Wieschollek et al. Learning blind motion deblurring. In *IEEE ICCV*, pages 231–240, Oct 2017.

[20] Tianyang Xu, Zhen-Hua Feng, Xiao-Jun Wu, and Josef Kittler. Learning adaptive discriminative correlation filters via temporal consistency preserving spatial feature selection for robust visual object tracking. *IEEE Transactions on Image Processing*, 28(11):5596–5609, 2019.

[21] Donghun Yeo, Jeany Son, Bohyung Han, and Joon Hee Han. Superpixel-based tracking-by-segmentation using markov chains. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 511–520, July 2017.

[22] Dong Zhang, Omar Javed, and Mubarak Shah. Video object segmentation through spatially accurate and temporally dense extraction of primary object regions. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 628–635, June 2013.