# Inference of Mechanical Properties of Dynamic Objects through Active Perception

Nikolaus Wagner[1][0000−0001−8448−1363]
and Grzegorz Cielniak[2][0000−0002−6299−8465]

University of Lincoln, Brayford Pool, Lincoln LN6 7TS, UK
[1]nwagner@lincoln.ac.uk
[2]gcielniak@lincoln.ac.uk

**Abstract.** Current robotic systems often lack a deeper understanding of their surroundings, even if they are equipped with visual sensors like RGB-D cameras. Knowledge of the mechanical properties of the objects in their immediate surroundings, however, could bring huge benefits to applications such as path planning, obstacle avoidance & removal or estimating object compliance.
In this paper, we present a novel approach to inferring mechanical properties of dynamic objects with the help of active perception and frequency analysis of objects' stimulus responses. We perform FFT on a buffer of image flow maps to identify the spectral signature of objects and from that their eigenfrequency. Combining this with 3D depth information allows us to infer an object's mass without having to weigh it.
We perform experiments on a demonstrator with variable mass and stiffness to test our approach and provide an analysis on the influence of individual properties on the result. By simply applying a controlled amount of force to a system, we were able to infer mechanical properties of systems with an eigenfrequency of around 4.5 Hz in about 2 s. This lab-based feasibility study opens new exciting robotic applications targeting realistic, non-rigid objects such as plants, crops or fabric.

**Keywords:** active perception · image flow · frequency analysis.

## 1 Introduction

When exploring unknown scenes, current state-of-the-art (SOTA) robots typically use RGB & depth (RGB-D) or RGB-only cameras to record, analyse and possibly reconstruct a model of their surroundings. This, however, provides only shape and geometry information but no internal mechanical properties. Human explorers on the other hand would either rely on previous experience or when encountering unknown objects would interact with them, observe the reactions haptically and visually and infer mechanical properties of objects therefrom.

This way of interactively exploring scenes is commonly referred to as ”active perception”. While it offers a lot of benefits for scene understanding, it also poses many, potentially yet unsolved, challenges, which is why most robots currently do not employ it. However, like humans, robots benefit from a deeper understanding of mechanical properties of objects. This knowledge can be incorporated when performing path planning, when interacting with soft materials such as cloth or flexible objects like plants, or generally in order to understand the compliance of nearby objects. In industrial settings, a lot of time and money could be saved by being able to infer the mass of fruits and crops or the stiffness of sheet materials without having to conventionally measure any of those properties.

Active perception has already been a prominent area of research in the past, however contributions are typically very application specific [1][10]. In this paper we present an easy and widely applicable way to infer mechanical properties of objects with help of an RGB-D camera through simple, direct interaction like controlled pushing. The contributions we present include:

- a novel vision-based approach for inferring mechanical properties of dynamic objects through direct interaction;
- an algorithm for 3D-vision-based motion segmentation;
- a feasibility study based on an adjustable spring-mass demonstrator which confirms the applicability of our approach in real world scenarios.

## 2 Previous Work

Prior to the ascent of machine learning algorithms, active perception was being investigated to improve object detection and recognition results [10], but interest in it faded again once neural networks significantly improved performance in these areas. The approach was also studied in relation to the reconstruction of 3D models [1], but in recent years mostly pure learning-based algorithms have dominated this area of interest as well. Nevertheless, as highlighted by recent advancements in object throwing robots [14], the combination of analytical and learning-based approaches, as in learning to estimate a “delta” correcting systematic errors, similar to residual networks [7], can bring significant improvements. A machine vision solution with the capabilities to not only control perception but also action could similarly learn to remove systematic errors.

As highlighted more recently, advances in object detection, object recognition and 3D-reconstruction depend on the capabilities of the robot to control its perception. Bajcsy et al. state that “an agent is an active perceiver if it knows why it wishes to sense, and then chooses what to perceive, and determines how, when and where to achieve that perception” [2]. This implies a situational awareness as well as physical capabilities of interacting with the scene. In the past, this has typically been achieved by changing the viewpoint and actively adjusting the pose of the camera. Novel approaches rather aim for interaction with objects themselves through pushing, for example [12]. Similarly, our algorithm entails applying a controlled amount of force to an object and observing the reaction.

In robotic applications active perception is enabled by kinematic elements of the robot interacting with objects of interest and monitoring the reactions with the vision system. Mavrakis et al. [12] explore this by inferring mechanical properties from pushing objects on a flat surface, however they rely on surface friction for their calculations, restricting applicability of their approach.

Nevertheless, using those properties a novel representation of the world can be created using a voxel map similar to the one presented by Macenski et al [11]. This representation may contain long- as well as short-term dynamic and material specific properties like eigenfrequency [3], maximum displacement under a certain stimulus or the degree of damping present for movable objects. From this, secondary properties can be obtained, like overall compliance of objects in the scene, which is highly beneficial for trajectory planning and scene understanding.

All this could be used to improve path-planning by aiming for obstacle-separation or -removal instead of -avoidance like in [13]. Furthermore, reconstruction of static parts of a scene with a separate reconstruction of movable objects as in [9] could benefit from this approach as it enables a robot to identify movable objects more easily and thus to remove them from static reconstruction.

Overall, the ideas proposed in this paper open up possibilities to obtain deeper insights into mechanical properties of objects without having to rely on conventional measurement methods.

## 3 Methods

This section provides an overview of our algorithm and uses images of the demonstrator we created for our experiments. A more detailed setup explanation with images follows in Sec. 4.1.

We aim to use a minimal amount of hardware additional to the typical equipment of a robot for interacting with and monitoring the behaviour of objects. We assume a way to apply a controlled amount of external force, like a robotic manipulator, and an RGB-D camera as a baseline for our system. Using this, we try to infer the eigenfrequency as well as secondary properties of objects, such as mass or stiffness, by exciting them and monitoring their frequency response.

Basic workflow of our algorithm, illustrated in Fig. 1, starts with excitation of the oscillating object of interest using a controlled amount of force. We record the reaction with an RGB-D camera and calculate image flow for every new image, storing results in a buffer. Once full, we perform pixelwise fast Fourier transform (FFT) on the buffer and extract the dominant frequency for each pixel. Finally, we cluster pixels by similarity in frequency response to achieve segmentation. A detailed explanation of the individual steps is provided hereinafter.

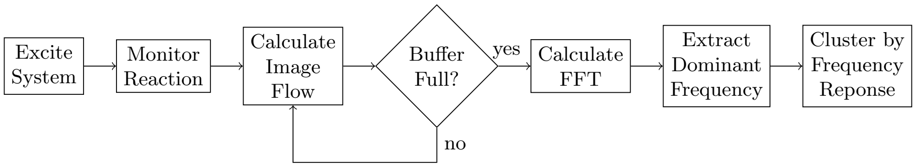


Fig. 1: Flowchart illustrating the basic workflow of our algorithm.

### 3.1 Eigenfrequency

The eigenfrequency $\omega$ is the frequency at which an excited system capable of oscillation moves around its idle point if no external forces are acting upon it. For a simple, undamped spring-mass model the eigenfrequency $\omega$ is given as

$$\omega = \sqrt{\frac{k}{m}}, \tag{1}$$

where $k$ represents the spring constant of the system and $m$ its mass.

If the system is excited by an external force $F$, we can monitor the maximum displacement $x$ from the system's idle position to deduce the spring constant according to Hooke's law given in Eq. 2.

$$k = \frac{F}{x} \tag{2}$$

We assume a known force of excitation $F$, since we typically control the robot exciting the system. Using the RGB-D camera we track objects of interest and determine the maximum deflection $x$. These two parameters allow us to calculate a system's spring constant $k$. By furthermore monitoring the system's frequency response we can obtain its eigenfrequency $\omega$ as described in more detail hereinafter. Knowing $\omega$ and $k$ allows us to infer the mass of objects of interest.

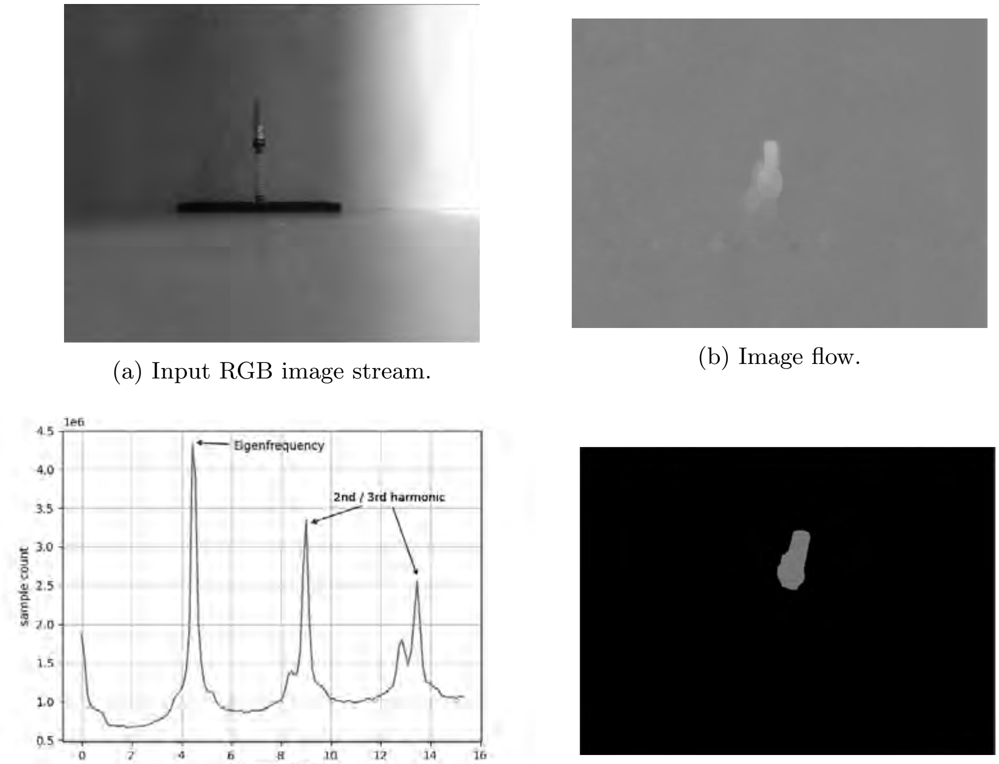
### 3.2 Image Flow

In order to monitor the system's frequency response after excitation, we calculate the image flow for each new image obtained by the camera relative to the last one, using the Farnebäck optical flow method [5]. This method uses polynomial expansion to estimate the motion of objects between two subsequent images, providing an estimate for motion direction as well as magnitude of interest points in the images. An image illustrating this concept, depicting the magnitude of image flow at each individual pixel as a gray scale value, is given in Fig. 2b.

Subsequently, we store each new image flow map in a circular buffer of a predefined size $N$. Once that buffer is full, we extract $w \cdot h$, i. e. the dimensions of the input images, vectors of length $N$ from the buffer, thus one vector containing change in magnitude of image flow over time for each pixel location. We use these vectors for inferring dynamic properties of objects, as explained in Sec. 3.3.

### 3.3 Inferring Dynamic Properties of Objects

By exciting an oscillator system such as described in Sec. 3.1 using a known force, we cause the mass to oscillate at the system's eigenfrequency, allowing us to monitor the process with an RGB-D camera. We use the methods described in Sec. 3.2 to obtain $w \cdot h$ vectors of length $N$ containing the variation in image flow magnitude over time. Next, we perform FFT using FFTW3 [6] on each of these vectors to obtain the spectral signature for each pixel location. We disregard the



(a) Input RGB image stream.

(b) Image flow.

(c) Spectral response.

(d) Clustering by spectral signature.

Fig. 2: Process flow of our algorithm to cluster objects by their spectral response. We use a stream of RGB images (a), calculate the image flow for each new image relative to the previous one (b), store the image flow maps in a buffer and perform pixelwise FFT on it. We then analyse the spectral response for each pixel (c) and cluster pixels by similarity in their spectral signature (d).

phase information obtained from FFT since it is insignificant to our analysis. The resulting vectors containing the signatures thus have the length $\frac{N}{2} - 1$ with the buffer size $N$ as before. By multiplying the indices of the vector with $\frac{FPS}{N}$, wherein the numerator is the frames per second (FPS) of the RGB camera, we receive the frequency in $\frac{1}{s}$, with the value at a certain index corresponding to the number of samples matching each specific frequency. By summing up the individual spectral signatures we receive the spectral signature for the entire image. This could look like the example provided in Fig. 2c.

In this image we see three peaks with the highest peak at roughly 4.5 Hz being the eigenfrequency of the monitored system and two subsequent peaks at 9 Hz and 13.5 Hz, being the $2^{\text{nd}}$ and $3^{\text{rd}}$ harmonic. By extracting the index of maximum value present in the result of the sum of FFTs, we obtain the eigenfrequency at 4.5 Hz by calculating it from the index as described above. The presence of the harmonics as additional peaks can have various reasons, one of them being the non-linearity of the spring-mass-system [4], but deeper investigation is required.

Using the buffer once more, we can extract the maximum value of image flow $V_{max}$ at the center of each cluster. Interpreting this value as object speed allows us to calculate an estimate of the maximum object displacement $x_{max}$:

$$x_{max} \approx \frac{V_{max}}{4 \cdot \hat{\omega}} \tag{3}$$

In this equation, we use $\hat{\omega}$ for the previously calculated value of the eigenfrequency, furthermore we divide by 4 since maximum deflection happens at a quarter of one full oscillation. This approach, chosen for its simplicity, yields only an estimation, e.g. because only the 2D components of movement parallel to the image plane are used.

Knowing the excitation force $F$ and having obtained the maximum displacement $x_{max}$ as well as the eigenfrequency $\omega$, we deduce the spring constant $k$ of the oscillator system as per Eq. 2 and from this the mass $m$ using Eq. 1. This allows us to deduce the mass of objects without actually weighing them.

We can then furthermore use the vectors containing the results of the FFT for each individual pixel to extract the maximum frequency present for each pixel and segment the image by frequency values thus obtained. An example of a segmentation map showing the clusters obtained for a video of our demonstrator, using only pixels with the single dominant (eigen-)frequency, is given in Fig. 2d.

## 4 Results & Analysis

In this section, we describe and evaluate the experiments performed, as well as the results obtained and the influence of system parameters on the outcome.

### 4.1 Experiment Scenario

We evaluate our approach and the system parameters on a demonstrator consisting of an exchangeable spring and a variable mass (see Fig. 3).

### 4.2 Mass Estimation

We experiment with a varying amount of weights and springs, comparing the measured eigenfrequency with the mass of the oscillator. Using these parameters to calculate a value for the spring constant $k = \omega^2 \cdot m$ according to Eq. 1 and comparing them allows us to conclude how well our estimation of the eigenfrequency actually performed. This is inverse to the intended mode of use in practice, but it allows us to evaluate performance in a lab setting. We use a buffer size $N$ of 256, requiring about 8 s of measurement at 30 FPS and excite

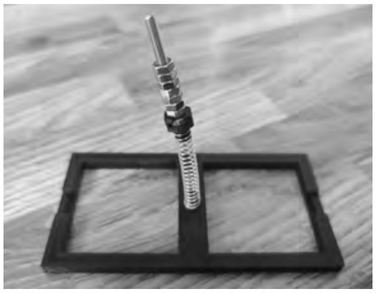


Fig. 3: Demonstrator consisting of a 3D-printed frame, an exchangeable steel spring and a variable amount of hex nuts for varying the mass.

| Spring # | Mass $m$ | Eigenfrequency $\omega$ | Spring constant $k$ | Mean $k$ | Deviation from mean $k$ |
|---|---|---|---|---|---|
| | [g] | [Hz] | [N/m] | [N/m] | [%] |
| 1 | 2.12 | 4.5 | 0.043 | 0.041 | 4.9 |
| | 3.13 | 3.8 | 0.045 | | 9.8 |
| | 4.04 | 3.0 | 0.036 | | 12.2 |
| 2 | 2.12 | 10.0 | 0.212 | 0.189 | 12.2 |
| | 3.13 | 7.9 | 0.195 | | 3.2 |
| | 4.04 | 6.3 | 0.160 | | 15.3 |

Table 1: Eigenfrequencies obtained for varying masses and springs using the demonstrator from Fig. 3.

the system manually. Since we know the weights of the system a priori, knowledge of excitation force is not necessary in this case. The results are shown in Tbl. 1.

We can see the calculated value of the spring constant has a maximum deviation from the mean of about ±12% and ±15% respectively. We can therefore assume the estimation of the eigenfrequency worked reasonably well, given that it is a very rudimentary setup with many possible error sources. Furthermore, an error of 15% in the calculated spring constant corresponds to an error of only $\sqrt{15\%}$, i. e. roughly 3.9%, in the deduced eigenfrequency, since $\omega = \sqrt{\frac{k}{m}}$.

### 4.3 Parameter Analysis

The default buffer size used in our experiments was chosen as $N = 256$, which limits the applicability of our approach since at a frame rate of 30 FPS, as provided by many industrial cameras, we would need to record roughly 8.5 s long samples to perform a single analysis. We therefore explore the influence a varying buffer size has on the quality of the results. This is illustrated in Fig. 4.

We can see the quality of the results directly correlates with buffer size, i. e. smaller buffer size leads to poorer results. For the sample we investigated with an eigenfrequency of about 4.5 Hz a minimum of 64 samples is necessary to achieve

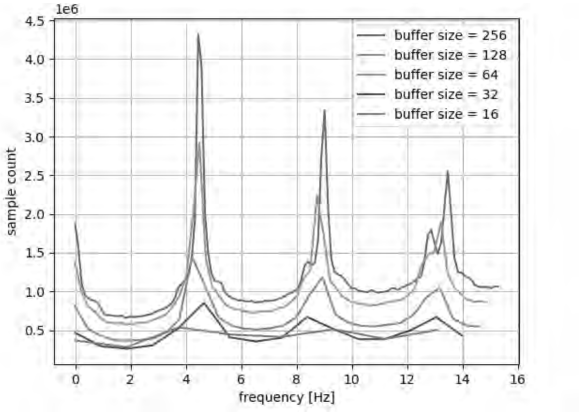


Fig. 4: Influence of the buffer size on the results of the FFT.

usable results. However, a lower buffer size also leads to a larger granularity of the frequency results causing more inaccurate results as well. This, as well as the largest and smallest frequency of interest, needs to be taken into consideration when choosing a buffer size. Furthermore, as mentioned above, a larger buffer size corresponds to a longer period of recording necessary for each sample. This means that, at a frame rate of 30 FPS and assuming the smallest buffer size $N = 64$, the time required to collect sufficient data for a single analysis is still roughly 2 s.

Next, we analyse the change in spectral signature during a period of ring-down after initial excitation. We collect samples for 3 s, 6 s and 9 s after excitation and compare the results to samples collected directly after excitation. This is shown in Fig. 5.

We can see a correlation between the absolute height of the peaks and time passed since excitation, i. e. the maximum amplitude of oscillation which decreases over time. However, further research is needed to establish the exact relationship between amplitude and results of the FFT.

## 5 Conclusions and Future Work

As we have shown in this paper, inference of dynamic properties from active perception and spectral analysis is an interesting and feasible approach, offering many benefits like allowing us to obtain the weight of an object without having to weigh it. We can obtain the spectral signature of objects, cluster images by eigenfrequency of objects and monitor the ring-down of an oscillator system. Nevertheless, many improvements are conceivable.

Using only RGB-based optical flow algorithms, an issue is shadows of objects being clustered in the same category as the actual object, since they move at the same frequency. This can be overcome by using depth based 3D optical flow as described in [8] and [15]. Furthermore, we have presented the spectral

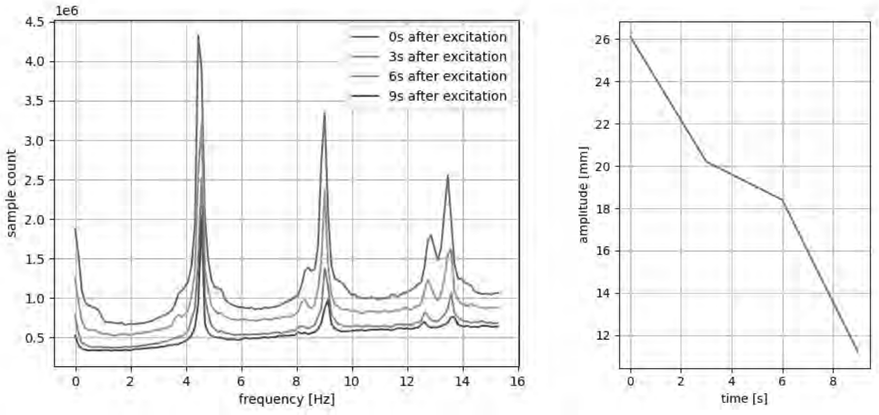


Fig. 5: Spectral signature and maximum amplitude of oscillation in relation to the time passed since initial excitation of the system.

signature containing the eigenfrequency as well as the 2nd and 3rd harmonic in Fig. 2c, however we only consider the eigenfrequency itself for clustering, sometimes leading to poorer results since a lot of valid pixels are filtered out. By incorporating pixels with harmonics as dominant frequency, which are likely due to non-linearities present in the system, we could account for the 2D-mapping of a 3D-oscillator system. Currently, we consider only 2D-oscillator systems, so our approach could fail for objects not oscillating parallel to the image plane. In future work, 3D-oscillations should also be considered, for example by mapping them to a 2D-plane so they can be estimated by a 2D-system. An additional limitation is the minimum recording time needed to perform a single analysis which is limited by buffer size and frame rate provided by the camera. A possible solution to this problem could be the use of a camera offering more FPS.

Further improvements to the system could include a different mode of excitation, e. g. by exciting objects with a stream of air to avoid direct contact. An analysis of the damping coefficient, deductible from the ring-down analyses shown in Fig. 5, can be performed to obtain further system understanding. Finally, creating a similarity score and transferring learned information to similar objects in the scene would allow for obtaining the properties of many objects by interacting with only one. This could subsequently be used to create a full 3D-compliance map of the scene incorporating all dynamic properties obtained.

Concluding, we have demonstrated in this paper the feasibility of using active perception to infer mechanical properties of dynamic objects. This requires minimal contact with the object and yields promising initial results. We have shown the possibility of inferring eigenfrequency, spring constant and mass of a system using nothing but a known force for excitation of the system and an RGB-D camera. Using these parameters, we were able to segment image pixels by similar mechanic properties. Many approaches for future work have been suggested, and there is large potential for further developments in this area.

## References

1. Aleotti, J., Lodi Rizzini, D., Caselli, S.: Perception and grasping of object parts from active robot exploration. Journal of Intelligent & Robotic Systems **76** (12 2014). https://doi.org/10.1007/s10846-014-0045-6
2. Bajcsy, R., Aloimonos, Y., Tsotsos, J.: Revisiting active perception. Autonomous Robots **42** (02 2018). https://doi.org/10.1007/s10514-017-9615-3
3. Chen, J.G., Wadhwa, N., Cha, Y.J., Durand, F., Freeman, W.T., Buyukozturk, O.: Modal identification of simple structures with high-speed video using motion magnification. Journal of Sound and Vibration **345**, 58–71 (2015). https://doi.org/https://doi.org/10.1016/j.jsv.2015.01.024, `https://www.sciencedirect.com/science/article/pii/S0022460X1500070X`
4. Chillara, V.K., Lissenden, C.: Towards a micro-mechanics based understanding of ultrasonic higher harmonic generation. In: Proceedings of SPIE - The International Society for Optical Engineering. vol. 9438 (03 2015). https://doi.org/10.1117/12.2179894
5. Farnebäck, G.: Two-frame motion estimation based on polynomial expansion. In: Image analysis. vol. 2749, pp. 363–370 (06 2003). https://doi.org/10.1007/3-540-45103-X_50
6. Frigo, M., Johnson, S.: The Design and Implementation of FFTW3. Proceedings of the IEEE **93**(2), 216 –231 (Feb 2005). https://doi.org/10.1109/JPROC.2004.840301
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015)
8. Hornácek, M., Fitzgibbon, A., Rother, C.: Sphereflow: 6 dof scene flow from rgb-d pairs. In: 2014 IEEE Conference on Computer Vision and Pattern Recognition. pp. 3526–3533 (2014). https://doi.org/10.1109/CVPR.2014.451
9. Jiang, C., Paudel, D., Fougerolle, Y., Fofi, D., Demonceaux, C.: Static-map and dynamic object reconstruction in outdoor scenes using 3-d motion segmentation. IEEE Robotics and Automation Letters **1**, 1–1 (01 2016). https://doi.org/10.1109/LRA.2016.2517207
10. Le, Q.V., Saxena, A., Ng, A.Y.: Active perception: Interactive manipulation for improving object detection (2010)
11. Macenski, S., Tsai, D., Feinberg, M.: Spatio-temporal voxel layer: A view on robot perception for the dynamic world. International Journal of Advanced Robotic Systems **17**, 172988142091053 (03 2020). https://doi.org/10.1177/1729881420910530
12. Mavrakis, N., Ghalamzan E., A.M., Stolkin, R.: Estimating an object's inertial parameters by robotic pushing: A data-driven approach. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 9537–9544 (2020). https://doi.org/10.1109/IROS45743.2020.9341112
13. Xiong, Y., Ge, Y., From, P.J.: Push and drag: An active obstacle separation method for fruit harvesting robots (2020)
14. Zeng, A., Song, S., Lee, J., Rodriguez, A., Funkhouser, T.: Tossingbot: Learning to throw arbitrary objects with residual physics (2020)
15. Zhang, T., Zhang, H., Li, Y., Nakamura, Y., Zhang, L.: Flowfusion: Dynamic dense rgb-d slam based on optical flow (2020)