GenFlow: Generalizable Recurrent Flow for 6D Pose Refinement of Novel Objects

Sungphill Moon* Hyeontae Son* Dongcheol Hur Sangwook Kim
NAVER LABS
{sungphill.moon, son.ht, dongcheol.hur, o.s.w.a.l.k}@naverlabs.com

Abstract

Despite the progress of learning-based methods for 6D object pose estimation, the trade-off between accuracy and scalability for novel objects still exists. Specifically, previous methods for novel objects do not make good use of the target object's 3D shape information since they focus on generalization by processing the shape indirectly, making them less effective. We present GenFlow, an approach that enables both accuracy and generalization to novel objects with the guidance of the target object's shape. Our method predicts optical flow between the rendered image and the observed image and refines the 6D pose iteratively. It boosts the performance by a constraint of the 3D shape and the generalizable geometric knowledge learned from an end-to-end differentiable system. We further improve our model by designing a cascade network architecture to exploit the multi-scale correlations and coarse-to-fine refinement. GenFlow ranked first on the unseen object pose estimation benchmarks in both the RGB and RGB-D cases. It also achieves performance competitive with existing state-of-the-art methods for the seen object pose estimation without any fine-tuning.

1. Introduction

Estimating accurate 6D object pose from an RGB or RGB-D image is crucial for robotic tasks and augmented reality applications. In the standard setup of 6D object pose estimation, 3D models of target objects are given and available in training and testing. Most state-of-the-art algorithms are learning-based and require training the model per object or dataset containing few objects. However, these methods are difficult to handle novel objects since they memorize the 3D shapes of the targets and suffer from the training cost to support them.

Recent works [5, 37, 51, 54] have proposed generalizable models that can estimate the 6D pose of novel objects, *i.e.*, objects unseen during training. Specifically, OSOP [54] and MegaPose [37] show remarkable performance with the generalizable pose refinement. Their refinement methods have the scalability to novel objects since the networks learn to compare the rendered image and the observed image without memorizing 3D shapes. However, their refiners avail themselves of 3D shape information indirectly. OSOP refiner is trained by the surrogate 2D-2D matching loss, which makes it suboptimal for 6D pose estimation. MegaPose refiner regresses the relative 6D pose between the rendered image and the input, but the regression makes no use of the projective geometry.

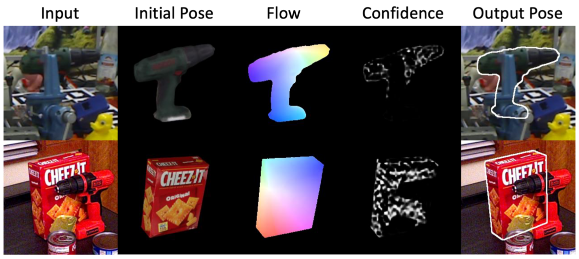


Figure 1. Examples of 6D pose estimation of novel objects. Our method estimates correspondences between the input image and rendered image, and 6D object pose.

Our research question is *Is there a better design for generalizable pose refinement that takes advantage of shape information?*. Recently, 2D optical flow-based methods [20, 31] show noticeable performance on 6D pose estimation. These methods estimate dense 2D-2D correspondences between the rendered image and the observed image. The estimated correspondences are lifted to 2D-3D correspondences on the 3D shape of the target object, then the 6D pose is recovered using a PnP solver [31] or pose regressor networks [20]. The optical flow-based pose refinement provides a clue for the generalization due to its robustness to the domain shift [31] and shows the potential to utilize 3D information in the model [20].

We propose GenFlow, the optical flow-based iterative

*These authors contributed equally.

refinement for the generalizable object pose estimation guided by the 3D shape information of target objects. Given an image and initial 6D pose of an object in the image, GenFlow estimates 2D dense correspondences between the rendered image and the input image and refined 6D pose through a PnP layer. This series of processes is performed iteratively, so it can improve accuracy and robustness to outliers [43].

We adopt the shape-constraint recurrent flow framework inspired by [20] for our model to use 3D information. This framework utilizes the shape information implicitly to enjoy both generalization and performance. Its end-to-end differentiable system learns to estimate the optical flow, confidence, and object pose. These estimations are updated iteratively complying with the 3D shape. We further introduce a cascade network design to make use of the multi-scale correlation volumes. It takes advantage of the coarse-to-fine refinement and helps the model estimate a more accurate pose.

We validate our method on various datasets provided in the BOP challenge [28]. Our model is designed to estimate 6D pose for the RGB input but is also applicable to the RGB-D case with the RANSAC-Kabsch algorithm [18, 34]. For both RGB and RGB-D cases, our method achieves the overall best performance for the unseen object pose estimation task using the default detection [50] provided in the BOP challenge 2023. It is also notable that our method achieves competitive performance with the state-of-the-art methods for seen objects when used with better detection results.

2. Related Work

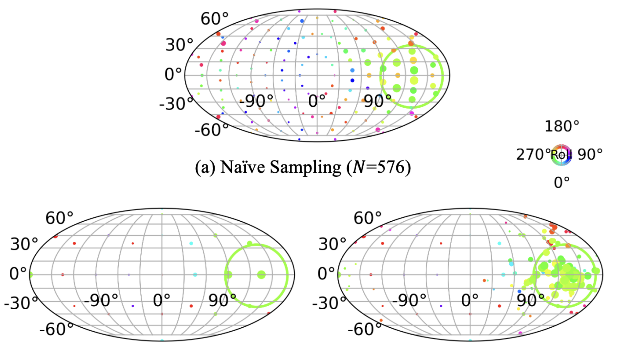
Iterative Refinement DeepIM [41] pioneered the iterative render-and-compare strategy for 6D pose estimation. It predicts the relative pose that better matches a rendered image against the observed image. By iterative refinement, the rendered and observed images become more similar. CosyPose [38] improved the idea by introducing better parameterization for rotation [69], network architecture [60], loss design [55, 62] and data augmentation. These iterative methods show accurate results and the possibility of being applied to the unseen objects [41]. In contrast to the previous works, CIR [43] introduces a novel differentiable PnP layer to utilize the projective geometry with deep learning. Unlike CIR, which assumes the RGB-D input, ours focuses on the RGB input. For the RGB input case, CIR obtains the rendered depth as the substitution of input depth for every flow update, so the inference suffers from time inefficiency. On the other hand, our refinement method requires fewer renderings.

Optical Flow Estimation The classical approaches [2, 6, 29] dealt with optical flow estimation as a hand-crafted optimization problem in pursuit of the balance of brightness consistency and motion plausibility. Deep learning methods [12, 32, 33, 57, 66] emerged as promising alternatives of the classical methods, achieving considerable performance with faster inference times. Recently, RAFT [61] introduced a recurrent deep network architecture which has shown significant improvement over the previous methods regarding accuracy, efficiency, and generalization. PFA [31] applied the optical flow estimation to 6D pose refinement for data-limited cases. Since learning to estimate dense optical flow helps the networks to focus on the lower-level information, it could be robust to domain shift. Following PFA, SCFlow [20] further improved the performance by introducing an end-to-end trainable system using a pose regressor and shape-constraint lookup. Our method capitalizes on a differentiable PnP layer to the framework in order to make use of projective geometry.

Unseen Object Pose Estimation Category-level 6D pose estimation [7, 40, 48, 62] tackled the case of unseen objects of which category is given in advance. Since these methods depend on the prior of the 3D shapes belonging to specific category, they struggle with handling the objects of unknown categories. Recently, several works proposed class-agnostic generalizable pose estimators that can handle unseen objects. A few works attempted the CAD model-free pose estimation [23, 24, 45, 53, 58] to cover everyday objects, but the results were only evaluated on datasets without occlusion and cluttered scenes. [49, 59, 64, 65] focused on estimating the orientation of novel objects by comparing the deep image features of the rendered image and the observed image. ZePHyR [51] proposed a generalizable scoring function combined with hand-crafted features [15, 47]. However, it uses the features selectively according to the richness of texture in the datasets. ZeroPose [5] estimates 6D pose by hierarchical feature matching which requires the depth input by its nature and pose refinement for better performance. Like ours, OSOP [54] and MegaPose [37] proposed CAD-based pose estimation methods that can process RGB and RGB-D inputs. Specifically, MegaPose achieved state-of-the-art performance by its iterative pose refiner. The pose refiner leverages the multi-view rendered images containing surface normals for utilizing rich information about the novel object's shape, anchor point, and coordinate system. Our method refines the pose with the guidance of the target object's shape without rendering multi-view images per refinement step.

3. Method

This section describes our methodology for 6D pose estimation of novel objects. Given an RGB image and 3D models of target objects, which are unseen during training, our goal is to estimate the 6D poses of the objects with respect to the camera. Our pose estimation pipeline consists of 3 stages: object detection, coarse pose estimation, and



(a) Naïve Sampling (N=576)

(b) Sampling using GMM (M=104)

Figure 2. Visualization of the scores predicted by the coarse model. Each dot denotes the sampled rotation, and its size represents the relative score of the coarse model for the corresponding rotation. The green circles are boundaries of the area where the refiner is interested. Figures show that the high-scoring poses are clustered in the vicinity of the green circles.

pose refinement. Note that the detection of novel objects is out of the scope of this paper. We first provide an overview of the whole process and then focus on the pose refinement method, which is the main contribution of this work.

3.1. Preliminaries

Given a calibrated RGB image $\mathcal{I}$ and a set of 3D models of the target objects $\{\mathcal{M}_i\}$, we estimate the 6D poses of the visible objects in the image. A 6D pose is defined by a matrix $\mathbf{P}$ representing the rigid transformation from the object space to the camera space. Note that we assume the input image is calibrated, *i.e.*, the intrinsic camera matrix $\mathbf{K}$ is known.

2D Object Detection Given an RGB image, the object detection method produces a list of object detections consisting of 2D bounding boxes and associated labels. For the unseen object detection, we apply the CNOS [50] detection leveraging the foundation models [36, 52]. To show the extensive results of our work, we also apply the YOLOX detector [19] from GDRNPP [44] which is used for seen object detection. For each detection, the image $\mathcal{I}$ is cropped and further resized to the specific resolution in the following steps. The intrinsic parameters are adjusted according to the resized resolution. For convenience, the following explanation focuses on the 6D pose estimation of a single detected object of which the 3D model is $\mathcal{M}$.

Coarse Pose Estimation Given object detection, the coarse pose estimator predicts rough initial poses. We used the classification-based method proposed in [37] for better generalization. Given an input image and an arbitrary pose $\mathbf{P}$ for the 3D model $\mathcal{M}$, the coarse model classifies whether or not the pose $\mathbf{P}$ could be improved to the pose in proximity to the ground-truth by the refiner. It uses a pre-defined set of 3D rotations of which the cardinality is N to generate the various arbitrary pose hypotheses. For each rotation, a rough 3D translation is estimated for the points of the object's 3D model to be approximately inside the 2D bounding box. We extract scores of the pose hypotheses from the model and select the top-n scoring poses. Such a classification-based estimation can handle novel objects, but it suffers from the cost of rendering a large number of images for the pre-defined rotations. Inspired by the property of high-scoring rotations to be sparse and clustered as shown in Fig 2, we introduce sampling using a Gaussian mixture model (GMM) [1] for efficiency. First, we choose the top k-scoring samples $\{R_i = (\psi_i, \theta_i, \phi_i)\}_{i=1}^{k}$ from M pre-defined rotations (Fig 2b left). Then we create a density distribution $p(X)$ for the Euler angle X using a GMM which satisfies that

$$p(X) = \sum_{i=1}^{k} \frac{1}{k} \mathcal{N}(X|\mu_i, \Sigma), \mu_i = R_i \tag{1}$$

Finally, we sample additional M rotations from the density distribution $p(X)$ (Fig 2b right) and choose the best-scoring n hypotheses among the $2M$ samples. This method renders $2M < N$ images while achieving better performance. We used $M = 104$ and $k = 16$ for our experiments.

Pose Refinement Given a coarse pose estimate, we refine the pose for the rendered image to match against the input image crop. Starting from the coarse estimate $\mathbf{P}_0$ as an initial pose, we iteratively refine the pose using our refiner. We denote the k^{th} 6D pose estimate of the refiner as $\mathbf{P}_k$. For the k^{th} iteration, we obtain an image $\mathcal{I}_{r,k}$ by rendering the 3D model $\mathcal{M}$ for the pose $\mathbf{P}_{k-1}$. Our refiner then estimates the 2D optical flow $\mathbf{F}_k$ between the image $\mathcal{I}_{r,k}$ and the input image crop and the confidence weights $\mathbf{W}_k$ which are used to solve the pose $\mathbf{P}_k$. A detailed description is in the following section.

3.2. GenFlow Refinement

3.2.1 Overview

We describe the specific operation of GenFlow refiner to estimate the refined pose $\mathbf{P}_k$ from the previous result $\mathbf{P}_{k-1}$ but hereafter, we omit the k notation if possible for convenience. First, we obtain a synthetic image by rendering the 3D model $\mathcal{M}$ for the pose $\mathbf{P}$ and adjusted intrinsic parameters. Specifically, the results of the rendering consists of the RGB image, depth image and binary rendered mask, denoted by $\mathcal{I}_r \in R^{H \times W \times 3}$, $\mathcal{D}_r \in R^{H \times W}$, and $\mathcal{M}_r \in \{0, 1\}^{H \times W}$ respectively. Then, we extract features from both images and refine the pose with the GenFlow module.

Feature Extraction We extract the image features from the input image crop and the rendered image $\mathcal{I}_r$ with the same feature encoder and the context features from

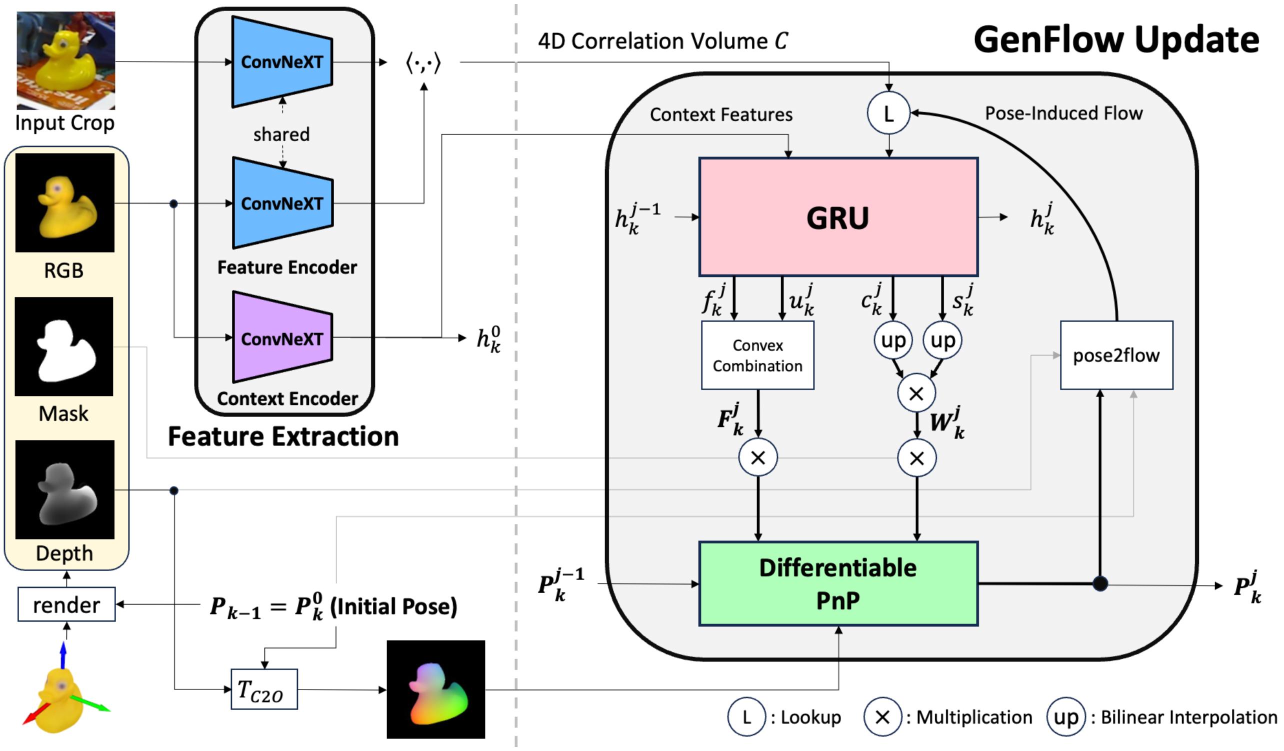


Figure 3. Overview of GenFlow refinement. We visualize the process of feature extraction and GenFlow update in the k^{th} refinement. T_{C2O} refers to a transformation that maps the coordinates from camera space to object space.

the $\mathcal{I}_r$. The feature maps from the i^{th} layer of the feature encoder are used to construct the correlation volumes $\mathbf{C}_i \in R^{H_i \times W_i \times H_i \times W_i}$ by the dot product.

GenFlow Module For each correlation volume and context feature map, we construct a module named "GenFlow module". A GenFlow module consists of a convolution-based Gated Recurrent Unit (GRU) [9], a differentiable PnP solver, and correlation lookup based on the pose-induced flow as shown in the Fig 3. GenFlow module iteratively updates the optical flow, confidence weights, and 6D pose. For the j^{th} update, the output of the module is the refined 6D pose $\mathbf{P}^j$. Note that we use subscripts for the outer updates and superscripts for the inner updates, *i.e.*, GenFlow updates. The detailed illustration of updating the 6D pose with a GenFlow module for i^{th} correlation volume is as follows.

3.2.2 GenFlow Update

GRU Update For the j^{th} update, the GRU of the GenFlow module consumes the previous hidden state h^{j-1}, correlation features r^{j-1} and context features. The output of the GRU $\{\mathbf{h}^j, \mathbf{f}^j, \mathbf{u}^j, \mathbf{c}^j, \mathbf{s}^j\}$ includes the updated hidden state h^j, optical flow $\mathbf{f}^j$, flow upsampling mask $\mathbf{u}^j$, certainty $\mathbf{c}^j$ and pose sensitivity $\mathbf{s}^j$. The flow $\mathbf{f}^j \in R^{H_i \times W_i \times 2}$ is upsampled to $\mathbf{F}^j$ to match the resolution of previous $(i-1)^{th}$ feature map using the mask $\mathbf{u}^j$. We apply the convex combination method of 3x3 neighbors for the optical flow proposed in [61]. The certainty $\mathbf{c}^j$ and pose sensitivity $\mathbf{s}^j$ are upsampled by bilinear interpolation, and then we multiply them to obtain the confidence weights $\mathbf{W}^j$. The head networks for certainty and pose sensitivity are supervised by different losses, respectively. This explicit factorization of the confidence weights helps to be robust to occlusion with the help of certainty estimation as shown in Fig 4. The upsampled flow $\mathbf{F}^j$ and confidence weights $\mathbf{W}^j$ are used to establish weighted 2D-3D correspondences.

Pose Update As we know the depth of synthetic image $\mathcal{D}_r$ for the pose $\mathbf{P}_k^0 = [\mathbf{R}_k^0|\mathbf{t}_k^0]$ and the intrinsic camera matrix $\mathbf{K}$, we can compute the object-space 3D coordinate of each pixel by

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \mathbf{R}_k^{\mathrm{T}}(\mathbf{K}^{-1}\mathcal{D}_r(u,v)\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} - \mathbf{t}_k) \qquad (2)$$

where (u,v) is the 2D coordinate of a pixel and the $\mathcal{D}_r(u,v)$ is depth of the pixel. Through this lifting process, we obtain the 2D-3D correspondences for the flow $\mathbf{F}^j$. We use $\mathbf{W}^j$ for the confidence of correspondences and binary rendered mask $\mathcal{M}_r$ to consider the object region only.

To obtain the 6D pose $\mathbf{P}^j$, we utilize the iterative PnP solver based on the Levenberg-Marquardt (LM) algorithm.

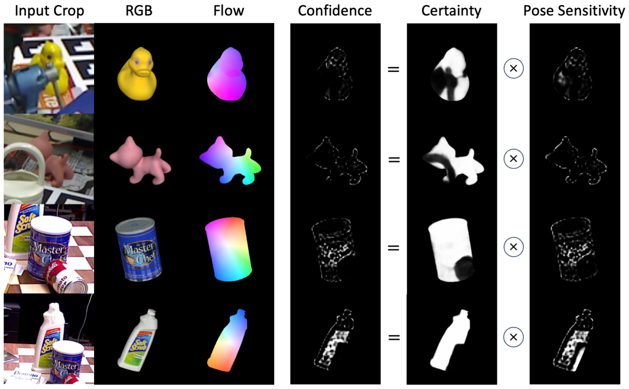


Figure 4. Visualization of the outputs of GRU. Our method factorizes the confidence weights into two terms: certainty and pose sensitivity. Certainty estimation helps to the robustness to occlusion. Pose sensitivity highlights the rich texture regions and extremities of the object.

It optimizes the 6D pose by minimizing the sum of squared weighted reprojection error:

$$\underset{R,t}{\operatorname{argmin}} \frac{1}{2} \sum_{u} \sum_{v} \| \mathbf{W}^j(u,v) \times (\pi(R \begin{pmatrix} x \\ y \\ z \end{pmatrix} + t) - (\begin{pmatrix} u \\ v \end{pmatrix} + \mathbf{F}^j(u,v)))\|^2, \tag{3}$$

where π is the projection function with the intrinsic camera parameters and $(x, y, z)^{\mathrm{T}}$ is the 3D coordinate of the pixel (u, v) computed by the equation 2. Due to its differentiable nature, the PnP layer enables end-to-end training. However, the backpropagation by unrolling the iterative optimization causes numerical instability, making the training process hard. For stable end-to-end training, we introduce the regularization method [4], which backpropagates the gradient through the last step of a differentiable optimization. Specifically, we first obtain a detached pose $\mathbf{P}^*$ through 3 LM steps and update the pose via another iteration of the Gauss-Newton algorithm to obtain the pose $\mathbf{P}^j$. Note that the gradient is not backpropagated through $\mathbf{P}^*$.

Correlation Lookup We compute the pose-induced flow, which is the 2D optical flow between the $\mathcal{I}_r$ and the image with respect to the pose $\mathbf{P}^j$. The pose-induced flow is calculated as the displacement of each pixel in the $\mathcal{I}_r$ by lifting and reprojecting the pixel using the depth $\mathcal{D}_r$. To embed the 3D shape of the target object in the lookup operation [20, 43], we use the pose-induced flow rather than the estimated flow $\mathbf{f}^j$ for indexing the correlation volume $\mathbf{C}$. This way of imposing a shape constraint on the model induces the networks to predict dense matching to comply with the target's 3D shape information. From the lookup operation, we obtain the correlation feature r^j, which is the input of GRU for the next GenFlow update.

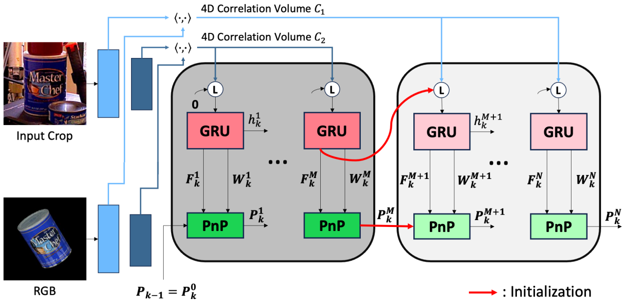


Figure 5. Cascade design of GenFlow modules. Multiple GenFlow modules are attached to each level on the feature pyramid. The last updated flow $\mathbf{F}_k^M$ and 6D pose $\mathbf{P}_k^M$ from the higher-level GenFlow module are used to initialize the flow and 6D pose of the lower-level module. With the cascade architecture, the 6D pose is recovered in a coarse to fine manner.

3.2.3 Cascade Architecture

Inspired by the previous methods that exploit the multi-scale features [8, 42], we design a cascade architecture using multiple GenFlow modules. Given correlation volumes $\{\mathbf{C}_i\}$, we assign a GenFlow module to each volume. Then, we use the 2D optical flow and refined 6D pose of the last inner updates to initialize those of the next module as shown in Fig 5. This cascade architecture takes advantage of progressive refinement, which makes use of the high-level semantics to the low-level fine details. We found that it improves the accuracy of 6D pose rather than using a single GenFlow module.

3.3. Training

Dataset Both the coarse and refiner models are trained using the large dataset provided by MegaPose [37]. It is a large synthetic dataset comprising 2 million RGB-D images generated by BlenderProc [11], which contains diverse 3D models with ground-truth 6D pose annotations. The 3D shapes are collected from ShapeNet [3] and Google Scanned Object (GSO) [14] datasets, and we trained our model on both of the datasets.

Refiner Model Our refiner model is trained similarly to the previous iterative refinement methods [37, 38]. Given an image with the ground-truth 6D pose $\mathbf{P}_{gt}$ for the 3D model $\mathcal{M}$, we perturb the pose to generate an initial pose $\mathbf{P}_{init}$ by applying random noises to the translation and rotation of the pose respectively. Translation noise is sampled from a normal distribution with a standard deviation of (0.01, 0.01, 0.05) centimeters, and rotation noise is sampled from another normal distribution of Euler angles with a standard deviation of 15 degrees in each axis. The refiner model learns to predict the pose $\mathbf{P}_{gt}$ from the perturbed initial pose $\mathbf{P}_{init}$.

Coarse Model Given an input RGB image $\mathcal{I}$ and a pose $\mathbf{P}$, a coarse model is trained to classify whether the refiner could estimate the 6D pose in proximity to the ground truth pose from the initial pose $\mathbf{P}$. We randomly generate the positive and negative pose samples and train the network to classify them using the binary cross entropy loss. For the positive samples, we use the same method used to generate the perturbed initial pose in the refiner model. Negative samples are generated using the method proposed in [37].

3.4. Implementation Details

Coarse Estimation We use 224×224 resolution for the resized input crop and the synthetic image $\mathcal{I}_r$. The coarse model is implemented using the ConvNeXt backbone [46] and fully connected head. We used the ConvNext model pre-trained on ImageNet [10] classification to initialize the networks' weights. For inference, we select top-n hypotheses using our GMM-based sampling strategy with $M = 104$.

GenFlow Refinement First, we obtain a synthetic image $\mathcal{I}_r$ of which resolution is 256×256 and an image crop resized to the same resolution as the synthetic image with adjusted intrinsic parameters. Like the coarse model, we use the pre-trained ConvNeXt for the feature and context encoders. The feature maps from the first two layers of the feature encoder are used to construct correlation volumes $\mathbf{C}_1 \in R^{\frac{H}{4}\times\frac{W}{4}\times\frac{H}{4}\times\frac{W}{4}}$ and $\mathbf{C}_2 \in R^{\frac{H}{8}\times\frac{W}{8}\times\frac{H}{8}\times\frac{W}{8}}$. Our GRU is implemented with a ConvGRU unit [61] and following four CNN-based heads for the 2D optical flow $\mathbf{f}$, flow upsampling mask $\mathbf{u}$, certainty $\mathbf{c}$ and pose sensitivity $\mathbf{s}$. Two GenFlow modules are attached to the correlation volumes for the cascade architecture. They are serially connected so that the estimations from higher-level features are utilized as initial values for the next module.

The optical flow and its upsampling mask produce the upsampled flow, which is supervised by the L1 endpoint error loss with the ground-truth optical flow. We also utilize the disentangled point matching loss for the pose loss, supervising the rotation, 2D object center, and depth of the object individually. Specifically, given an estimated pose $\mathbf{P} = [\mathbf{R}|[\mathbf{t}_x, \mathbf{t}_y, \mathbf{t}_z]^\mathrm{T}]$ and ground truth pose $\mathbf{P}_{gt} = [\mathbf{R}^\star|[\mathbf{t}_x^\star, \mathbf{t}_y^\star, \mathbf{t}_z^\star]^\mathrm{T}]$, the pose loss is computed as follows:

$$\begin{aligned}\mathcal{L}_{pose} &= \mathcal{D}([\mathbf{R}|[\mathbf{t}_x^\star, \mathbf{t}_y^\star, \mathbf{t}_z^\star]^\mathrm{T}], \mathbf{P}_{gt}) \\ &+\mathcal{D}([\mathbf{R}^\star|[\mathbf{t}_x, \mathbf{t}_y, \mathbf{t}_z^\star]^\mathrm{T}], \mathbf{P}_{gt}) \\ &+\mathcal{D}([\mathbf{R}^\star|[\mathbf{t}_x^\star, \mathbf{t}_y^\star, \mathbf{t}_z]^\mathrm{T}], \mathbf{P}_{gt})\end{aligned} \tag{4}$$

where the distance $\mathcal{D}$ between two 6D poses $\mathbf{P}_1$ and $\mathbf{P}_2$ using the 3D points $\mathcal{X}_\mathcal{M}$ on the 3D model $\mathcal{M}$ is defined as

$$\mathcal{D}(\mathbf{P}_1, \mathbf{P}_2) = \frac{1}{|\mathcal{X}_\mathcal{M}|}\sum_{x\in\mathcal{X}_\mathcal{M}} |\mathbf{P}_1 x - \mathbf{P}_2 x| \tag{5}$$

To supervise the certainty $\mathbf{c}$, we introduce the certainty estimation by classifying the depth consistency similarly to [17]. Given 6D poses $\mathbf{P}_{init}$ and $\mathbf{P}_{gt}$ for the 3D model $\mathcal{M}$ and intrinsic camera parameters $\mathbf{K}$, we compute the projected depth $\mathcal{D}_{r\rightarrow t}$ by warping the rendered image to the target input image using the depth $\mathcal{D}_r$. We also obtain the depth map of warped pixels $\mathcal{D}_t$ given the synthetic depth of the target image. Then we train the networks by minimizing the binary cross entropy loss $\mathcal{L}_{cert}$ between $\mathbf{c}$ and $\mathbf{c}_{r\rightarrow t}$ which is a binary mask $\mathbf{c}_{r\rightarrow t} = |\mathcal{D}_{r\rightarrow t} - \mathcal{D}_t| < d_{th}$ where d_{th} is the distance threshold. It is worth noting that we place a stop-gradient on the certainty mask $\mathbf{c}^j$ when computing the pose loss $\mathcal{L}^j_{pose}$ for disentangling the certainty $\mathbf{c}$ and pose sensitivity $\mathbf{s}$.

Given the RGB-D images with annotated ground truth 6D poses, the overall loss is defined as

$$\mathcal{L} = \sum_{j=1}^{N} \gamma^{j-N}(\mathcal{L}^j_{flow} + \alpha\mathcal{L}^j_{cert} + \beta\mathcal{L}^j_{pose}) \tag{6}$$

following the strategy of exponentially increasing weights [61] where we use the weight $\gamma = 0.8$ and the iteration of GenFlow updates $N = 8$ in this work. In specific, the GenFlow module assigned to $\mathbf{C}_2$ outputs $\{\mathbf{P}^j\}_{j=1}^{N/2}$ and the another module outputs $\{\mathbf{P}^j\}_{j=N/2}^{N}$ from $\mathbf{C}_1$.

Multi-Hypotheses Strategy We select n best-scoring hypotheses in the coarse estimation. The selected hypotheses are refined, and we choose the best estimate by evaluating the refined hypotheses using the coarse model again. This strategy requires more inference time for evaluating n iterations of the rendering and refinement process, but it can boost the accuracy by avoiding the risk of the "Winner-Takes-All" strategy.

RGB-D Input If the observed image consists of depth information, we apply the depth refinement after the GenFlow loop for every refinement step. First, we obtain the optical flow $\mathbf{F}$ and upsampled certainty from the results of the last GenFlow update. 3D-3D correspondences are established by the optical flow and the input depth. We filter out the correspondences of certainty lower than a specific threshold to remove the outliers. Then we apply the RANSAC-Kabsch [18, 34] for the depth refinement. Using certainty for filtering out the outliers is more effective than using confidence weights since the high-confidence pixels are sparse and the input depth is prone to be noisy.

		2D Localization		Pose Initialization		Pose Refinement		Average Recall (↑)							
	RGB-D Input	Method	Novel Objects	Method	Novel Objects	Method	Novel Objects	LM-O	T-LESS	TUD-L	IC-BIN	ITODD	HB	YCB-V	MEAN
1	✗	Mask-RCNN [22, 38]	✗	CosyPose [38]	✗	CosyPose [38]	✗	63.3	64.0	68.5	58.3	21.6	65.6	57.4	57.0
2	✗	Mask-RCNN [22, 38]	✗	SurfEmb [21]	✗	BFGS	✗	66.3	73.5	71.5	58.8	41.3	79.1	64.7	65.0
3	✗	YOLOX [19, 44]	✗	ZebraPose [56]	✗	-	✗	**72.9**	81.1	75.6	59.2	**50.4**	**92.1**	**72.9**	**72.0**
4	✗	YOLOX [19, 44]	✗	MegaPose [37]	✓	MegaPose+MH [37]	✓	64.8	78.1	74.1	56.9	42.2	86.3	70.2	67.5
5	✗	YOLOX [19, 44]	✗	Ours	✓	Ours+MH	✓	68.3	**82.8**	**77.8**	**59.6**	50.1	89.7	70.8	71.3
6	✓	YOLOX [19, 44]	✗	WDR-Pose [30]	✗	PFA [31]+Kabsch	✗	**79.2**	**84.9**	**96.3**	**70.6**	52.6	86.7	**89.9**	**80.0**
7	✓	YOLOX [19, 44]	✗	MegaPose [37]	✓	MegaPose+MH [37]+Teaserpp [67]	✓	70.4	71.8	91.6	59.2	55.3	87.2	85.5	74.4
8	✓	YOLOX [19, 44]	✗	Ours	✓	Ours+MH+Kabsch	✓	74.2	78.3	92.8	64.9	**65.2**	**92.0**	88.3	79.4
9	✗	OSOP [54]	✓	OSOP [54]	✓	OSOP+PnP+MH [54]	✓	31.2	-	-	-	-	49.2	33.2	-
10	✗	CNOS-det. [50]	✓	MegaPose [37]	✓	MegaPose+MH [37]	✓	56.0	50.8	68.7	41.9	34.6	70.6	62.0	54.9
11	✗	CNOS-det. [50]	✓	Ours	✓	Ours+MH	✓	**57.5**	**53.0**	**69.1**	**45.6**	**40.8**	**74.5**	**63.9**	**57.8**
12	✓	OSOP [54]	✓	OSOP [54]	✓	OSOP+Kabsch+MH [54]+ICP	✓	48.2	-	-	-	-	60.5	57.2	-
13	✓	CNOS-seg. [50]	✓	ZeroPose [5]	✓	MegaPose+MH [37]	✓	53.8	40.0	83.5	39.2	52.1	65.3	65.3	57.0
14	✓	CNOS-det. [50]	✓	MegaPose [37]	✓	MegaPose+MH [37]+Teaserpp [67]	✓	62.6	48.7	85.1	46.7	46.8	73.0	76.4	62.8
15	✓	CNOS-det. [50]	✓	Ours	✓	Ours+Kabsch+MH	✓	**62.9**	**51.7**	**85.8**	**53.3**	**55.9**	**78.2**	**82.5**	**67.2**

Table 1. 6D pose estimation results on the BOP challenge datasets. We report the Average Recall (AR) scores across the datasets for various methods. The higher score the better. The best results among the comparable methods are in bold. We denote the multi-hypotheses strategy as MH for simplicity.

4. Experiments

4.1. Datasets and Evaluation Metrics

Datasets We evaluate our method on seven datasets of the BOP challenge [28]: LM-O [25], T-LESS [26], TUD-L [27], IC-BIN [13], ITODD [16], HomebrewdDB [35] and YCB-V [63]. These datasets contain cluttered real-world scenes captured from different lighting and camera sensors. Each scene consists of multiple objects including occlusions between them. The objects vary in terms of texture richness, symmetry, and usage.

Evaluation Metrics Following the evaluation methodology of BOP challenge [28], we report the average recall (AR) considering three pose-error functions: Visible Surface Discrepancy (VSD), Maximum Symmetry-Aware Surface Distance (MSSD), and Maximum Symmetry-Aware Projection Distance (MSPD). VSD measures the misalignment of the visible part by computing the ratio of pixels in which the difference between the estimated distance to the camera center and ground truth is under a specific tolerance. In consideration of the global symmetry information, MSSD computes the maximum distance between the estimated camera-space coordinates and corresponding ground truth. In contrast, MSPD computes the maximum distance between the estimated image-space coordinates and the corresponding ground truth. Concerning each pose-error function, the AR is obtained by averaging recall calculated for multiple settings of the correctness thresholds (and misalignment tolerances for VSD).

4.2. BOP Benchmark Results

Table 1 shows the results of our method on the BOP datasets. For a fair comparison, all reported pose initialization and pose refinement methods are trained with synthetic data containing images rendered with BlenderProc [11]. We reported the AR scores of MegaPose and ours by setting the number of hypotheses n as 10 for the multi-hypotheses strategy and the number of outer updates as 5. For ours, the number of GenFlow updates per single outer update is set to 8.

Method	# of renderings (↓)	Average Recall (↑)					
		LM-O	T-LESS	TUD-L	IC-BIN	YCB-V	MEAN
Naïve	576	22.5	25.6	30.6	17.5	16.4	22.5
Ours	144	28.8	35.5	33.8	29.7	28.9	31.3
	208	**32.9**	**40.4**	**40.0**	**31.1**	**34.7**	**35.8**

Table 2. Ablation study of the hypotheses generation strategy for the coarse estimation. The AR scores are reported for the best pose hypothesis from the candidates from different sampling methods.

Unseen Objects For both RGB input (row 9-11) and RGB-D input (row 12-15), the results show that our approach outperforms all other methods for novel objects.

Seen Objects From row 1 to 8, the pose estimation results on the 2D detection trained for target datasets are reported. Our method still outperforms MegaPose for the same detection results. It is noticeable that our method provides competitive results to the state-of-the-art methods, though it is not the best, even without fine-tuning on the target datasets.

4.3. Ablation Study

We conduct ablation studies to validate the efficacy of our method. The ablation results are reported for the RGB inputs we mainly focus on. All experiments were performed on 5 BOP datasets: LM-O, T-LESS, TUD-L, IC-BIN, and YCB-V. We used the 2D detection results of YOLOX [19,

	Method	Training	Average Recall (↑)					
			LM-O	T-LESS	TUD-L	IC-BIN	YCB-V	MEAN
1	No shape-constraint + RANSAC-PnP [39]	$\mathcal{L}_{flow}$	61.7	77.6	72.4	54.2	65.7	66.3
2	Shape-constraint	$\mathcal{L}_{flow}, \mathcal{L}_{pose}$	64.9	81.8	75.1	57.0	70.1	69.8
3	Shape-constraint + Cascade	$\mathcal{L}_{flow}, \mathcal{L}_{pose}$	65.7	82.3	75.0	58.9	69.9	70.4
4	Shape-constraint + Confidence factorization	$\mathcal{L}_{flow}, \mathcal{L}_{cert}, \mathcal{L}_{pose}$	64.5	81.8	75.3	57.3	70.5	69.9
5	Shape-constraint + Confidence factorization + Cascade	$\mathcal{L}_{flow}, \mathcal{L}_{cert}, \mathcal{L}_{pose}$	65.9	82.0	76.1	59.5	69.8	**70.7**

Table 3. Ablation study of GenFlow design. The best result is in bold. Our method accomplishes the best performance (row 5).

44] trained on target datasets to focus on the accuracy of 6D pose estimation by preventing the performance bottleneck caused by the detection methods for novel objects.

4.3.1 Coarse Pose Estimation

Table 2 shows the effectiveness of the GMM-based sampling method for generating pose hypotheses. Pre-defined rotations are generated following [68] for the Naïve sampling and for creating GMM of our method. We used the $N = 576$ rotations for the Naïve sampling, and $M = 72$ and $M = 104$ which conduct $2M$ renderings (144 and 208, respectively) for ours. The results show that our method can generate a better initial pose despite the fewer renderings and network inferences.

4.3.2 GenFlow design

Table 3 shows the results on different refiner designs. All experiments are conducted using the same result of coarse estimation, which uses our GMM-based sampling with $M = 104$ without the multi-hypotheses strategy. We run 5 iterations of GenFlow refinement, each containing the 8 iterations of the inner update. We evaluated the final 6D pose from the GenFlow refiner.

Shape-Constraint Learning to estimate 2D correspondences with flow loss only is suboptimal for 6D pose estimation since it makes no use of the target shape (row 1). Imposing shape-constraint on the model benefits from the guide of 3D information without loss of generalization performance (row 2-5).

Cascade Architecture The proposed cascade architecture with multiple GenFlow modules contributes to the performance improvement (row 3, 5).

Confidence Factorization Factorizing the confidence weights to certainty and pose sensitivity helps to increase the accuracy of the output pose. Certainty estimation leverages the training depth data and makes our method robust to occlusion (row 4, 5).

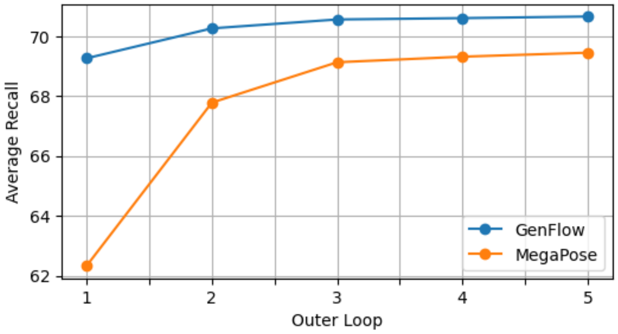


Figure 6. GenFlow vs MegaPose refiner for the RGB inputs.

4.3.3 MegaPose Refiner vs Ours

We compare the performance of the MegaPose refiner and ours for the RGB input. The same pose initializations from our coarse estimation were used for a fair evaluation. We reported the changes in AR score according to the iteration of the outer updates for both methods in Fig 6. The reported score is computed as the arithmetic mean of the AR scores for the five datasets. A single outer update takes 66.5 milliseconds for the MegaPose refiner and 98.2 milliseconds for ours, respectively, on an RTX 3090 GPU and Core i9-10920X CPU. Although ours is slower than the MegaPose refiner for a single refinement step, it achieves better performance for the same and even fewer iterations. Note that ours performs effectively by comparing a single-view rendered image for an outer update, while MegaPose refiner uses multiple rendered views as input.

5. Conclusion

In this paper, we present GenFlow, an optical flow-based method for 6D pose refinement of novel objects. First, we generate pose hypotheses using a GMM-based sampling strategy for efficient and effective coarse estimation. Then, we refine the high-scoring hypotheses with our pose refiner, our main contribution. GenFlow iteratively refines the flow and confidence with the guidance of 3D shape. Our method achieves state-of-the-art performance of 6D pose estimation for novel objects concerning RGB and RGB-D inputs.

References

[1] Christopher M. Bishop. *Pattern recognition and machine learning, 5th Edition*. Springer, 2007. 3

[2] Michael J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Fourth International Conference on Computer Vision, ICCV 1993, Berlin, Germany, 11-14 May, 1993, Proceedings*, pages 231–236. IEEE Computer Society, 1993. 2

[3] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *CoRR*, abs/1512.03012, 2015. 5

[4] Hansheng Chen, Pichao Wang, Fan Wang, Wei Tian, Lu Xiong, and Hao Li. Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2781–2790, 2022. 5

[5] Jianqiu Chen, Mingshan Sun, Tianpeng Bao, Rui Zhao, Liwei Wu, and Zhenyu He. Zeropose: Cad-model-based zero-shot pose estimation, 2023. 1, 2, 7

[6] Qifeng Chen and Vladlen Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 4706–4714. IEEE Computer Society, 2016. 2

[7] Xu Chen, Zijian Dong, Jie Song, Andreas Geiger, and Otmar Hilliges. Category level object pose estimation via neural analysis-by-synthesis. 2020. 2

[8] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 5

[9] Kyunghyun Cho, Bart van Merrienboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In *Proceedings of SSST@EMNLP 2014, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, 25 October 2014*, pages 103–111. Association for Computational Linguistics, 2014. 4

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*, pages 248–255. IEEE Computer Society, 2009. 6

[11] Maximilian Denninger, Dominik Winkelbauer, Martin Sundermeyer, Wout Boerdijk, Markus Knauer, Klaus H. Strobl, Matthias Humt, and Rudolph Triebel. Blenderproc2: A procedural pipeline for photorealistic rendering. *Journal of Open Source Software*, 8(82):4901, 2023. 5, 7

[12] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015. 2

[13] Andreas Doumanoglou, Rigas Kouskouridas, Sotiris Malassiotis, and Tae-Kyun Kim. Recovering 6d object pose and predicting next-best-view in the crowd. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 3583–3592. IEEE Computer Society, 2016. 7

[14] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas Barlow McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation, ICRA 2022, Philadelphia, PA, USA, May 23-27, 2022*, pages 2553–2560. IEEE, 2022. 5

[15] Bertram Drost, Markus Ulrich, Nassir Navab, and Slobodan Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 998–1005. Ieee, 2010. 2

[16] Bertram Drost, Markus Ulrich, Paul Bergmann, Philipp Härtinger, and Carsten Steger. Introducing mvtec ITODD - A dataset for 3d object recognition in industry. In *2017 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2017, Venice, Italy, October 22-29, 2017*, pages 2200–2208. IEEE Computer Society, 2017. 7

[17] Johan Edstedt, Ioannis Athanasiadis, Mårten Wadenbäck, and Michael Felsberg. Dkm: Dense kernelized feature matching for geometry estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17765–17775, 2023. 6

[18] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981. 2, 6

[19] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. YOLOX: exceeding YOLO series in 2021. *CoRR*, abs/2107.08430, 2021. 3, 7

[20] Yang Hai, Rui Song, Jiaojiao Li, and Yinlin Hu. Shape-constraint recurrent flow for 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4831–4840, 2023. 1, 2, 5

[21] Rasmus Laurvig Haugaard and Anders Glent Buch. Surfemb: Dense and continuous correspondence distributions for object pose estimation with learnt surface embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6749–6758, 2022. 7

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 2980–2988. IEEE Computer Society, 2017. 7

[23] Xingyi He, Jiaming Sun, Yuang Wang, Di Huang, Hujun Bao, and Xiaowei Zhou. Onepose++: Keypoint-free one-shot object pose estimation without CAD models. In *Advances in Neural Information Processing Systems*, 2022. 2

[24] Yisheng He, Yao Wang, Haoqiang Fan, Jian Sun, and Qifeng Chen. Fs6d: Few-shot 6d pose estimation of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6814–6824, 2022. 2

[25] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary R. Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Computer Vision - ACCV 2012 - 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I*, pages 548–562. Springer, 2012. 7

[26] Tomas Hodan, Pavel Haluza, Stepán Obdrzálek, Jiri Matas, Manolis I. A. Lourakis, and Xenophon Zabulis. T-LESS: an RGB-D dataset for 6d pose estimation of texture-less objects. In *2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017, Santa Rosa, CA, USA, March 24-31, 2017*, pages 880–888. IEEE Computer Society, 2017. 7

[27] Tomáš Hodaň, Frank Michel, Eric Brachmann, Wadim Kehl, Anders Glent Buch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiří Matas, and Carsten Rother. BOP: Benchmark for 6D object pose estimation. *European Conference on Computer Vision (ECCV)*, 2018. 7

[28] Tomás Hodan, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiri Matas. BOP challenge 2020 on 6d object localization. In *Computer Vision - ECCV 2020 Workshops - Glasgow, UK, August 23-28, 2020, Proceedings, Part II*, pages 577–594. Springer, 2020. 2, 7

[29] Berthold K. P. Horn and Brian G. Schunck. Determining optical flow. *Artif. Intell.*, 17(1-3):185–203, 1981. 2

[30] Yinlin Hu, Sébastien Speierer, Wenzel Jakob, Pascal Fua, and Mathieu Salzmann. Wide-depth-range 6d object pose estimation in space. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 15870–15879. Computer Vision Foundation / IEEE, 2021. 7

[31] Yinlin Hu, Pascal Fua, and Mathieu Salzmann. Perspective flow aggregation for data-limited 6d object pose estimation. In *ECCV*, 2022. 1, 2, 7

[32] Tak-Wai Hui, Xiaoou Tang, and Chen Change Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[33] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[34] Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A: Crystal Physics, Diffraction, Theoretical and General Crystallography*, 32(5):922–923, 1976. 2, 6

[35] Roman Kaskman, Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Homebreweddb: RGB-D dataset for 6d pose estimation of 3d objects. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 2767–2776. IEEE, 2019. 7

[36] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 3

[37] Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. MegaPose: 6D Pose Estimation of Novel Objects via Render & Compare. In *CoRL*. 1, 2, 3, 5, 6, 7

[38] Y. Labbe, J. Carpentier, M. Aubry, and J. Sivic. Cosypose: Consistent multi-view multi-object 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 5, 7

[39] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o(n) solution to the pnp problem. *International Journal Of Computer Vision*, 81:155–166, 2009. 8

[40] Fu Li, Ivan Shugurov, Benjamin Busam, Minglong Li, Shaowu Yang, and Slobodan Ilic. Polarmesh: A star-convex 3d shape approximation for object pose estimation. *IEEE Robotics Autom. Lett.*, 7(2):4416–4423, 2022. 2

[41] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *European Conference on Computer Vision (ECCV)*, 2018. 2

[42] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 5

[43] Lahav Lipson, Zachary Teed, Ankit Goyal, and Jia Deng. Coupled iterative refinement for 6d multi-object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6728–6737, 2022. 2, 5

[44] Xingyu Liu, Ruida Zhang, Chenyangguang Zhang, Bowen Fu, Jiwen Tang, Xiquan Liang, Jingyi Tang, Xiaotian Cheng, Yukang Zhang, Gu Wang, and Xiangyang Ji. Gdrnpp. https://github.com/shanice-l/gdrnpp_bop2022, 2022. 3, 7, 8

[45] Yuan Liu, Yilin Wen, Sida Peng, Cheng Lin, Xiaoxiao Long, Taku Komura, and Wenping Wang. Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images. In *ECCV*, 2022. 2

[46] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11976–11986, 2022. 6

[47] David G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision, Kerkyra, Corfu, Greece,*

September 20-25, 1999, pages 1150–1157. IEEE Computer Society, 1999. 2

[48] Fabian Manhardt, Gu Wang, Benjamin Busam, Manuel Nickel, Sven Meier, Luca Minciullo, Xiangyang Ji, and Nassir Navab. Cps++: Improving class-level 6d pose and shape estimation from monocular images with self-supervised learning, 2020. 2

[49] Van Nguyen Nguyen, Yinlin Hu, Yang Xiao, Mathieu Salzmann, and Vincent Lepetit. Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6771–6780, 2022. 2

[50] Van Nguyen Nguyen, Thibault Groueix, Georgy Ponimatkin, Vincent Lepetit, and Tomas Hodan. Cnos: A strong baseline for cad-based novel object segmentation. *arXiv preprint arXiv:2307.11067*, 2023. 2, 3, 7

[51] Brian Okorn, Qiao Gu, Martial Hebert, and David Held. Zephyr: Zero-shot pose hypothesis rating. In *IEEE International Conference on Robotics and Automation, ICRA 2021, Xi'an, China, May 30 - June 5, 2021*, pages 14141–14148. IEEE, 2021. 1, 2

[52] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 3

[53] Keunhong Park, Arsalan Mousavian, Yu Xiang, and Dieter Fox. Latentfusion: End-to-end differentiable reconstruction and rendering for unseen object pose estimation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[54] Ivan Shugurov, Fu Li, Benjamin Busam, and Slobodan Ilic. Osop: A multi-stage one shot object pose estimation framework. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6835–6844, 2022. 1, 2, 7

[55] Andrea Simonelli, Samuel Rota Bulo, Lorenzo Porzi, Manuel Lopez-Antequera, and Peter Kontschieder. Disentangling monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[56] Yongzhi Su, Mahdi Saleh, Torben Fetzer, Jason Rambach, Nassir Navab, Benjamin Busam, Didier Stricker, and Federico Tombari. Zebrapose: Coarse to fine surface encoding for 6dof object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6738–6748, 2022. 7

[57] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[58] Jiaming Sun, Zihao Wang, Siyu Zhang, Xingyi He, Hongcheng Zhao, Guofeng Zhang, and Xiaowei Zhou. Onepose: One-shot object pose estimation without cad models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6825–6834, 2022. 2

[59] Martin Sundermeyer, Maximilian Durner, En Yen Puang, Zoltan-Csaba Marton, Narunas Vaskevicius, Kai O. Arras, and Rudolph Triebel. Multi-path learning for object pose estimation across domains. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[60] Mingxing Tan and Quoc V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 6105–6114. PMLR, 2019. 2

[61] Zachary Teed and Jia Deng. RAFT: recurrent all-pairs field transforms for optical flow. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part II*, pages 402–419. Springer, 2020. 2, 4, 6

[62] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[63] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. 2018. 7

[64] Yang Xiao, Xuchong Qiu, Pierre-Alain Langlois, Mathieu Aubry, and Renaud Marlet. Pose from shape: Deep pose estimation for arbitrary 3d objects. In *30th British Machine Vision Conference 2019, BMVC 2019, Cardiff, UK, September 9-12, 2019*, page 61. BMVA Press, 2019. 2

[65] Yang Xiao, Yuming Du, and Renaud Marlet. Posecontrast: Class-agnostic object viewpoint estimation in the wild with pose-aware contrastive learning. In *International Conference on 3D Vision (3DV)*, 2021. 2

[66] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 793–803, 2019. 2

[67] Heng Yang, Jingnan Shi, and Luca Carlone. TEASER: fast and certifiable point cloud registration. *IEEE Trans. Robotics*, 37(2):314–333, 2021. 7

[68] Anna Yershova, Swati Jain, Steven M. LaValle, and Julie C. Mitchell. Generating uniform incremental grids on $SO(3)$ using the hopf fibration. *Int. J. Robotics Res.*, 29(7):801–812, 2010. 8

[69] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2