Optimizing RGB-D Fusion for Accurate 6DoF Pose Estimation

Lounes Saadi, Bassem Besbes, Sebastien Kramm, and Abdelaziz Bensrhair

***Abstract*—Today's standard object localization systems often do not meet the industry's demands regarding 2D and 3D accuracy for digital manufacturing applications. Two targets are considered: digital-based assistance and robotic inspection. 2D precision is necessary to provide accurate assistance whilst 3D precision is crucial to get an inspection as much close to the object's true state. In this letter, we propose a new pose estimation system which ensures highest both 2D and 3D precision. While most RGB-based solutions focus on obtaining best 2D accuracy, RGBD-based systems mainly use depth information to maximize 3D accuracy. Very few solutions propose a way to jointly optimize both constraints. Nonetheless, pose estimation should produce high accuracy as a slight 2D error can result in a large 3D error (and inversely). To address this problem, we present a new system which uses RGB-D to fully take advantage of the depth information. A new 3D primitive is proposed in order to minimize the effect of RGB-D noise on 3D coordinates accuracy. CNN Keypoint Detector (KPD) method is used to localize this new primitive in order to achieve pose estimation task. Finally, we propose a novel refinement method which ensures optimal precision as both RGB and depth information are fused. We show the results of our experimentation on widely-used and challenging Linemod and Occlusion datasets. We demonstrate that our solution outperforms state-of-the-art methods when taking into account both 3D and 2D accuracy.**

***Index Terms*—2D & 3D accuracy, keypoint detector, object's localization, refinement, RGB-D.**

I. Introduction

MORE and more industries turn to digital manufacturing to optimize industrial processes. Consequently, new applications have been developed to meet this modern need. Computer vision, and more precisely object's localization techniques, have mainly been used as a solution for some of these applications. In this paper, we consider two ones: digital-based assistance and robotic inspection. These two applications can be used together. A typical process first uses digital-based assistance to help operators for production and then robotic inspection to control (shown in Fig. 1). Robotic inspection introduces a hardware constraint as limited computing power is mostly embedded in robots. The process can also be inverted using robotic inspection for defect detection and digital-based assistance to accurately display these errors. Augmented reality is particularly well-suited for this last purpose. Digital-based assistance using augmented reality techniques requires optimal 2D accuracy to provide accurate instructions. On the other hand, robotic inspection needs an excellent 3D precision to analyze object's possible defects, such as, for example, assembly errors or scratched pieces. In the literature, most methods either target 2D or 3D accuracy [1]–[3], but none explicitly proposes a solution to jointly optimize such constraints. Hence, we propose a new generic pose estimation system which ensures optimal both 2D and 3D precision.

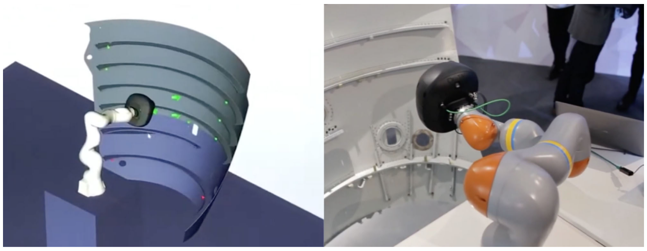


Fig. 1. **Visualization of the employed robot for the object's inspection**: This robot must accurately localize an object in order to provide consistent inspection information. An RGB-D camera and a PCV-Nuvo computer (CPU-only) are embedded in the robot. Left: A virtual visualization of the robot in front of the object and the parts to inspect (green). Right: The robot in front of the real object while providing a localization.

Traditionally, object's pose estimation was achieved by matching image features with the object model [4], [5]. 6D pose was estimated using these 2D-3D correspondences. However, such methods suffer from texture-less objects or cluttered environments. Recent works show that CNN-based architectures can solve pose estimation task and achieve high performances. These methods can be divided in two categories. Firstly, [6], [7] represent a category of methods which consider pose estimation task as a regression problem and output object's 3D rotation and translation. Secondly, methods [2], [8] try to establish 2D-3D correspondences between model and image by predicting keypoints 2D locations. 6D pose is recovered using

Manuscript received October 15, 2020; accepted February 3, 2021. Date of publication February 23, 2021; date of current version March 15, 2021. This letter was recommended for publication by Associate Editor P. de la Puente and Editor C. Cadena Lerma upon evaluation of the reviewers' comments. This work was supported by Diotasoft SAS and the French National Research and Technology Agency (ANRT). *(Corresponding author: Lounes Saadi.)*

Lounes Saadi is with the Laboratoire d'Informatique, de Traitement de l'Information et des Systemes (LITIS), INSA Rouen, Rouen 76000, France, and also with the Vision Research Department at DIOTA Massy 91300, France (e-mail: lounes.saadi@insa-rouen.fr).

Bassem Besbes is with the Vision Research Department, DIOTA, Rouen, France (e-mail: bbe@diota.com).

Sebastien Kramm and Abdelaziz Bensrhair are with the Laboratoire d'Informatique, de Traitement de l'Information et des Systemes (LITIS), INSA Rouen, Rouen 76000, France (e-mail: sebastien.kramm@univ-rouen.fr; abdelaziz.bensrhair@insa-rouen.fr).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2021.3061347, provided by the authors.

Digital Object Identifier 10.1109/LRA.2021.3061347

Perspective-n-Points [9] method. These methods achieve interesting performance but their accuracy remains extremely sensitive to 2D-3D correspondences because of the lack of depth information.

RGB-D sensors have been used to improve pose estimation accuracy. However, depth data are poorly stable as there can be discontinuities and noise. Recently, Deep Learning techniques have proven to be efficient and inspire [3], [10] to extend common CNN architectures to integrate depth information as a joint input with RGB image. Even if these approaches provide the highest 3D accuracy, they rarely investigate 2D constraint.

To address this problem, we propose a new system which takes advantage of depth information from an RGB-D sensor to efficiently localize objects in terms of both 2D and 3D accuracy. A KPD is used to predict 2D locations of a new primitive optimized for depth backprojection. As our solution does not explicitly focus on the employed KPD method, any architectures can be used. Even though a simple KPD has been chosen in this work, we demonstrate that our method achieves state-of-the-art performances. Starting with a low 2D error, we inject depth data to significantly increase performance for both 2D and 3D accuracy. The proposed pipeline contains: i) CNN-based prediction of a new primitive; ii) 3D coordinates estimation with the depth information and 3D-3D correspondences pose estimation; iii) a new RGB-D refinement step. Our system demonstrates the best compromise regarding 2D and 3D precision.

We evaluate our method on two popular datasets: Linemod [11] and Occlusion [12]. We first show that our method achieves high scores for 3D metric (ADD) [7] and 2D metric (2D projection) [13] in front of state-of-the-art methods. As we propose a generic solution for digital-based assistance and robotic inspection, 3D and 2D accuracy cannot be distinguished. Finally, we demonstrate our method's robustness to a very challenging dataset with highly occluded objects.

To summarize, our work presents the following main contributions:

- We propose a new method to take advantage of RGB-D sensors without integrating it in complex and time-consuming CNN-based architectures. We benefit from 3D-3D correspondences pose estimation with a new primitive perfectly fitted for accurate depth backprojection.
- We introduce a novel refinement method which fuses geometric information from CAD with depth information from the RGB-D sensor.
- We evaluate and compare our method with state-of-the-art methods and demonstrate both 2D and 3D optimal accuracy.

II. Related Work

Pose from RGB images: Recent approaches using deep learning show their ability to tackle pose estimation task. PoseNet [6] proposes to directly regress 6D pose from a single RGB image using CNN-based architectures. However, directly estimating 6D pose from an RGB image remains complex because of the huge search 3D space. To overcome this problem, PoseCNN [7] disentangles rotation and translation estimation to use object's 3D translation as a prior for the rotation regression. Non-linearity of the 3D space makes the task quite difficult and unstable for the training. In a way to simplify the task, [15], [16] discretize 3D space and transform 6D pose estimation as a classification task. Such methods predict a very coarse result and need a robust refinement step.

Most recent methods [1], [2], [12] choose to use a two-stage pipeline. Using CNN-based keypoint detector architectures to predict 2D keypoints on the object and PnP with 2D-3D correspondences. In this way, [13], [17] detect the 8 corners of the 3D bounding box. However, such methods are very sensitive to occlusions. Inspired by human pose estimation, [1], [8] propose to use heatmaps to predict 2D locations. Heatmaps encounter difficulties when keypoints are outside of the image. To address this problem, PVNet [2] uses semantic segmentation and dense voting to predict 2D keypoints locations. Every pixel on the object votes for the keypoints 2D directions. [2] is the current most accurate method in terms of 2D accuracy. More recently, some techniques [18], [19] predict a dense map of 2D-3D correspondences using Auto-Encoder architectures to improve the robustness to partial occlusions. Even though these methods demonstrate high precision, accurately predicting 6D pose from an RGB image remains a challenging task due to the lack of depth information. We address this problem using a CNN-based keypoint detector to accurately detect 2D keypoints and injecting depth information in the pose estimation and refinement processes to increase 3D accuracy without lowering 2D's one.

Pose from RGB-D images: Depth information can be an important asset to improve pose estimation accuracy. Some of traditional methods focus on point clouds registration [20], [21]. These methods are very sensitive to sensor noise. The task of registering a CAD model to a point cloud is still a challenging problem.

Inspired by the promising performances on RGB data, [22] proposes a CNN-based architecture which fuses RGB images with depth information, extending the 3-channel input to a 4-channel input with depth image. However, this method does not use the full potential of 3D information. DenseFusion [10] proposes a new CNN-based architecture that fuses 3D data to 2D appearance information in a single feature vector. PVN3D [3] extends [2] dense voting scheme to 3D using 3D Hough to predict 6D pose and outperforms state-of-the-art methods. However, 2D accuracy is not entirely ensured as these methods mainly target 3D precision. In this work, we show that we can obtain state-of-the-art performances without integrating depth information in a complex CNN-based architecture.

Refinement: While some methods [2], [3] show good performances without refinement, accuracy can significantly be increased with a refinement step. Following current trends, promising results are shown using deep learning techniques. [23], [24] propose a network which is trained to align predicted and observed object's mask. However, using a specific CNN architecture for the refinement can be complex and time consuming. Traditional methods can directly refine a pose without a deep network and still achieve good performances. Most RGB-D methods which integrate a refinement step use famous Iterative Closest Points (ICP) [25] or its variants [26], [27] which align

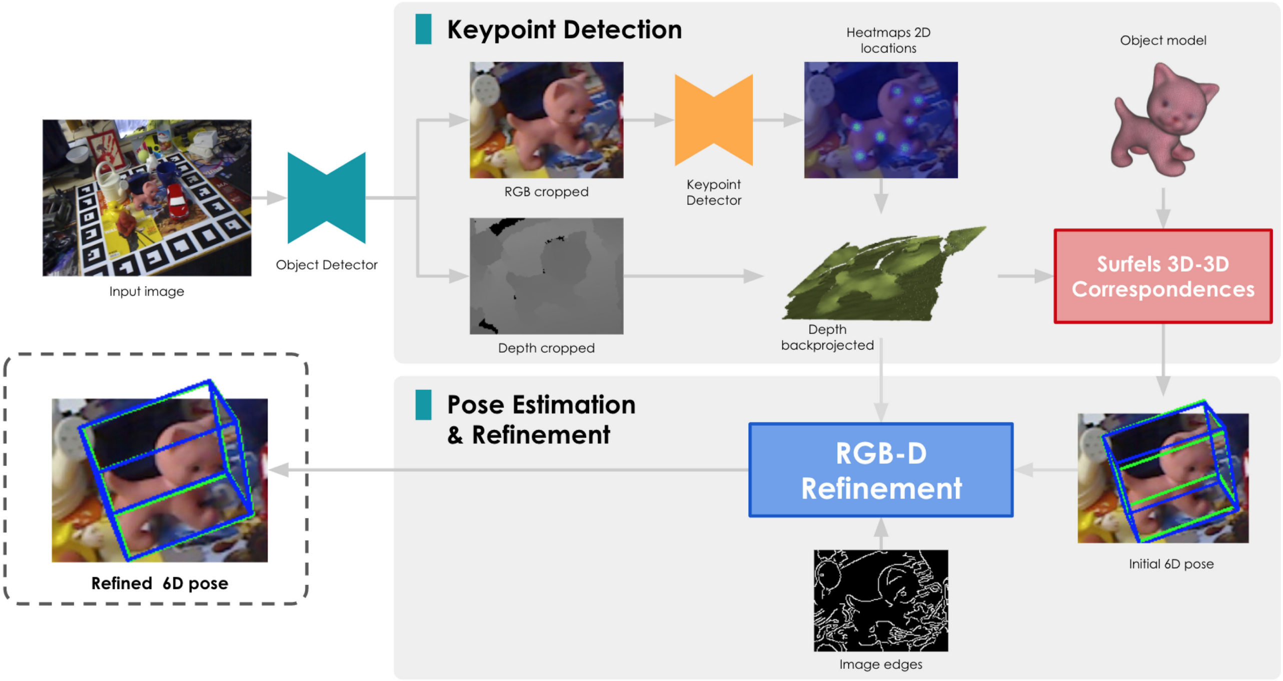


Fig. 2. **Overview of our 6D pose estimation pipeline:** We predict Surfels 2D locations via heatmap-based KPD architecture (see Sec. III-A). Depth image is used to backproject Surfels predictions and obtain 3D-3D correspondences. Robust initial 6DoF pose (blue bounding box) is estimated using the Kabsch algorithm [14] (see Sec. III-B) before an RGB-D refinement step (blue bounding box) is achieved using object's model, image's edges, depth image and initial pose (see Sec. III-C). Green bounding box shows object's ground truth pose.

two point clouds. Such methods are accurate but the registration step is computationally demanding and very sensitive to the initial pose. Moreover, 2D accuracy is not ensured as only 3D information is optimized. When only RGB information is available, edge-based methods [4], [5] are commonly used and demonstrate high performances. Pose refinement is achieved by matching the model's salient edges [4], [28] and occluding contours [29] with the image's edges. Occluding contours refer to the object's silhouette (object's external boundary) related to a specific camera point of view. Even though edge-based refinement is fast and accurate, pose accuracy is sensitive to edges-contours correspondences and their distributions over the image. In order to solve this issue, we propose a novel refinement method taking advantage from both edge-based methods and depth information to perfectly constraint 2D and 3D accuracy.

III. Proposed Approach

In this work, we present a novel 6DoF pose estimation pipeline using RGB-D images. Given inputs RGB and depth images I_{RGB} and I_{depth}, containing a known object, we estimate the camera rotation and translation in the model coordinate system. The 6D pose is a rigid homogenous transformation $T \in SE(3)$ composed of 3D rotation $R \in SO(3)$ and 3D translation $t \in \mathbb{R}^3$, $T = [R|t]$.

Inspired by the recent works [1]–[3], [8], we choose to use a CNN-based architecture to detect 2D keypoints from I_{RGB}. We backproject these keypoints using depth image I_{depth} to obtain 3D-3D correspondences between object camera and object model. Our method uses a RANSAC scheme [30] to estimate initial rigid transformation. Finally, the 6D pose is refined with a novel refinement module which fuses image edges and depth information.

A. Keypoint Detection

1) Keypoint Detector: Choosing the proper KPD is often a hard task and strongly depends on the considered application (real-time, 2D or 3D accuracy). In this work, we propose a pipeline which can be employed with any KPD. In this context, we selected a simple KPD architecture following [1]. This method uses Resnet-101 as a backbone and outputs heatmaps representing prediction's confidence (shown in Fig. 2). Heatmap-based methods show high 2D accuracy as they train to minimize 2D distances between predicted and ground truth keypoints locations. Similar to [1], we train the network optimizing the following Loss function:

$$\mathcal{L} = \frac{1}{n}\sum_{i=1}^{n}\left(\tilde{\mathcal{H}}_i - \mathcal{H}_i\right)^2 \tag{1}$$

where $\tilde{\mathcal{H}}_i$ and $\mathcal{H}_i$ represent predicted and ground truth heatmaps. The number of heatmaps n is defined by the number of points to predict.

2) 3D Keypoint: Using a KPD implies choosing a type of primitive to detect. Several primitives are presented in the literature, some methods [13], [17] use 8 corners of the object's bounding box, whereas [1] uses a 3D version of SIFT. Most recent methods [2], [3] use Farthest Point Sampling (FPS) which ensures 3D uniform distribution. Even if selecting one of these primitives might seem straightforward, the task is not easy and should be related to both the KPD and the pose estimation method implemented.

In this work, we demonstrate that pose estimation accuracy can be increased with both the right primitive and pose estimation methods. To limit the negative impact of 2D points distribution, we use 3D-3D correspondences obtained thanks to depth information. Unfortunately, RGB-D sensors are affected by noise and range distances which quadratically decrease their depth accuracy [31].

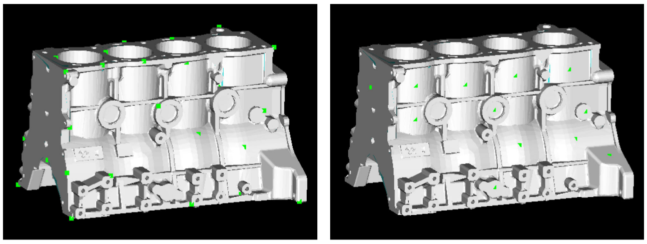
We address this problem with a new 3D primitive which minimizes the impact of RGB-D accuracy. RGB-D sensors encounter problems to accurately estimate depth on surfaces close to the object's contours. Detecting 2D keypoints near these areas would produce poor performance when backprojecting depth information. In contrast, smooth surfaces with low geometric distinctiveness maximize depth accuracy as there can be significant errors on highly curved surfaces. We define smooth surfaces as the most distant areas from contours. We argue that the further points are from 3D contours neighborhood, the more optimal the accuracy would be. We call such surfaces **Surfels**. The key advantage is that every object has smooth surfaces (even complex ones) whereas sharp edges cannot be found on many objects (e.g. round objects). We propose an offline extraction process achieved on the object's model. A small sequence with annotated 6D poses and depth images is required. We project extracted 3D contours [29] on each model's rendered image. We select the most diverged points from the projected contours thanks to a 2D distance transform. For each of these points, we assign a confidence score meant to check the local depth accuracy. This score is computed after backprojecting the point and compares its 3D position and orientation to the ground truth (thanks to the annotated 6D poses). We define acceptance thresholds on Euclidean and angular errors. At the end of the sequence, a list of Surfels candidates with their confidence is obtained. In order to select Surfels, a K-Means clustering is used. Each cluster has an overall confidence score which is the sum of every point's confidence in the cluster. We retain N best clusters and select the points with the highest confidence of the clusters. The number N of clusters is an input parameter of the algorithm. We use this new primitive to train a heatmap-based KPD. Fig. 3 illustrates sampling's differences between FPS and Surfels points.

B. Pose Estimation

As presented in Fig. 2, we use depth image I_{depth} to backproject Surfels 2D predictions. Backprojection allows us to obtain 3D Surfels in the camera coordinate system. From a depth image, we can compute 3D point $P = [x, y, z]$:

$$P = \pi^{-1}(u, v) = d \cdot \begin{bmatrix} (u - c_u)/f_u \\ (v - c_v)/f_v \\ 1 \end{bmatrix} \quad (2)$$

where d is the depth value at $[u, v]$, $[c_u, c_v]^T$ and $[f_u, f_v]^T$ are the principal point and focal length of the RGB camera.



(a) Farthest Point Sampling (b) Surfels

Fig. 3. Comparison between commonly-used FPS (3(a)) and Surfels (3(b)) sampling on an industrial object. We can see that Surfels are sampled on smooth surfaces (low curvatures) whereas FPS points are distributed all over the object.

PnP algorithms have two strong constraints to ensure an accurate pose estimation: i) a minimum number of correspondences; ii) well distributed correspondences on the image. On the other hand, 3D-3D correspondences-based methods overcome the distribution constraint as depth information is provided with 3D-3D correspondences. Using a sparse detection method (see Sec. III-A), distribution's constraint is not always satisfied. We tackle this problem by using 3D-3D correspondences to estimate the initial pose. Along with 3D Surfels extracted on the object's model, we obtain 3D-3D correspondences between the camera and the model. The rigid transformation is estimated using the Kabsch algorithm [14]. To increase robustness, we employ this algorithm within a RANSAC scheme to filter outliers correspondences out.

C. Refinement

As shown in Fig. 2, the last step of our system consists in iteratively refining the initial pose in order to provide optimal performance. We observed that edge-based methods [4], [5] produce a small reprojection error but have trouble to perfectly estimate depth. In comparison to ICP, these kinds of methods are faster and less dependent on sensor quality. We present a novel robust and accurate refinement method that fuses edge constraint accuracy with depth information. While edge-based methods ensure perfect accuracy in the image plane, we jointly use depth error minimization to correct errors in 3D space. The process iteratively refines a coarse initial pose by concurrently minimizing edge and depth constraints.

1) Edge Constraint: Thanks to the initial predicted pose, we can extract image edges within a region of interest. Inspired by [32], we use directional Chamfer to robustly match edges with the model's 3D contours. Most methods extract 3D contours from the 3D model's sharp edges using an off-line algorithm. Unfortunately, many objects do not have sharp edges (e.g. objects with rounded shapes). To overcome this limitation, we extract occluding contours according to the predicted camera point of view. The matching step between image edges and 3D model contours is achieved at each refine iteration. This step provides 2D-3D correspondences between the image edges and the model contours. Refined pose is robustly estimated similarly to [28]. During the optimization, the edges residual vector which sums point-to-plane distances is minimized following:

$$r_{edges}(T) = \sum_{i=1}^{k} \rho(|\pi(TX_i) - x_i) \cdot n_i|) \qquad (3)$$

where X_i represents 3D contours of the model, x_i and n_i represent images edges and normals. $\pi(\, . \,)$ projects 3D points. $\rho(\, . \,)$ denotes a robust Geman-McClure function [28] to handle outliers.

2) Depth Constraint: Depth constraint enables to correctly estimate depth. At each iteration, we project model vertices with predicted transformation on I_{depth} and backproject coordinates with (2). We then compute the error between predicted and measured 3D points. As we want to guarantee accurate depth, residual vector sums Z-value distances:

$$r_{depth}(T) = \sum_{j=1}^{l} \rho(|\lfloor TY_j \rfloor_Z - \lfloor \pi^{-1}(\pi(TY_j)) \rfloor)_Z|) \qquad (4)$$

where Y_i represents 3D vertices sampled on the model, $\lfloor \, . \, \rfloor_Z$ represents the Z component from a 3D point. $\rho(\, . \,)$ denotes a robust Geman-McClure function to handle outliers.

3) Joint Optimization: Final stage of the refinement is the optimization. Fusing (3) and (4), we propose a bi-constraint equation optimized with the Levenberg-Marquardt algorithm [28]:

$$T_{refined} = \underset{T}{\text{argmin}} \quad \alpha \, r_{edges}(T) + \beta \, r_{depth}(T) \qquad (5)$$

where α and β represent normalization factors. As r_{edges} and r_{depth} are respectively expressed in pixels and meters, the scale is different making the optimization sub-optimal. In other works, an empirical factor is often applied to solve this problem. We propose a way to automatically turn our residual vectors into the same scale. We express pixels and meters errors with respect to their corresponding 2D and 3D bounding boxes. Therefore, α and β are the normalization factors to turn residuals to the same scale and are automatically defined before each refinement with the initial pose to refine.

IV. Experiments

We now evaluate our approach on two challenging datasets Linemod and Occlusion which contain annotated 6D poses for different types of objects in challenging environments. Firstly, the Linemod dataset is used to demonstrate motivations of each step of our system. Then, overall performance is given for both Linemod and Occlusion datasets.

A. Implementation Details

We extracted Surfels for every object of the Linemod dataset. In the experiments, we fixed the number of Surfels to 15 with acceptance thresolds set to 1 cm and $20 \deg$ for Euclidean and angle errors. We trained the heatmap-based KPD using PyTorch 1.5.0 with the Adam optimizer on an Nvidia RTX 2060 GPU. Training was done with a batch size of 8 and an initial learning rate of 0.0001 during 200 epochs. To estimate the initial pose from 3D-3D correspondences, we used an implementation proposed by the Point Cloud Library. Finally, refinement's joint optimization is implemented using the G2o library.

TABLE I
Performance Comparison of Surfel and FPS Primitives at the Depth Backprojection Step

methods	FPS	Surfels
Accuracy (m)	0.023	**0.014**
Depth availability (%)	93.4	**97.7**

B. Datasets

1) Linemod: This dataset has become over time a standard for 6D pose estimation benchmarks. Most recent methods [2], [3], [10], [13], [17] use this dataset to compare to state-of-the-art. Linemod collects data for 13 objects captured with a Kinect V2 on challenging localization conditions: different sizes and poorly-textured objects, cluttered environments, etc.

2) Occlusion: This dataset is an extension of the Linemod dataset. It was created from a subset of Linemod and proposes 6D poses annotations for every object in images. This dataset levels difficulty up and challenges robustness of solutions to occlusions.

C. Metrics

Our method is benchmarked and compared to state-of-the-art using two widely-used metrics: average 3D distance of model points (ADD) and 2D projection.

1) *ADD Metric:* This metric represents 3D accuracy and computes the mean Euclidean distance between predicted transformed model points and ground truth transformed model points. A pose is classified as successful if the mean distance is below 10% [7] of the object's diameter. For symmetric objects, the metric is slightly different, the mean distance with closest point (ADD-S) is computed.

2) *2D Projection metric:* This metric represents 2D accuracy and computes the mean pixel distance between predicted projected model points and ground truth projected model points. A pose is classified as successful if the mean distance is below 5 pixels [13].

D. System's Step Analysis

Keypoint detection: To validate Surfels as the most suitable KPD input, we compared commonly-used FPS and Surfels with Linemod's sequences. Annotated ground truth 6D poses were used to project both primitives on depth images. Then, we employed the Ray-Casting method, as proposed in [33], to get real 3D coordinates and compute mean error between backprojected points and real coordinates. As depth estimation cannot be ensured on the entire depth image, we report availability rate for each primitive. Table I presents the average result of every Linemod's object. Accuracy is expressed in meters and Surfels are almost twice more accurate than FPS points. Besides, Surfels ensure to achieve a higher rate of success backprojection. They are very well suited to accurately achieve depth backprojection in order to estimate 6D pose from 3D-3D correspondences.

TABLE II
POSE ESTIMATION PERFORMANCE COMPARISON USING EPNP [9] WITH 2D-3D CORRESPONDENCES AND KABSCH ALGORITHM [14] WITH 3D-3D CORRESPONDENCES IN TERMS OF ADD(-S) AND 2D PROJECTION METRICS

methods	2D-3D	3D-3D
ADD(-S)	59.4	**90.6**
2D Projection	**88.6**	82.5

TABLE III
REFINEMENT PERFORMANCE OF DIFFERENT METHODS IN TERMS OF ADD(-S) AND 2D PROJECTION METRICS

methods	Initial pose	Edge-based	ICP	OURS
ADD(-S)	91.9	62.1	92.9	**95.6**
2D Projection	84.7	76.9	87.1	**90.2**

Pose estimation: Table II compares 6D pose accuracy when using 2D-3D and 3D-3D correspondences for pose estimation. 3D-3D correspondences pose estimation presents a lower performance (6.1% lower) in terms of 2D projection errors. This result is mainly due to the fact that PnP algorithm minimizes reprojection error and is supposed to produce optimal 2D accuracy. 3D-3D correspondences pose estimation is significantly more accurate in terms of ADD(-S) (31.2% higher). Contrary to PnP which needs enough accurate 2D-3D correspondences to optimally constraint depth, fewer accurate 3D-3D correspondences are needed to estimate accurate initial pose. Table II confirms our choice to use 3D-3D correspondences instead of traditional 2D-3D ones. We argue that lower 2D performance will be corrected and improved with the RGB-D refinement step.

Refinement: As mentioned in Section III-C, edge-based and ICP-based methods have shown interesting results in the literature. To demonstrate our contribution, we implemented both methods to compare to ours. Table III shows that our RGB-D refinement outperforms both edge-based and ICP-based methods in terms of both 3D and 2D metrics. Note that edge-based method decreases initial pose's accuracy and experiences difficulties with Linemod challenging cluttered environments. ICP-based method outputs interesting results but is outperformed by our RGB-D refinement module. Finding ICP's optimal parameters can sometimes be cumbersome. It is interesting to mention that our refinement method is generic and does not need specific settings. We demonstrate that our method allows to find the optimal balance between edges and depth information.

E. Ablation Study

Fig. 4 highlights the performance of our refinement method when evaluating the initial and refined pose on the Linemod dataset. As we can see, our refinement step increases accuracy both in terms of ADD(-S) and 2D Projection in almost every case. An improvement of respectively almost 4% and 6% can be seen. However, we can observe a significative regression with the object Iron. On the 2D Projection metric, the refinement experiences lower performance than the initial pose (drop of ≈ 13%). This specific case can be explained as the edge constraint is not satisfied due to a lack of 3D contours extracted from the model.

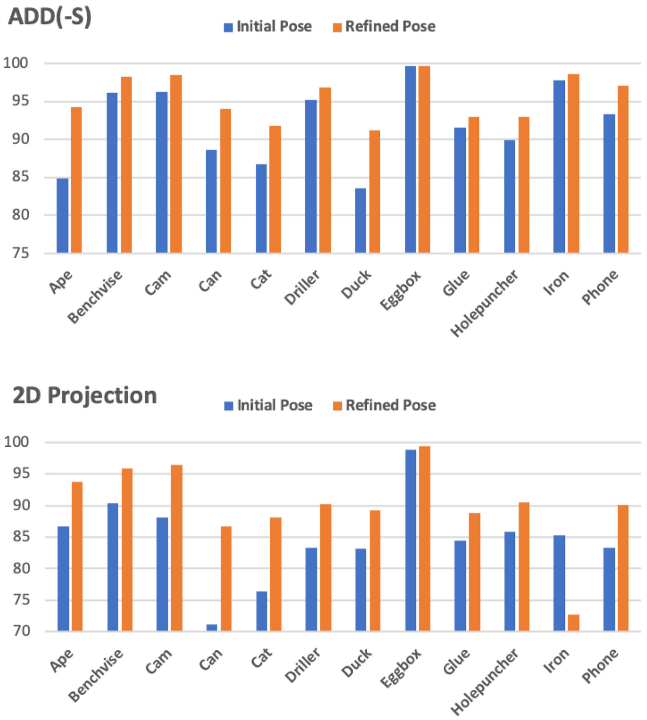


Fig. 4. Ablation study which demonstrates the influence of the refinement step on the Linemod dataset.

TABLE IV
QUANTITATIVE RESULTS ON THE LINEMOD DATASET USING **ADD(-S)** METRIC. (*) INDICATES SYMMETRIC OBJECTS

methods	Tekin [17]	Pix2Pose [19]	PVNet [2]	DF [10]	PVN3D [3]	OURS
ape	21.6	58.1	43.6	92	**97.3**	94.2
benchvise	81.8	91	**99.9**	93	99.7	98.2
cam	36.6	60.9	86.7	94	**99.6**	98.5
can	68.8	84.4	95.5	93	**99.5**	94
cat	41.8	65	79.3	97	**99.8**	92
driller	63.5	76.3	96.4	87	**99.3**	97.2
duck	27.2	43.8	52.6	92	**98.2**	91.5
eggbox*	69.6	96.8	99.1	**100**	99.8	99.6
glue*	80	79.4	95.7	**100**	**100**	92.5
holepuncher	42.6	74.8	81.9	92	**99.9**	92.1
iron	75	83.4	98.9	97	**99.7**	98.7
lamp	71.1	82	99.3	95	**99.8**	96.5
phone	47.7	45	92.4	93	**99.5**	97.2
average	55.6	72.4	86.3	94	**99.4**	95.6

F. Overall Performance on Linemod and Occlusion

We evaluated our method on every object of the Linemod dataset. Table IV compares our method with state-of-the-art [2], [3], [10], [17], [19] using ADD(-S) metric. The employed KPD proposed by [1] provides low ADD(-S) performance (72.6%). We significantly increase this accuracy (95.6%) and achieve second best performance, ahead of RGB-D CNN-based architecture DenseFusion [10]. As we target both digital-based assistance and robotic inspection applications, we also have to compare with 2D projection metric. Table V compares the few methods [2], [13], [17] that propose 2D Projection results. Our solution achieves good performance but is outperformed by [2]. However, as mentioned, ADD(-S) and 2D Projection metrics

TABLE V
QUANTITATIVE RESULTS ON THE LINEMOD DATASET USING **2D PROJECTION** METRIC

methods	Tekin [17]	BB8 [13]	PVNet [2]	OURS
ape	92.1	96.6	**99.2**	94.3
benchvise	95.1	90.1	**99.8**	95.7
cam	93.2	86	**99.2**	96.4
can	96.4	91.2	**99.9**	85.8
cat	97.4	98.8	**99.3**	87.9
driller	79.4	80.9	**96.9**	90.3
duck	94.6	92.2	**98**	90
eggbox	90.3	91	99.3	**99.4**
glue	96.5	92.3	**98.4**	89
holepuncher	92.9	95.3	**100**	90.1
iron	82.9	84.9	**99.2**	73.3
lamp	76.9	74.4	**98.3**	87.5
phone	86.4	85.3	**99.4**	90.2
average	90.4	89.3	**99**	90.2

TABLE VI
QUANTITATIVE RESULTS ON THE OCCLUSION DATASET USING **ADD(-S)** METRIC. (*) INDICATES SYMMETRIC OBJECTS

methods	Pix2Pose [19]	PVNet [2]	Seg-Driven [34]	Hu [35]	OURS
ape	22	15.81	12.1	19.2	**70.16**
can	44.7	63.3	39.9	65.1	**82.79**
cat	22.7	16.68	8.2	18.9	**54.17**
duck	15	25.24	45.2	**69**	53.36
driller	44.7	65.65	17.2	25.3	**70.7**
eggbox*	25.2	50.17	21.1	52	**77.33**
glue*	32.4	49.62	35.8	**51.4**	49.7
holepuncher	49.5	39.67	36	45.6	**78.92**
average	32	40.77	26.94	43.31	**67.14**

cannot be distinguished and have to be analyzed together. We demonstrate that we outperform state-of-the-art methods when taking into consideration both 3D and 2D metrics.

To show the efficiency and the robustness of our method, we extend the experiments to the Occlusion dataset which contains highly occluded objects. In order to be consistent with our results on Linemod, we compare to state-of-the-art for both ADD(-S) and 2D projection metrics. Table VI reports performance for ADD(-S) metric and shows that our method achieves the highest performance and outperforms [2], [8], [19], [34], [35] by almost 25%. The results show that our pipeline achieves high robustness to occlusions. Moreover, we found interesting to compare our method with the results presented in the *2020 BOP Challenge* [36]. The only method which achieves better accuracy than ours is the BOP Challenge Winner CosyPose+ICP (71.4%). Note that we outperform Pix2Pose+ICP (58.8%). Table VII compares accuracy in terms of 2D projection metric and shows that we achieve lower performance than [2], [8], [35]. However, similarly to the Linemod dataset, we analyse both ADD(-S) and 2D projection together. Our method outperforms state-of-the-art and shows the lowest deviation regarding both ADD(-S) (12.9%) and 2D projection (17.9%). These results demonstrate that our system is generic and performs almost equally for every object. Qualitative results are provided in Fig. 5 and show the accuracy of our method. Initial estimation (see Section III-B) and refined poses (see Section III-C) are compared with the ground truth. Poses are presented as projections of objects 3D bounding boxes.

TABLE VII
QUANTITATIVE RESULTS ON THE OCCLUSION DATASET USING **2D PROJECTION** METRIC

methods	Oberweger [8]	PVNet [2]	Seg-Driven [34]	Hu [35]	OURS
ape	69.9	69.14	59.1	**70.3**	70.16
can	82.6	**86.09**	59.8	85.2	74.37
cat	65.1	65.12	46.9	**67.2**	49.61
duck	61.4	61.44	59	**71.8**	49.95
driller	**73.8**	73.06	42.6	63.6	58.2
eggbox	13.1	8.43	11.9	12.7	**40.8**
glue	54.9	55.37	16.5	**56.5**	20.17
holepuncher	66.4	69.84	63.6	**71**	68.57
average	60.9	61.1	44.92	**62.29**	53.98

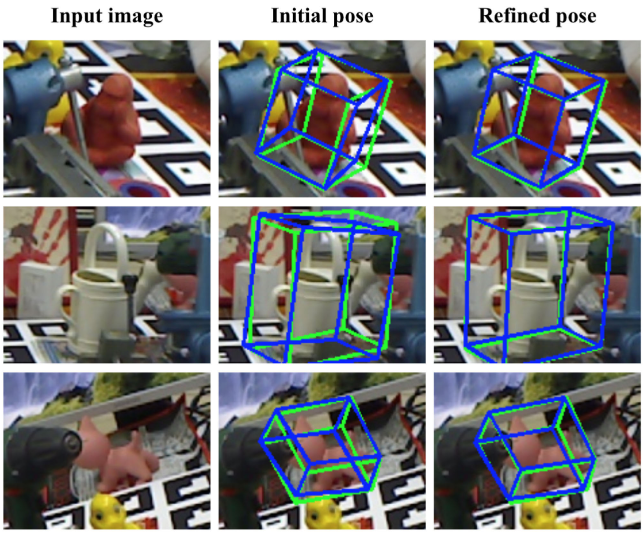


Fig. 5. **Qualitative results**: Predicted poses before and after the refinement step on 3 different objects of the Occlusion dataset. After the refinement step, high accuracy can be observed as green (ground truth) and blue (prediction) bounding boxes are very close.

Besides evaluating the accuracy of our system, it is important to notify the runtime time. As mentioned, we have a strong hardware limitation, our method can only run on the CPU. The heatmap-based CNN, pose estimation and refinement are executed on an Intel i7-6700 CPU. The entire system performs at approximately 350 ms.

V. CONCLUSION

We presented a novel system for object's localization from RGB-D images for digital-based assistance and robotic inspection. We showed that we can optimize the use of depth information without a complex and time-consuming integration within a CNN-based architecture. From KPD-based new primitive's 2D predictions, initial 6D pose is estimated using 3D-3D correspondences obtained with depth information. Final pose is provided by a novel RGB-D refinement method which jointly optimizes edges and depth. Every step of our system has been evaluated and validated with experiments. We demonstrated high 3D and 2D accuracy of our pipeline and compared it to state-of-the-art methods. We also showed robustness with occlusions and cluttered backgrounds. Furthermore, we evaluated our method using another sensor (Occipital StructureCore) on industrial objects. In future work, we plan to publish the content of this experiment along with a new dataset of complex industrial objects. In order to better address the industry's needs, our dataset will feature bigger objects with complex geometry and several appearance variations.

References

[1] Z. Zhao, G. Peng, H. Wang, H. S. Fang, C. Li, and C. Lu, "Estimating 6D pose from localizing designated surface keypoints," vol. abs/1812.01387, 2018.

[2] S. Peng, Y. Liu, Q. Huang, X. Zhou, and H. Bao, "Pvnet: Pixel-wise voting network for 6DoF pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4556–4565.

[3] Y. He, W. Sun, H. Huang, J. Liu, H. Fan, and J. Sun, "Pvn3d: A deep point-wise 3D keypoints voting network for 6DoF pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 11 629–11 638.

[4] V. Lepetit and P. Fua, Monocular Model-Based 3D Tracking of Rigid Objects: A Survey, *Foundations Trends Comput. Graph. Vis.*, vol. 1, pp. 1–89, 2005, doi: 10.1561/0600000001.

[5] T. Drummond and R. Cipolla, "Real-time visual tracking of complex structures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 932–946, Jul. 2002.

[6] A. Kendall, M. Grimes, and R. Cipolla, "Posenet: A. convolutional network for real-time 6-dof camera relocalization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2938–2946.

[7] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox, "Posecnn: A convolutional neural network for 6D object pose estimation in cluttered scenes," *Robot.: Sci. Syst.*, vol. abs/1711.00199, 2018.

[8] M. Oberweger, M. Rad, and V. Lepetit, "Making deep heatmaps robust to partial occlusions for 3D object pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 119–134.

[9] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o(n)solution to the pnp problem," *Int. J. Comput. Vis.*, vol. 81, no. 2, pp. 155–166, 2009.

[10] C. Wang *et al.*, "Densefusion: 6D object pose estimation by iterative dense fusion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3338–3347.

[11] S. Hinterstoisser *et al.*, "Model based training, detection and pose estimation of texture-less 3D objects in heavily cluttered scenes," in *Proc. Asian Conf. Comput. Vis.*, pp. 548–562, 2012.

[12] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother, "Learning 6D object pose estimation using 3D object coordinates," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 536–551.

[13] M. Rad and V. Lepetit, "Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3D poses of challenging objects without using depth," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 3848–3856.

[14] W. Kabsch, "A solution for the best rotation to relate two sets of vectors," *Acta Crystallographica Sect. A: Cryst. Phys., Diffraction, Theor. Gen. Crystallogr.*, vol. 32, no. 5, pp. 922–923, 1976.

[15] H. Su, C. R. Qi, Y. Li, and L. J. Guibas, "Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3D model views," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 2686–2694.

[16] M. Sundermeyer, Z.-C. Marton, M. Durner, M. Brucker, and R. Triebel, "Implicit 3D orientation learning for 6D object detection from rgb images," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 699–715.

[17] B. Tekin, S. N. Sinha, and P. Fua, "Real-time seamless single shot 6D object pose prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 292–301.

[18] S. Zakharov, I. Shugurov, and S. Ilic, "Dpod: 6D pose object detector and refiner," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 1941–1950.

[19] K. Park, T. Patten, and M. Vincze, "Pix2pose: Pixel-wise coordinate regression of objects for 6D pose estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 7667–7676.

[20] A. Aldoma *et al.*, "Cad-model recognition and 6DoF pose estimation using 3D cues," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2011, pp. 585–592.

[21] L. Malleus *et al.*, "Kppf: Keypoint-based point-pair-feature for scalable automatic global registration of large rgb-d scans," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2017, pp. 2495–2502.

[22] C. Li, J. Bai, and G. D. Hager, "A unified framework for multi-view multi-class object pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 254–269.

[23] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox, "Deepim: Deep iterative matching for 6D pose estimation," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 683–698.

[24] F. Manhardt, W. Kehl, N. Navab, and F. Tombari, "Deep model-based 6D pose refinement in rgb," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 800–815.

[25] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

[26] A. Segal, D. Haehnel, and S. Thrun, "Generalized-Icp," *Robot.: Sci. Syst.*, vol. 2, no. 4, p. 435, 2009.

[27] J. Yang, H. Li, D. Campbell, and Y. Jia, "Go-ICP: A globally optimal solution to 3D icp point-set registration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 11, pp. 2241–2254, Nov. 2016.

[28] M. Tamaazousti, V. Gay-Bellile, S. N. Collette, S. Bourgeois, and M. Dhome, "Nonlinear refinement of structure from motion reconstruction by taking advantage of a partial knowledge of the environment," in *Proc. Conf. Comput. Vis. Pattern Recognit.* 2011, pp. 3073–3080.

[29] A. Loesch, S. Bourgeois, V. Gay-Bellile, and M. Dhome, "Generic edgelet-based tracking of 3D objects in real-time," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2015, pp. 6059–6066.

[30] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[31] J. Smisek, M. Jancosek, and T. Pajdla, "3D with kinect," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2011, pp. 1154–1160.

[32] M. Imperoli and A. Pretto, "D2CO: Fast and robust registration of 3d textureless objects using the directional chamfer distance," in *Proc. Int. Conf. Comput. Vis. Syst* Springer, 2015, pp. 316–328.

[33] L. Vacchetti, V. Lepetit, and P. Fua, "Stable real-time 3D tracking using online and offline information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 10, pp. 1385–1391, Oct. 2004.

[34] Y. Hu, J. Hugonot, P. Fua, and M. Salzmann, "Segmentation-driven 6D object pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3380–3389.

[35] Y. Hu, P. Fua, W. Wang, and M. Salzmann, "Single-stage 6D object pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 2927–2936.

[36] T. Hodan *et al.*, "Bop challenge 2020 on 6D object localization," *European Conference on Computer Vision*, pp. 577–594, 2020.