LDFpose: A lightweight deep fusion network for 6D occlusion object pose estimation

Guirong Dong[a,*], Yi Wu[a], Leyu Dai[a], Han Chang[a], Zhaoxun Tang[a] and Dianzi Liu[b]

[a]Faculty of Printing, Packaging Engineering and Digital Media Technology, Xi'an University of Technology, Xi'an, 710054, China
[b]Faculty of Engineering, University of east Anglia, Norwich, NR4 7UJ, The United Kingdom

ARTICLE INFO

Keywords:
6D pose estimation
Cross-modality attention fusion
Embedding-free hierarchical attention
Adaptive hierarchical keypoints sampling

ABSTRACT

Effectively implementing color and depth modalities into the model for accurate 6D object pose estimation in the presence of occlusions and complex backgrounds remains a challenge in the field of robotics vision. To tackle these problems, a novel method LDFpose, a lightweight deep fusion network based on RGB-D input is proposed for real-time and precise object pose estimation in this paper. Unlike existing methods, this developed method effectively captures global semantic information from complex textured backgrounds for the color feature extraction using an Embedding-Free Hierarchical Transformer (EHT) encoder, while significantly reducing computational overhead. In the point cloud feature extraction, an Adaptive Hierarchical Keypoints (AHK) sampling strategy is applied, which employs sparse sampling of distant key points to expand the receptive field and improve the efficiency of point cloud interactions. Furthermore, the model adaptability to heterogeneous inputs and its global representation capability are enhanced by integrating RGB and geometric features in use of the Adaptive Modal Fusion (AMF) block, facilitating more effective complementary data fusion. Experimental results on the datasets including Linemod, YCB-Video, Occlusion Linemod, and a custom dataset of Occlusion of Apes (OccoA) demonstrate that LDFpose outperforms existing methods in pose estimation accuracy and inference speed, particularly in scenarios with different lighting conditions and high degrees of occlusion arising from practical industrial applications.

1. Introduction

Vision-based pose estimation of objects in 6 Degree of Freedoms (6D) involves the calculation of their precise spatial orientations and locations referring to the camera's coordinate system in use of visual features. This plays a crucial role in industrial applications such as autonomous driving [1, 2], robotic arm grasping [3, 4], and augmented reality [5, 6]. 6D pose estimation provides essential scene spatial information by estimating the object's orientation $R(R_x, R_y, R_z)$ and translation $T(T_x, T_y, T_z)$,enabling visual servo systems to perform precise operations. Vision-based pose estimation methods have evolved from non-learning to deep learning-based approaches, as techniques are applied to solve other computer vision tasks [7, 8, 9]. Traditional geometry-based methods, which rely on handcrafted features [10, 11, 12], typically utilize manually designed local feature extraction algorithms to detect key points in images and establish the connections between 2D images and 3D models. Commonly used algorithms for this purpose include SIFT (Scale Invariant Feature Transform) [13], SURF (Speeded-Up Robust Features) [11], FPFH (Fast Point Feature Histograms) [14], and ORB (Oriented FAST and Rotated BRIEF) [12]. However, practical robotic grasping under factory environments, complex backgrounds and inter-object occlusions often face the difficulties, such as the loss or mismatch of key features. Consequently, these challenges increasingly demand a solution to reliable extraction of keypoints and descriptors under significant illumination or viewpoint variations. Moreover, occlusions introduce the increased computational complexity, thereby making it difficult for real-time systems to operate in such scenarios effectively. This ultimately impacts the accuracy and robustness of pose estimation.

Driven by advances in object detection [15, 16], deep-learning networks have been increasingly investigated and utilized for pose estimation tasks. Early developed approaches primarily relied on RGB data to address the detection [17, 18]. Nevertheless, these methods entail reconstructing 3D structural information from 2D image observations, which is fundamentally an ill-posed inverse problem [19]. The non-uniqueness of depth solutions often

*Corresponding author
dongguirong2005@xaut.edu.cn (G. Dong); 2230821141@stu.xaut.edu.cn (Y. Wu); 2250820003@stu.xaut.edu.cn (L. Dai); 2240820040@stu.xaut.edu.cn (H. Chang); 2240820030@stu.xaut.edu.cn (Z. Tang); Dianzi.Liu@uea.ac.uk (D. Liu)
ORCID(s): 0000-0002-8009-3563 (G. Dong); 0009-0009-7978-6678 (Y. Wu); 0000-0003-3390-5327 (D. Liu)

restrains the effectiveness of these techniques from practical applications, particularly in complex scenarios. With the widespread deployment of industrial-grade RGB-D cameras [20, 21], the primary challenge now lies in effectively integrating color and depth information from a single RGB-D image to achieve accurate 6D pose estimation of objects. In early works, a cascade design was used to handle RGB and depth information [22, 23]. This approach initially generated a preliminary estimation from the RGB image, which was subsequently refined using the depth map. In this process, the depth image offers complementary information to the RGB image, thereby aiding in the resolution of scale ambiguity. Also, RGB-D fusion strategies have evolved from basic concatenation or pixel-wise addition [24] to feature-level deep fusion approaches [21, 25, 26, 27, 28]. Classic feature fusion methods [25] focused on late-stage point-wise fusion, thereby enhancing global information. To better combine modality information, some studies considered bidirectional fusion at earlier stages [21], realizing more robust pixel-wise features. Recently, the fusion of information [28] has been improved by leveraging the global semantic similarity between color and geometric features, which helps in tackling issues such as texture loss caused by reflections. However, this method exhibits limitations in distinguishing local-level differentiation, particularly requiring the edge refinement when objects are occluded during robotic grasping. This makes difficult to effectively model the global relationship between RGB and depth data, thereby posing the ongoing challenges for robust 6D pose estimation of objects in open scenes.

With the successful implementation of Transformer into natural language and vision tasks [29, 30], this enables researchers to model long-range dependencies, thereby enhancing RGB-D information fusion. Several studies have employed Transformers for RGB-D feature extraction [31], establishing robust 2D-3D correspondences between RGB images and point clouds [25, 32]. Though these approaches can effectively handle complex environments, Transformer-based architectures generally incur higher computational burden as compared to CNNs. Additionally, keypoints correspondence methods may generate redundant information, further increasing computational cost. Therefore, the development of efficient task inference structures is crucial for real-time applications.

In this work, a lightweight deep fusion network for accurate estimation of 6D object pose under significant illumination or viewpoint variations is proposed. Using the RGB-D inputs, this heterogeneous network processes color and depth data separately through distinct encoder-decoder modulars to realize the accurate pose estimation. First, a lightweight attention module to enhance detail capturing in occluded scenes is developed to capture fine-grained features within locally occluded regions while retaining the global contextual information. To further enhance inference efficiency, non-parametric transformations and average pooling are adopted to spatially compress the key and value features. Following that, the subdivision of 3D space into non-overlapping independent windows is designed to address the challenges of processing point clouds in complex occlusion scenarios. Based on this windowed feature, an adaptive hierarchical keypoints sampling strategy is developed to reduce the computational cost of the transformer blocks and enhance the specificity of point cloud interactions, thereby improving the spatial reasoning abilities of the model in occluded scenes.

2. Brief review of 6D pose estimation

2.1. RGB-based 6D Pose Estimation

Methods estimating object pose using single RGB images by the extraction of salient features are usually assisted by PnP (Perspective-n-Points)/RANSAC (Random Sample Consensus) [17, 33, 34]. However, these approaches are not capable of achieving differentiable pose estimation. Recently, some methods directly predict the dense correspondences instead of sparse ones [35, 36]. For example, PoseCNN [23] decomposed the task into three distinct components, enabling the network to explicitly model their dependencies and independencies, and introduced ShapeMatch-Loss for more accurate pose estimation. Based on this, DeepIM [37] tackled the challenge of inaccurate RGB-based pose estimation by integrating a holistic loss function that progressively refined pose accuracy. The efficient pose [38] was further advanced by the state-of-the-art techniques, such as extending the 2D object detector (EfficientDet) to the capability of 6D pose estimation. However, due to the non-linearity of rotation space, solutions to directly estimating object pose using RGB images remains suboptimal, particularly when objects are partially occluded, the sole use of RGB features is insufficient to infer the complete pose of the object.

2.2. RGB-D Fusion-based 6D Pose Estimation

Traditional methods [39, 40] directly convert depth maps into point clouds and use geometric relationships to compute the 3D coordinates of, each pixel. These methods then apply conventional point cloud processing techniques for 3D object detection. With the introduction of PointNet [41] and PointNet++ [27], researchers began to extract

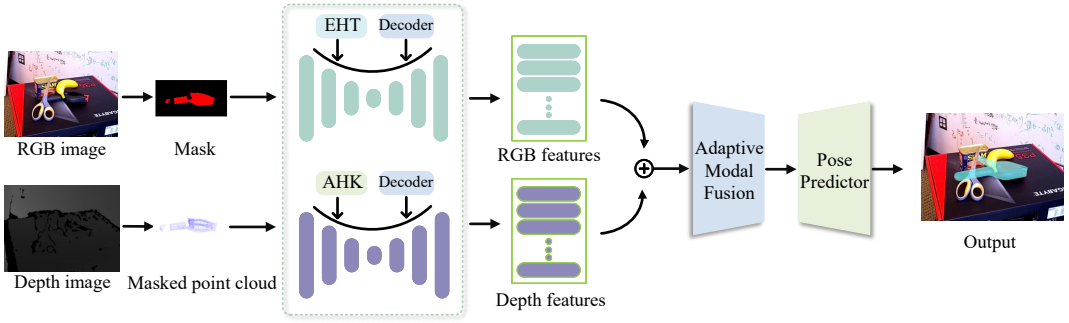

Figure 1: Overview of LDFpose implemented by the feature concatenation operation.

geometric features directly from raw point clouds [22], while RGB data was processed via Convolutional Neural Networks (CNNs). The features from both modalities were then fused in subsequent network layers. Recently, some studies have adopted a dual-branch structure to integrate the texture and shape features for point-wise fusion [25, 42, 43]. Though this method enhances the discriminative power, it remains relatively sensitive to local noise and fails to fully exploit the complementary nature between the two modalities.

2.3. Transformer Applied in 6D Pose Estimation

Transformers have become the dominant architecture in the field of natural language processing [44], enabling the self-supervised training on large-scale datasets. Compared to CNNs [45, 46, 47], Transformers demonstrate the superior efficiency in capturing global dependencies. Recently, this architecture has also been widely applied in computer vision tasks [48, 49, 50]. In the context of 6D pose estimation, the Trans6D [51] method developed a hybrid architecture capable of handling RGB images. However, due to the absence of absolute scale information, its robustness was compromised in complex scenes. Similarly, YOLOPose [52] proposed a novel single-stage model based on the Transformer that directly regressed 2D keypoints, thereby improving both the speed and accuracy in simple scene applications. FusionNet [53] proposed a CNN-Transformer hybrid model that primarily predicts pose estimation using dense 2D-3D correspondences of points. Although FusionNet can be applied for pose estimation in complex scenes, it encounters the issue to extract sufficient features when the scene is sparse or keypoints are missing, therefore leading to the decreased accuracy of pose estimation. The above issues highlight that further optimization and enhancement are necessary to effectively address the challenging occlusion scenarios, even Transformers have demonstrated significant potentials in solving vision-based tasks.

3. The proposed framework for 6D pose estimation considering occlusion

The proposed framework shown in Figure 1 is mainly designed to determine the transformation matrix of an object relative to the camera coordinate system using the given RGB-D images. The matrix consists of a rotation component $R \in \mathrm{SO}(3)$ defined in a three-dimensional space and a translation component $t \in \mathbb{R}^3$ for the alignment of a known rigid object in its object coordinate system to the camera coordinate system. To effectively capture global multi-source and multi-scale feature representations, the initial image I_r and point cloud P are first processed through two lightweight, independent encode-decode modules (E_r and E_p) for feature extraction. These extracted features are then fused using an adaptive modal fusion module to enable robust interaction between modalities. Finally, the object pose is predicted using a simple layer. Specifically, the prediction model is mathematically expressed by:

$$F_r = E_r(I_r), \quad F_p = E_p(P) \tag{1}$$

$$F_m = \Phi(F_r, F_p; \theta) \tag{2}$$

$$O(R,T) = \Psi(F_m) \tag{3}$$

where F_r and F_p are the extracted initial features; Φ means the adaptive fusion module parameterized by θ; F_m presents the fused multi-modal feature; Ψ denotes the regression model and $O(R,T)$ is defined as the output of object pose.

3.1. Embedding-Free Hierarchical Transformer

In multi-modal 6D pose estimation networks, the RGB feature extraction is usually realized by network architectures for addressing 2D semantic segmentation tasks. To meet the specific demands on the real-time 6D pose estimation under significant illumination or viewpoint variations, an Embedding-free Hierarchical Transformer (EHT) encoder-decoder module is designed, as shown in Figure 2. The proposed model is represented by a hierarchical architecture composed of four stages (from Stage 1 to Stage 4), each of which integrates a patch embedding or patch merging sets and EHT Blocks. The patch embedding set partitions the input image into patches and converts them into embedding vectors, while the patch merging set progressively merges adjacent patches in the following stages to reduce spatial resolution and expand receptive fields. As the input passes through each stage, the spatial dimensions of the feature maps are decreased progressively by 1/4, 1/8, 1/16, and 1/32, while the channel numbers are increased by 80, 160, 320 and 640 accordingly. This hierarchical design enables the framework to extract both high-level and low-level features from the input data, therefore effectively capturing local and global contextual information. The EHT Blocks in each stage further process these embedded patches, utilizing the hierarchical transformer architecture to enhance feature representation and model expressiveness, thereby enabling robust feature learning across different scales.

Vanilla Transformer module relies on independent linear transformation layer to generate query, key, and value embeddings, therefore introducing additional parameters and local biases as follows:

$$Q = W_q X, \quad K = W_k X, \quad V = W_v X \tag{4}$$

where $W_q, W_k, W_v \in \mathbb{R}^{d\times d}$ are learnable parameter matrices and X represents the input features.

Inspired by [54], the developed EHT (Embedding-Free Hierarchical Transformer) mitigates the reliance on linear embeddings by directly dealing with raw input features $X \in \mathbb{R}^{N\times d}$, enabling lightweight deep models while avoiding interference with global information caused by the embedding layers. Moreover, preprocessing raw features by front-end max-pooling is denoted as X_{pregd}. This preprocessing step facilitates global feature pre-guidance and activates the attention mechanism, thereby enabling it to focus more effectively on enhancing global modeling capabilities.

$$X_{\text{pregd}} = \text{MaxPool}(X) \in \mathbb{R}^{N/s\times d} \tag{5}$$

To realize globally-guided input features, the FSR (Flexible Spatial Reduction) module is employed to adaptively compress the resolution of non-critical regions for optimal allocation of computational resources, better model accuracy, and significant reduction of computational complexity. It can be mathematically defined as:

$$X' = F_{\text{FSR}}(X_{\text{pregd}}; \mu) \tag{6}$$

where μ is learnable compress parameters.

Subsequently, the compressed features X' are partitioned into $m \times m$ non-overlapping windows $\{W_i\}_{i=1}^{m^2}$, within which the local self-attention is computed to capture fine-grained details of occluded regions by the local attention blocks using Eq. (7).

$$F_{\text{local}}(W_i) = \text{softmax}\left(\frac{Q_l K_f^T}{\sqrt{d_k}}\right) V_f \tag{7}$$

Simultaneously, key patches within each window are extracted by ranking the top-K high-response regions on the basis of the sum of attention weights, as these weights represent the salient content of the window through aggregation of local semantic core information by Eq. (8).

$$\mathcal{P}_{\text{key}} = \bigcup_{i=1}^{m^2} \text{TopK}(\{W_i\}, K) \tag{8}$$

Then, these key patches from all windows are concatenated into a global sequence $S_{\text{global}} \in \mathbb{R}^{km^2\times d}$ and fed into a global attention block for modelling the cross-window interactions, thereby possessing the capability of capturing long-range dependencies and comprehensive scene context. This could be expressed by

$$F_{\text{global}}(S_{\text{global}}) = \text{softmax}\left(\frac{Q_g K_g^T}{\sqrt{d_k}}\right) V_g \tag{9}$$

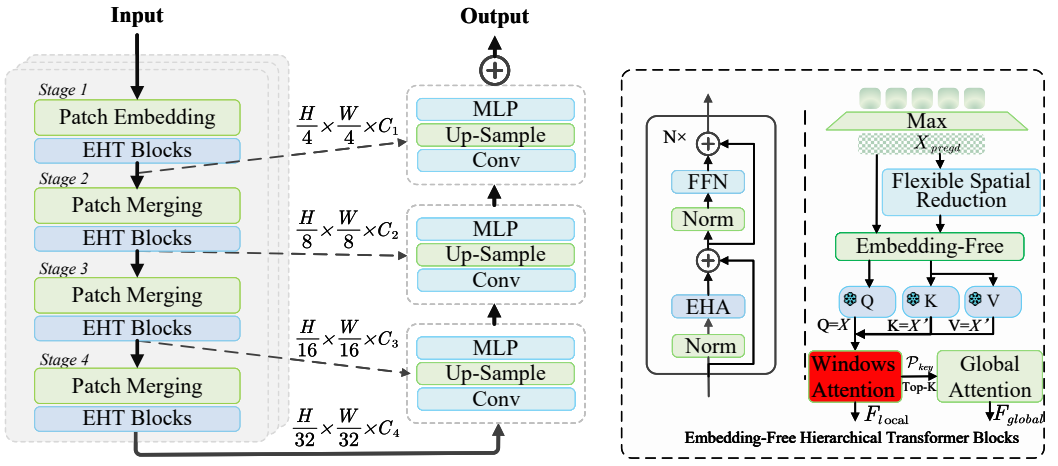

Figure 2: The overall architecture of the Embedding-Free Hierarchical Transformer Module.

Finally, the outputs of local window attention and global attention are adaptively fused via learnable weighting parameters $\alpha \in [0, 1]$, achieving the optimal balance between local detail enhancement and global semantic integration by Eq. (10).

$$F_{\text{rgb}} = \alpha \cdot F_{\text{local}} + (1 - \alpha) \cdot F_{\text{global}} \tag{10}$$

The above procedure of EHT approach leverages hierarchical attention mechanisms to preserve multi-scale feature representations, enabling the model to capture both fine-grained details in occluded regions and global contextual information used for 6D pose estimation. With the reduction of computational overhead through lightweight designs, EHT ensures real-time inference while empowering the model's ability to distinguish object semantics from complex backgrounds.

3.2. Flexible Spatial Reduction Method

The conventional spatial resolution compression method for attention mechanisms employs the fixed reduction parameters or predefined dynamic spatial compression strategies [46, 23]. The input features $X \in \mathbb{R}^{h \times w \times c}$ are downsampled using a compression ratio μ to achieve the reduced spatial resolution and denser feature representations $X' \in \mathbb{R}^{h/\mu \times w/\mu \times c}$. However, this compression approach fails to identify crucial features, leading to indiscriminate compression of both critical and non-critical regions, which inevitably causes loss of high-frequency details. To address this issue, the Flexible Spatial Reduction (FSR) module is proposed to perform the post-K and V-generation compression ranked by attention-aware importance scores. The FSR adaptively adjusts spatial resolution after generating K and V for task-aware computational resource allocation.

First, the FSR module calculates spatial importance scores s_{ij} for each position (i, j) and head k of the generated K and V by aggregating attention weights from the buffered attention map, defined in Eq. (11).

$$s_{ij} = \frac{1}{N} \sum_{k=1}^{N} A_{ij}^{k} \tag{11}$$

where A_{ij}^{k} is the attention weight between the input query q_k and key k_{ij}.

A dynamic threshold β is then defined to divide regions into critical ($s_{ij} \geq \beta$) and non-critical ($s_{ij} < \beta$) categories for resolution adjustment. Non-critical regions are processed for resolution reduction via adaptive pooling with compression ratios μ_{ij} between 1 and $\mu_{\max}$, while critical regions retain full resolution intact. Finally, the compressed non-critical features and uncompressed critical features are concatenated to form the output K_f and V_f for the purpose of optimal

computational efficiency while preserving task-relevant spatial details, mathematically expressed by Eqs. (12-15) as follows:

$$\beta = \alpha \cdot \text{mean}(s_{ij}) + (1-\alpha) \cdot \text{median}(s_{ij}) \tag{12}$$

$$\mu_{ij} = \text{Clip}\left(\left[\frac{(\beta - s_{ij})}{\beta} \cdot \mu_{\max}\right], 1, \mu_{\max}\right) \tag{13}$$

$$K_{\text{compressed}} = \text{Avgpool}_{\mu_{ij}}(K_{nc}) \tag{14}$$

$$K_f = \text{Concat}(K_{nc}, K_c) \tag{15}$$

3.3. Adaptive Hierarchical Keypoints Sampling

For point cloud feature extraction, similar to the working mechanism of EHT, an Adaptive Hierarchical Keypoints Sampling (AHK) strategy is applied to combine the local and global contextual information while considering the features of 3D geometric data as shown in Figure 3. AHK partitions the feature space into non-overlapping windows to enable the localized feature interactions and introduces the ability of hierarchical samplings to capture the long-range dependencies. It should be noted that EHT realizes uniform window samplings for 2D visual semantics and AHK adaptively performs the sampling density for point clouds to avoid their irregularity and sparsity. Therefore, AHK sparsely takes samples of distant keypoints to expand the receptive field and densely selects points in local regions for fine details. This design not only reduces computational overhead by avoiding redundant point interactions but also enhances the model's spatial reasoning ability in occluded scenes.

For a given point cloud $P = \{p_i \in \mathbb{R}^3 | i = 1, 2, \dots, N\}$, the depthwise separable convolution structure of PointNeXt [59] is employed to extract local geometric features F_{init}. These initial features capture the local shape and neighborhood relationships of each point, serving as the foundation for subsequent geometric complexity analysis. Specifically, for each point p_i, its k-nearest neighbor set $N(p_i)$ is computed, and the curvature is estimated via principal component analysis (PCA) as follows:

$$C_i = \frac{1}{|N(p_i)|} \sum_{p_j \in N(p_i)} (p_j - \bar{p}_i)(p_j - \bar{p}_i)^T \tag{16}$$

$$C_{\text{curv}}(p_i) = \sigma^2\left(\left\{\frac{\lambda_3}{\lambda_1 + \lambda_2 + \lambda_3} \middle| p_j \in \mathcal{N}(p_i)\right\}\right) \tag{17}$$

where $\bar{p}_i$ depicts the centroid of the area constructed by the neighboring points, C_i is decomposed into $(\lambda_1, \lambda_2, \lambda_3)$ by the eigenvalue decomposition.

Meanwhile, the density change rate of its domain points can be calculated by Eqs. (18-20):

$$\rho(p_i) = \frac{|\mathcal{N}(p_i)|}{V_i} \tag{18}$$

$$\Delta_\rho(p_i) = \frac{1}{|\mathcal{N}(p_i)|} \sum_{p_j \in \mathcal{N}(p_i)} \left|\frac{\rho(p_j) - \rho(p_i)}{\rho(p_i) + \epsilon}\right| \tag{19}$$

$$C_{\text{dens}}(p_i) = \frac{\Delta_\rho(p_i)}{\max_{p_j \in \mathcal{N}(p_i)} \Delta_\rho(p_j) + \epsilon} \tag{20}$$

where V_i is approximated as $V_i = \frac{4\pi}{3} \max_{p_j \in N(p_i)} \|p_j - p_i\|^3$.

Combining the curvature variance and density change with the weighted fusion, the final geometric complexity is mathematically defined by

$$C(p_i) = w_1 \cdot C_{\text{curv}}(p_i) + w_2 \cdot C_{\text{dens}}(p_i) \tag{21}$$

and the adaptive window size associated with the geometric complexity is determinate by Eq. (22):

$$s^i_{\text{win}} = s_{\text{base}} \cdot \exp\left(\alpha \cdot \frac{C(p_i) - \mu}{\sigma}\right) \tag{22}$$

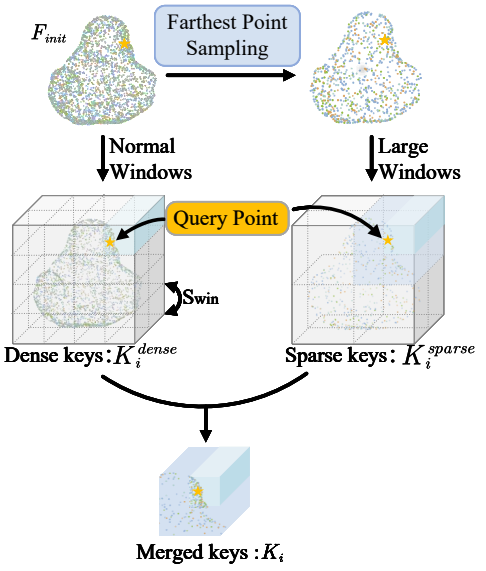

Figure 3: An illustration of the Adaptive Hierarchical Keypoints Sampling strategy with yellow star representing the given query point.

where s_{base} means the base window size; α is the adjustment factor; μ and σ represent the complexity mean and standard deviation, respectively.

Also, a dense set of keypoints is sampled based on the geometric complexity density in the adaptive window s^i_{win} by

$$D(p_j) = \exp(\beta \cdot C(p_j)) \tag{23}$$

$$K_i^{\text{dense}} = \{p_j | p_j \in \mathcal{N}_{s^i_{\text{win}}}(q_i) \wedge D(p_j) = d(F_j)\} \tag{24}$$

where τ is the sampling threshold.

Then, the point cloud is downsampled by Farthest Point Sampling (FPS) and also, the sparse sampling is performed in the extended window by Eq. (25):

$$K_i^{\text{sparse}} = \{p_j | p_j \in \mathcal{N}_{s^{\text{large}}_{\text{win}}}(q_i) \wedge D(p_j) = d(F_j)\} \tag{25}$$

where $s^{\text{large}}_{\text{win}} = k \cdot s^i_{\text{win}}$ is obtained.

The query points (Q_i) are used to compute attention score with its keypoints sampling set K_i followed by feature aggregation using Eqs. (26-27):

$$K_i = K_i^{\text{dense}} \cup K_i^{\text{sparse}} \tag{26}$$

$$F_{\text{geo}} = \text{softmax}\left(\frac{Q_i K_i^T}{\sqrt{d_k}}\right) V_i \tag{27}$$

It is worth noting that all geometric computations in the AHK module, including K-nearest neighbor search, covariance matrix construction, eigenvalue decomposition, and density change rate estimation, are fully implemented in parallel on GPU. Specifically, the KNN search is performed using batched CUDA-based nearest neighbor operations with a fixed small neighborhood size k. The covariance matrices are of size 3×3, and their eigen-decomposition is computed using efficient batched symmetric solvers. As each PCA operation involves only low-dimensional matrices, the computational overhead is negligible on modern GPUs. Moreover, all curvature and density computations are

vectorized and executed in a fully tensorized manner without iterative CPU-GPU data transfers. The time consumption of the entire AHK module is included in the inference time in Figure 8. Specifically, quantitative profiling shows that the AHK module consumes only 2.6 ms per frame out of the 40 ms total inference time, representing less than 7% of the computational load. This confirms that AHK accounts for only a small fraction of the total runtime, while the dominant cost remains in backbone feature extraction and pose regression. Therefore, the proposed geometric complexity analysis does not cause a computational bottleneck and remains consistent with the real-time objective of the LDFpose framework.

3.4. Adaptive Modal Fusion

The effective fusion of modal features from RGB images and point clouds is critical for handling occlusion and complex scenes in the field of machine vision. After feature extraction by EHT and AHK modules, Adaptive Modal Fusion (AMF) block is developed to address the issue of cross-modal information integration, as shown in Figure 4. With the RGB feature map $F_{\text{rgb}} \in \mathbb{R}^{\hat{H}\times\hat{W}\times C}$ and the geometric feature map $F_{\text{geo}} \in \mathbb{R}^{\hat{N}\times C}$ at the i-th layer, modality interaction between these two branches is initially considered by the cross-attention module. Following that, the transformer-based module is utilized to model these two modalities data in a coarse way. Specifically, feature maps F_{rgb} and F_{geo} are flattened into sequences. And then, max-pooling is applied to generate global features $F_{\text{rgb}}^{\max} \in \mathbb{R}^{1\times C}$ and $F_{\text{geo}}^{\max} \in \mathbb{R}^{1\times C}$. To compute the Q, K, and V values, the queries for each modality come from the opposite modality, while the keys and values come from the respective modality. This interaction scheme merges the information of geometric and color features with the transformation formulated as follows:

$$(Q^{\max}, K, V) = F^{\max}(W_q^{\max}, W_k, W_v) \tag{28}$$

where $W_q^{\max}$, W_k, W_v are learnable projection matrices, and $F^{\max}$ represents the pooled feature obtained from $F_{\text{rgb}}^{\max} - F_{\text{geo}}^{\max}$.

The output feature $F_{\text{out}}^{\text{rgb}}$ and $F_{\text{out}}^{\text{geo}}$ are then refined through cross-attention defined by Eqs. (29-30):

$$F_{\text{out}}^{\text{rgb}} = \text{MultiHead}(Q_{\text{rgb}}, K_{\text{geo}}, V_{\text{geo}}) = \text{concat}\left(\sigma\left(\frac{Q_i^{\max}(K_i)^T}{\sqrt{d_k}}\right)V_i \middle| i = 1, 2 \dots h\right) W^O \tag{29}$$

$$F_{\text{out}}^{\text{geo}} = \text{MultiHead}(Q_{\text{geo}}, K_{\text{rgb}}, V_{\text{rgb}}) = \text{concat}\left(\sigma\left(\frac{Q_i^{\max}(K_i)^T}{\sqrt{d_k}}\right)V_i \middle| i = 1, 2 \dots h\right) W^O \tag{30}$$

where σ represents the softmax normalization, i is the number of attention heads, and W^O is denoted as the output projection matrix. It is noted that the output feature F_{out} is achieved by the concatenation with the features $F_{\text{rgb-geo}}$ and $F_{\text{geo-rgb}}$, which are obtained by the addition of the original features F_{rgb} and F_{geo} at the pixel level and cross-modality features $F_{\text{out}}^{\text{rgb}}$ and $F_{\text{out}}^{\text{geo}}$, shown by Eq. (32).

$$F_{\text{rgb-geo}} = F_{\text{rgb}} + F_{\text{out}}^{\text{rgb}}, \quad F_{\text{geo-rgb}} = F_{\text{geo}} + F_{\text{out}}^{\text{geo}} \tag{31}$$

$$F_{\text{fusion}} = F_{\text{rgb-geo}} \oplus F_{\text{geo-rgb}} \tag{32}$$

where $+$ denotes the element-wise addition, and $\oplus$ represents the direct concatenation.

After the cross-attention process, the obtained features are fed into a transformer-based module, resulting in the final fused feature F_{fusion} after the ith attention block shown in Fig.5:

$$F_{\text{fusion}}^{i} = \text{MLP}\left(\text{Norm}\left(\text{MSA}(F_{\text{fusion}}^{i-1})\right)\right) + F_{\text{fusion}}^{i-1} \tag{33}$$

3.5. Pose estimation and loss function

With the cross-modal features F_{fusion} by the AMF block, the pose estimation module transforms these integrated RGB-depth representations into 6D pose parameters (R, t, c) through a Geometric Projection Layer (GPL). GPL processes F_{fusion} using two parallel branches: a Pose Regression Branch that is composed of a 3-layer MLP with group

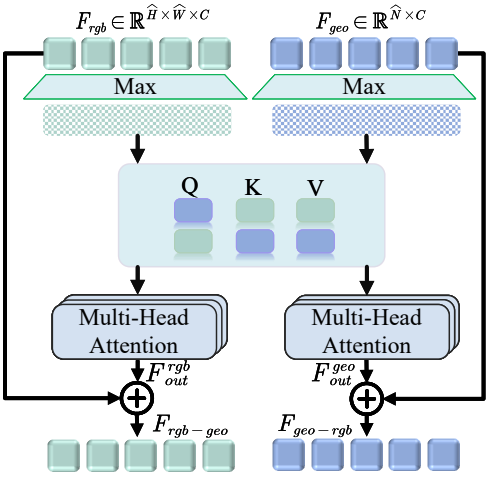

Figure 4: An illustration of the Adaptive Modal Fusion block.

normalization maps each pixel-wise feature to a 6D pose vector parameterized by a unit quaternion R (to avoid gimbal lock), camera coordinates t and a Confidence Map Branch using a convolutional layer to generate a reliability score $c \in [0, 1]$ for each pixel to suppress noisy predictions in occluded regions. It is noted that the unit quaternion R is defined to avoid gimbal lock. To handle symmetric objects, the module integrates a nearest-point matching strategy within a loss function balancing the pose error L_{add} ($L_{\text{add-s}}$ for symmetry-aware scenarios) with confidence regularization, encouraging the model to assign low scores to ambiguous regions by leveraging the hierarchical and cross-modal nature of F_{fusion} to prioritize geometrically consistent features.

$$L_{\text{add}} = \frac{1}{M} \sum_{p_j \in O} \left\| (\hat{R}p_j + \hat{t}) - (R^* p_j + t^*) \right\|_2 \tag{34}$$

$$L_{\text{add-s}} = \frac{1}{M} \sum_{p_j \in O} \min_{p_k \in O} \left\| (\hat{R}p_j + \hat{t}) - (R^* p_k + t^*) \right\|_2 \tag{35}$$

$$L = \frac{1}{N} \sum_{i=1}^{N} \left[\hat{c}_i \cdot L_{\text{add/s}}^i - \lambda \log(\hat{c}_i) \right] \tag{36}$$

where $O = \{p_1, p_2, \ldots, p_M\}$ is the sampling point set of the 3D object model; $\hat{R}, \hat{t}$ are the predicted values; R^*, t^* are denoted as the ground-truth values; and N represents the number of sample points.

A weighted voting mechanism is applied to aggregate predictions by clustering poses using cosine similarity for rotations and Euclidean distance for translations. Finally, the final pose $(R_{\text{final}}, t_{\text{final}})$ is computed to measure the consistency between predicted and ground-truth poses using the confidence-weighted mean of cluster centers.

$$R_{\text{final}} = \underset{R}{\text{argmin}} \sum_{k=1}^{K} w_k \cdot \text{dist}(R, R_k) \tag{37}$$

$$t_{\text{final}} = \frac{\sum_{k=1}^{K} w_k \cdot t_k}{\sum_{k=1}^{K} w_k} \tag{38}$$

where $w_k = \sum_{i \in k} \hat{c}_i$ is the total confidence level of the kth cluster.

4. Experimental results

4.1. Validation on four datasets

To validate the correctness and feasibility of the proposed framework for 6D pose estimation under the condition of object occlusion, three existing datasets including Linemod [55], Occlusion Linemod [56], and YCB-Video [23] are examined in this study. Moreover, a user-defined OccoA dataset is developed to evaluate the network generalization considering the complex working environments in robot grasping processes. A brief description of each dataset is introduced as follows: **Linemod Dataset**, a well-known benchmark, contains over 18,000 real images, RGB-D videos and 3D models for 13 objects. Each image is annotated with ground-truth 6D pose information and reflects on the inherent features including cluttered scenes, textureless objects, and diverse lighting conditions in pose estimation tasks. **Occlusion Linemod Dataset** is an augmented library based on the Linemod dataset, specifically designed for realizing 6D pose estimation tasks under the consideration of challenging conditions of object occlusion. There are 8 objects originated from the Linemod dataset and each of them is augmented by varying degrees of occlusion across diverse scenes. **YCB-Video** contains 21 objects and over 130,000 frames of video sequences from real-world scenes. Each frame includes RGB-D data and the detailed annotations of the objects in the scene. Also, this dataset incorporates synthetic data to facilitate the network training process. **OccoA** is a custom dataset which contains 1,000 real images covering 12 distinct objects with different sizes and shapes. These objects are subjected to various levels of occlusion by an ape. Moreover, to validate the generalization ability of the model, the scene range is expanded. The 1,000 images in the dataset were captured under diverse backgrounds, including both indoor and outdoor scenes, and lighting conditions ranging from natural to dim lighting.

4.2. Evaluation Metric

To evaluate the proposed LDFpose framework, three metrics are employed: the Average Distance of Model Points (ADD), the Average Closest Point Distance (ADD-S), and the ADD(S), where ADD(S) computes the ADD distance for non-symmetric objects and the ADD-S distance for symmetric objects. The primary difference between ADD and ADD-S lies in the way to handle objects with varying shapes. Specifically, ADD is more effective for asymmetric objects, and its mathematical formulation is defined as follows:

$$\text{ADD} = \frac{1}{M} \sum_{p \in \mathcal{O}} \left\| (Rp + t) - (R^* p + t^*) \right\| \tag{39}$$

where $[R, t]$ represents the estimated pose, and $[R^*, t^*]$ denotes the ground-truth pose. p is one of the M points in the set $\mathcal{O}$ represented by 3D model points.

As the ADD-S metric suits for the evaluation of symmetric objects and effectively addresses the ambiguity caused by symmetry, it calculates the distance between each predicted point and its nearest corresponding ground-truth point using Eq. (40) as follows:

$$\text{ADD-S} = \frac{1}{M} \sum_{p_1 \in \mathcal{O}} \min_{p_2 \in \mathcal{O}} \left\| (Rp_1 + t) - (R^* p_2 + t^*) \right\| \tag{40}$$

where p_1 and p_2 are the transformed result of the set $\mathcal{O}$ represented by 3D model points under the estimated pose $[R, t]$ and the ground-truth pose $[R^*, t^*]$, respectively.

Followed by the previous research [25], for YCB-Video dataset, the area under the accuracy-threshold curve obtained by varying the distance threshold (ADD-S AUC and ADD(S) AUC) is reported. For the Linemod and Occlusion Linemod datasets, the accuracy of distance less than 10% of the objects diameter (ADD-0.1d) is reported. For the custom dataset (OccoA), accuracy is evaluated using both the ADD(S) AUC and the 2 cm metric. Moreover, to provide a more comprehensive assessment of 6D pose accuracy, the proposed method on Occlusion Linemod and YCB-Video is evaluated using the BOP benchmark metrics: Visible Surface Discrepancy (VSD), Maximum Symmetry-Aware Surface Distance (MSSD), and Maximum Symmetry-Aware Projection Distance (MSPD). These metrics account for visible surface alignment and object symmetry, offering complementary evaluation to ADD-based metrics.The AUC represents the area under the ADD-S curve, while the 2 cm metric quantifies the percentage of ADD-S values less than 2 cm.

Table 1
Quantitative evaluation using the ADD-0.1d metric on the Linemod Dataset.

Method	PVNet [46]	DPOD [35]	PointFusion [26]	DF [25]	G2L-Net [57]	PVN3D [20]	FFB6D [21]	DFTr [28]	LDFpose
Ape	43.6	87.7	70.4	92.3	96.8	97.3	98.4	98.6	98.9
Benchvise	99.9	98.5	80.7	93.2	96.1	99.7	100.0	100.0	100.0
Camera	86.9	96.1	60.8	94.4	98.2	99.6	99.9	100.0	99.7
Can	95.5	99.7	61.1	93.1	98.0	99.5	99.8	100.0	99.8
Cat	79.3	94.7	79.1	96.5	99.2	99.8	99.9	100.0	99.6
Driller	96.4	98.8	47.3	84.0	99.8	99.3	100.0	100.0	99.7
Duck	52.6	86.3	63.0	92.3	97.7	99.2	98.4	99.1	98.4
Eggbox	99.2	99.9	99.9	99.8	100.0	99.8	100.0	100.0	100.0
Glue	95.7	96.8	99.3	100.0	100.0	100.0	100.0	100.0	99.8
Hole	82.0	86.9	71.8	92.1	99.0	99.9	99.8	100.0	100.0
Iron	98.9	100.0	83.2	97.0	99.3	99.7	99.9	99.9	99.7
Lamp	99.3	96.8	62.3	95.3	99.5	99.8	99.9	100.0	99.8
Phone	92.4	94.7	78.8	92.8	98.9	99.5	99.7	99.6	99.6
Mean	86.3	95.2	73.7	94.3	98.7	99.4	99.7	99.8	99.6

4.3. Implementation Details

Through the experiment, a single RTX4090 GPU is utilized for model training and evaluation. The entire processing is divided into two steps: pre-segmentation by Yolo11 and pose estimation. First, a pre-trained model built by Yolo11 is employed to perform object segmentation on the RGB images and generate masks that are converted into point clouds. Second, to realize the color pixel-level feature extraction using an Embedding-Free Hierarchical Transformer (EHT) in the pose estimation stage, the number of encoder layers and attention heads are set to 4 and 16, respectively. For the point cloud feature extraction, Adaptive Hierarchical Transformer (AHT) scheme for point sampling includes the encoder layers and attention heads, which are set to 4 and 8, respectively. In this study, AHT sampling on the raw input points with a grid size of 0.04 m is conducted. In the training process, the maximum number of input points is set to $80,000$ and the window size is initially set to 0.16 m, which is then doubled after each downsampling layer with the ratio of 8. The final fused RGB-D features are input into the head network containing a shared MLP, to output the predicted pose and confidence for each point. The pose with the highest confidence is determined using an argmax operation.

4.4. Result comparisons with State-of-the-Art approaches

To verify the correctness and robustness of the proposed framework for pose prediction under inter-object occlusion situations, the predictions by LDFpose are compared with those by state-of-the-art approaches on four datasets.

4.4.1. Evaluation on the Linemod dataset

Table 1 reports the ADD-0.1d results with a mean accuracy of 99.6% on the Linemod dataset using the proposed method. Also, results demonstrate that the proposed framework outperforms RGB methods by 13.3% and 4.4% higher than PVNet and DPOD, respectively. As compared to current popular RGB-D methods, results by LDFpose are comparable to those by FFB6D and DFTr, which incorporate 3D keypoint detection heads to enhance pose estimation accuracy of the model. Overall, the lightweight LDFpose significantly improves the network efficiency and reduces computational overhead, enabling it more suitable to address real-time grasping by industrial robots in the online detection process.

4.4.2. Evaluation of the Occlusion Linemod dataset

The same training strategy and evaluation metrics are used to evaluate the prediction accuracy on the Occlusion Linemod dataset. Results among LDFpose, PVNet, HybridPose, PVN3D, PR-GCN, FFB6D, and DFTr, are compared and shown in Table 2. The ADD-0.1d metric value of LDFpose is 79.0, which is increased by 1.3% than that of DFTr. This accuracy improvement demonstrates that the proposed framework exhibits the stronger competitiveness in occluded scenes, as compared to results obtained under the regular scenario (Linemod). The environmental setting of the

Table 2
Quantitative evaluation using the ADD-0.1d metric on the Occlusion Linemod Dataset.

Method	PoseCNN [23]	HybridPose [58]	PVN3D [20]	PR-GCN [59]	FFB6D [21]	DFTr [28]	LDFpose
Ape	9.6	20.9	33.9	40.2	47.2	64.1	66.3
Can	45.2	75.3	88.6	76.2	85.2	96.1	93.8
Cat	0.9	24.9	39.1	57.0	45.7	52.2	56.2
Driller	41.4	70.2	78.4	82.3	81.4	95.8	95.8
Duck	19.6	27.9	41.9	30.0	53.9	72.3	77.4
Eggbox	22.0	52.4	80.9	68.2	70.2	75.3	75.8
Glue	38.5	53.8	68.1	67.0	60.1	79.3	80.2
Hole puncher	22.1	54.2	74.7	97.2	85.9	86.8	86.8
Mean	24.9	47.5	63.2	65.0	66.2	77.7	79.0

Figure 5: Qualitative results on the Occlusion Linemod dataset.

Occlusion Linemod dataset is described in Figure 5. As shown in Table 3 ,Compared with GDRNPP, LDFpose further improves MSPD from 87.2 to 87.6, suggesting better 2D projection alignment. Overall, the proposed framework achieves stronger visible-surface alignment while maintaining competitive geometric consistency, validating the effectiveness of global semantic modeling and adaptive hierarchical keypoints sampling strategy.The boundary profile highlighted in green, that is a shape of an ape, represents the segmented mask, while its orientation is defined as the predicted pose. It can be observed that under different viewpoints and various degrees of occlusion, sufficient feature information are captured by the proposed framework to restore the correct pose information. Quantitative results considering occlusion of objects demonstrate that LDFpose exhibits robust resistance to geometric occlusion and photometric interference, enabling it to maintain better pose estimation in dynamic environments.

Table 3
Comparison of VSD, MSSD, and MSPD on Occlusion Linemod.

	VSD	MSSD	MSPD
FFB6D-CVPR21-PBR-NoRefinement [21]	55.0	71.8	79.2
RCVPose 3D SingleModel VIVO PBR [60]	72.0	72.3	74.5
GDRNPP-PBR-RGBD-MModel [61]	62.9	82.2	87.2
LDFpose	69.4	81.6	87.6

Table 4
Quantitative evaluation results (ADD-S and ADD(S) AUC) on the YCB-Video Dataset.

Method	ADD-S (%)	ADD(S) (%)
PoseCNN [23]	75.8	59.9
PointFusion [26]	83.9	-
DCF [62]	85.7	77.9
DF(per-pixel) [25]	91.2	82.9
PVN3D [20]	95.5	91.8
PR-GCN [59]	95.8	-
FFB6D [21]	96.6	92.7
DFTr [28]	96.7	94.4
LDFpose	96.6	94.1

Table 5
Comparison of VSD, MSSD, and MSPD on YCB-Video.

	VSD	MSSD	MSPD
FFB6D-CVPR21-PBR-NoRefinement [21]	70.6	82.7	74.0
RCVPose 3D SingleModel VIVO PBR [60]	84.1	84.6	84.3
GDRNPP-PBR-RGBD-MModel [61]	87.5	94.9	89.5
LDFpose	86.8	95.6	90.1

4.4.3. Evaluation using the YCB-Video dataset

To validate the effectiveness of the proposed framework for 6D pose estimation, the ADD-S and ADD(S) metrics are calculated for performance evaluation using the YCB-Video dataset. As illustrated in Table 4, LDFpose has the ability to achieve an ADD-S metric value of 96.6%, which indicates its performance as great as those state-of-the-art methods and significantly surpasses PR-GCN and PVN3D. Specifically, its ADD(S) value is notably higher than most other methods, while it is slightly lower by 0.6% than the best result achieved by DFTr. This marginal gap is attributed to the weighted voting mechanism adopted in DFTr, by which fine-grained pose refinement is enhanced and robustness under complex geometric variations is improved. A more transparent analysis of the accuracy-efficiency trade-off is provided by further investigating the influence of the FSR module. While non-critical spatial regions are effectively compressed by FSR to reduce computational overhead, high-frequency texture cues for certain small-scale or highly textured objects in the YCB-Video dataset may be attenuated by its inherent downsampling operation. This effect is governed by the dynamic threshold β, which controls the balance between feature preservation and inference speed. When the β is configured to prioritize higher efficiency, more aggressive spatial reduction may occasionally suppress subtle geometric details, thereby slightly affecting peak ADD(S) performance. Nevertheless, the overall results confirm that LDFPose achieves a favorable balance between computational efficiency and estimation accuracy.

The specific experimental visualization results are shown in Figure 6. As shown in Table 5,Compared with GDRNPP, LDFpose further improves MSSD from 94.9 to 95.6 and MSPD from 89.5 to 90.1, indicating consistent advantages in geometric consistency and projection accuracy brought by the adaptive multi-modal fusion mechanism. Although the VSD score is slightly lower than GDRNPP, the difference remains within 1%, suggesting comparable performance in visible surface alignment. These significant gains demonstrate the robustness of the proposed framework in solving complex scenes with heavy occlusion and multiple object interactions.

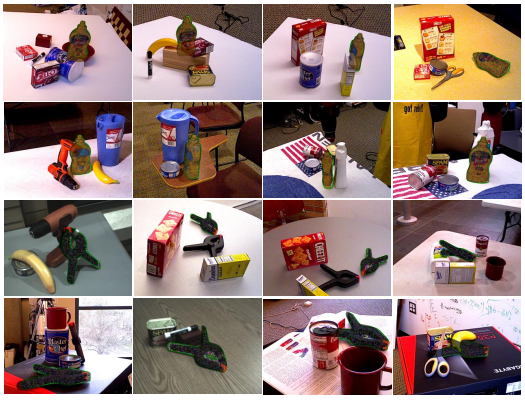

Figure 6: Qualitative results on the YCB-Video dataset.

Table 6
Quantitative evaluation results on the OccoA Dataset, where D-environments represent different environments.

Method	Metric	IS_e	IN_l	OS_e	ON_l
FFB6D	ADD(S)	91.8	92.1	92.6	91.5
	2 cm	94.3	95.2	95.3	94.2
DFTr	ADD(S)	93.1	93.8	93.2	92.4
	2 cm	95.7	96.4	95.8	94.4
LDFpose	ADD(S)	93.2	94.1	93.8	93.1
	2 cm	96.1	96.6	96.4	95.4

4.4.4. Evaluation using the user-defined OccoA dataset

To verify the feasibility of the proposed framework under more complex working conditions in real-world open scenarios, a custom dataset OccoA containing photographs with diverse scenarios and lighting conditions is first developed. OS_e and ON_l represent the outdoor shadow environment and outdoor natural lighting, while IS_e and IN_l denote the corresponding indoor working conditions, respectively. The ADD(S) AUC and the 2 cm metric are considered to validate the LDFpose, FFB6D, and DFTr using OccoA. As shown in Table 6, LDFpose achieves the best results under different lighting conditions and occlusion levels.

Overall, outdoor scenarios are more challenging due to issues such as image overexposure caused by object reflections from direct natural lighting. It is also noted that in indoor scenarios, the pose estimation will be predicted with a reduced accuracy, as shadowy lighting leads to the loss of image details. Results by LDFpose under the condition of IN_l outperform those under the outdoor natural lighting environment by 1.0% using the ADD(S) evaluation metric and by 1.2% using the 2cm metric. This indicates that the developed framework performs better under indoor uniform illumination condition.

Moreover, experimental results demonstrate that the ADD(S) and 2cm metrics under the condition of IN_l have the largest values as compared to results under other environments. It is noted that the proposed framework achieves the highest ADD(S) and 2cm metric values across all environments in comparison of FFB6D and DFTr. This can be attributed to the enhanced capability of LDFpose in handling the long-range dependencies.

Also, the values of ADD(S) and 2cm metric evaluations fluctuate across all four different environments, where the highest values are achieved under the condition of IN_l. This suggests that differences in environmental conditions affect the pose estimation results by a certain level of accuracy.As shown in Figure 7, LDFpose demonstrates the ability of generalization in terms of both accuracy and stability under dynamic environmental changes, making its application to real-time pose estimation feasible.

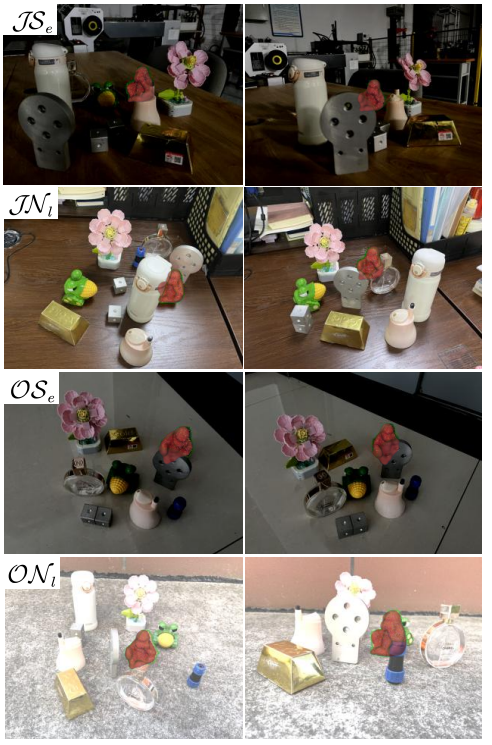

Figure 7: Qualitative results on the YCB-Video dataset.

4.4.5. Ablation Studies

To examine the contribution of each developed module implemented in LDFpose ablation experiments are conducted. Table 7 evaluates the contributions of EHT, AHK, and AMF on the YCB-Video dataset. Starting from the baseline, each component improves pose-estimation performance, while the whole combination of components generates the strongest overall result.

Compared to the baseline model, the LDFpose network performance in terms of pose estimation accuracy is significantly improved when incorporating Adaptive Modal Fusion (AMF). This enhancement can be attributed to the ability of AMF to filter minor features and establish the robust global feature connections across modalities, thereby significantly strengthening the feature representation capability.

Moreover, LDFpose outperforms the baseline model, demonstrating a particularly notable improvement of the ADD-S metric value, which could be observed to equate 96.6%. Specifically, in the fusion scenario, without any additional modules implemented in LDFpose, the accuracy is just achieved by 82.9%. While, with the incorporation of all modules, the accuracy is increased to 94.1%, indicating that the proposed framework exhibits the strong generalization ability, especially in complex scenarios.

4.4.6. Influence of Different Parameters in LDFpose

In this experiment, the influence of varying the number of the encoder layers and multi-self-attention (MSA) heads on the performance of the Transformer encoder block are investigated. Subsequently, a thorough analysis is performed considering different parameter configurations of LDFpose. Finally, the combination of the above two factors for results with the highest ADD(S) value is determined to construct the final model.

As the proposed framework includes four parameters: the numbers of encoder layers and MSA heads in both the pixel-wise feature extraction module and the multi-modal fusion module, the experimental results are summarized in Table 8. The effect of the number of encoder layers in the pixel-wise feature extraction block is examined while the values of the other three parameters are fixed. The similar analysis for effects of the other three parameters on ADD(S)

Table 7
Effect of components of LDFpose blocks on the YCB-Video dataset.

Method			Pose Result	
Pixel-wise feature extraction EHT	Multi-Modal fusion AHT	AMF	ADD-S (%)	ADD(S) (%)
			91.2	82.9
✓			91.8	83.0
	✓		92.4	86.6
		✓	93.4	90.6
✓	✓		93.8	91.3
✓		✓	94.0	92.4
	✓	✓	96.4	93.8
✓	✓	✓	96.6	94.1

Table 8
Experimental results of different parameters for transformer encoder.

Pixel-wise feature extraction		Multi-Modal fusion		ADD(S)
Layers	Heads	Layers	Heads	(%)
2	8	4	8	95.4
4	8	4	8	96.1
6	8	4	8	96.3
4	16	4	8	96.6
4	32	4	8	96.5
4	16	2	8	96.2
4	16	6	8	96.5
4	16	4	2	95.6
4	16	4	16	96.4

are conducted.

It is noted that the ADD(S) metric reaches its highest value of 96.6% when all four parameters are set to 4, 16, 4, and 8, respectively. Therefore, the values of 4, 16, 4, and 8 for these four parameters to construct the proposed framework are determined.

4.4.7. Analysis of model efficiency and performance

As shown in Figure 8, the reported FPS is calculated from the overall inference time, which includes the complete computational cost of the AHK module. LDFpose achieves an ADD(S) score approximately 13 percentage points higher than that of FFB6D. Meanwhile, its inference speed is nearly 8 times that of FFB6D. This demonstrates that the proposed framework attains the enhanced real-time processing capability, while preserving the competitive pose estimation accuracy. Consequently, LDFpose could offer the significant advantages in time-sensitive robotic manipulation scenarios, such as the industrial grasping, in which LDFpose enables robotic end-effectors to continuously acquire 6D object poses, providing sufficient time for the system to focus on the closed-loop motion planning for adjustment of end-effector trajectories subject to the task requirements.

To validate the lightweight characteristic of the proposed framework, the model complexity and pose estimation accuracy with representative RGB-D 6D pose estimation methods on the YCB-Video dataset are compared. In terms of computational complexity, as shown in Table 9, LDFpose requires only 36.4G FLOPs, which is significantly lower than PVN3D, DF, FFB6D and DFTr. Compared with DFTr, LDFpose reduces computational cost by approximately 50% while maintaining comparable accuracy. This demonstrates a substantially improved accuracy-efficiency trade-off. Regarding the model size, LDFpose contains 26.4M parameters, which is notably smaller than PVN3D, FFB6D and DFTr, while achieving superior or comparable performance. Although DenseFusion has slightly fewer parameters, its accuracy is significantly lower with the value of 82.9%. Therefore, LDFpose achieves high pose estimation precision while reducing computational and memory overhead. In summary, the above results confirm that LDFpose effectively balances performance and efficiency, validating its lightweight design for real-time 6D object pose estimation.

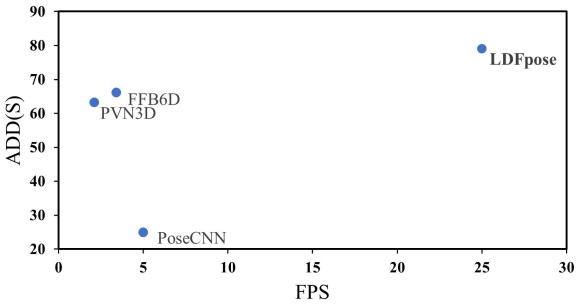

Figure 8: Balance the speed and accuracy.

Table 9
Performance comparison of different methods on the YCB-Video dataset.

	PVN3D	DF	FFB6D	DFTr	LDFpose
ADD(S)	91.8	82.9	92.7	94.4	93.8
FLOPs (G)	105.4	38.3	54.6	72.8	36.4
Parameters (M)	39.2	22.6	33.8	36.3	26.4

5. Conclusion

In this work, a novel 6D pose estimation framework called LDFpose is proposed to leverage feature extraction and information fusion of RGB images and point cloud data for the improved robotic grasp precision under the consideration of different lighting and high degrees of occlusion conditions. A transformer-based network is introduced to extract color and geometric features leveraging the designed EHT (Embedding-Free Hierarchical Transformer) and AHT (Adaptive Hierarchical Transformer) modules, which enable more efficient aggregation of global semantic information while boosting performance. Moreover, an adaptive multi-modal fusion mechanism is developed to strengthen global representations.

Results demonstrate that the proposed architecture outperforms state-of-the-art (SOTA) approaches in specific aspects. On the Occlusion Linemod dataset, an ADD-0.1d value of 79.0 is achieved by 1.3% higher than 77.7 by DFTr. It is noted that LDFpose achieves the highest ADD(S) and 2cm metric values across all evaluation metrics under varying conditions as compared with FFB6D and DFTr.

Overall, the proposed framework enables highly accurate and real-time 6D object pose estimation in challenging environments and provides a useful insight into the development of robust vision systems under the considerations of computational efficiency and effective multimodal fusion in the fields of robotics, industrial automation, and augmented reality.

Additionally, the current research retains limitations, for example, the model lacks few-shot/zero-shot pose estimation capabilities. It is also noted that the framework exhibits to some extents performance degradation on unseen object categories. Future extensions could target real-time object pose estimation applications, such as autonomous driving or robotic grasping, to further improve the generalization of LDFpose for pose estimation in more complex situations.

Acknowledgements

References

[1] Jesse Levinson, Jake Askeland, Jan Becker, Jennifer Dolson, David Held, Soeren Kammel, J Zico Kolter, Dirk Langer, Oliver Pink, Vaughan Pratt, et al. Towards fully autonomous driving: Systems and algorithms. In *2011 IEEE intelligent vehicles symposium (IV)*, pages 163–168. IEEE, 2011.

[2] Gu Wang, Fabian Manhardt, Xingyu Liu, Xiangyang Ji, and Federico Tombari. Occlusion-aware self-supervised monocular 6d object pose estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(3):1788–1803, 2021.

[3] Xinke Deng, Yu Xiang, Arsalan Mousavian, Clemens Eppner, Timothy Bretl, and Dieter Fox. Self-supervised 6d object pose estimation for robot manipulation. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3665–3671. IEEE, 2020.

[4] Yiming Li, Tao Kong, Ruihang Chu, Yifeng Li, Peng Wang, and Lei Li. Simultaneous semantic and collision learning for 6-dof grasp pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3571–3578. IEEE, 2021.

[5] Hayet Belghit, Abdelkader Bellarbi, Nadia Zenati, and Samir Otmane. Vision-based pose estimation for augmented reality: a comparison study. *arXiv preprint arXiv:1806.09316*, 2018.

[6] Yan Zhao, Shaobo Zhang, Wanqing Zhao, Ying Wei, and Jinye Peng. Augmented reality system based on real-time object 6d pose estimation. In *2023 2nd International Conference on Image Processing and Media Computing (ICIPMC)*, pages 27–34. IEEE, 2023.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] Shaoqing Ren. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497*, 2015.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[10] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.

[11] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[12] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pages 2564–2571. Ieee, 2011.

[13] Tony Lindeberg. Scale invariant feature transform. 2012.

[14] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (fpfh) for 3d registration. In *2009 IEEE international conference on robotics and automation*, pages 3212–3217. IEEE, 2009.

[15] J Redmon. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[16] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[17] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE international conference on computer vision*, pages 3828–3836, 2017.

[18] Wadim Kehl, Fausto Milletari, Federico Tombari, Slobodan Ilic, and Nassir Navab. Deep learning of local rgb-d patches for 3d object detection and 6d pose estimation. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pages 205–220. Springer, 2016.

[19] W Yuan, X Gu, Z Dai, S Zhu, and P Tan. New crfs: Neural window fully-connected crfs for monocular depth estimation. arxiv 2022. *arXiv preprint arXiv:2203.01502*, 2022.

[20] Yisheng He, Wei Sun, Haibin Huang, Jianran Liu, Haoqiang Fan, and Jian Sun. Pvn3d: A deep point-wise 3d keypoints voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11632–11641, 2020.

[21] Yisheng He, Haibin Huang, Haoqiang Fan, Qifeng Chen, and Jian Sun. Ffb6d: A full flow bidirectional fusion network for 6d pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3003–3013, 2021.

[22] Chi Li, Jin Bai, and Gregory D Hager. A unified framework for multi-view multi-class object pose estimation. In *Proceedings of the european conference on computer vision (eccv)*, pages 254–269, 2018.

[23] Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

[24] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[25] Chen Wang, Danfei Xu, Yuke Zhu, Roberto Martín-Martín, Cewu Lu, Li Fei-Fei, and Silvio Savarese. Densefusion: 6d object pose estimation by iterative dense fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3343–3352, 2019.

[26] Danfei Xu, Dragomir Anguelov, and Ashesh Jain. Pointfusion: Deep sensor fusion for 3d bounding box estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 244–253, 2018.

[27] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[28] Jun Zhou, Kai Chen, Linlin Xu, Qi Dou, and Jing Qin. Deep fusion transformer network with weighted vector-wise keypoints voting for robust 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13967–13977, 2023.

[29] Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[30] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in neural information processing systems*, 34:12077–12090, 2021.

[31] Ola Benderius, Christian Berger, and Krister Blanch. Are we ready for beyond-application high-volume data? the reeds robot perception benchmark dataset. *arXiv preprint arXiv:2109.08250*, 2021.

[32] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286, 2019.

[33] 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*, pages 6738–6748, 2022.

[34] Bugra Tekin, Sudipta N Sinha, and Pascal Fua. Real-time seamless single shot 6d object pose prediction. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 292–301, 2018.

[35] Sergey Zakharov, Ivan Shugurov, and Slobodan Ilic. Dpod: 6d pose object detector and refiner. In *Proceedings of the IEEE/CVF international*

conference on computer vision, pages 1941–1950, 2019.

[36] Zhigang Li, Gu Wang, and Xiangyang Ji. Cdpn: Coordinates-based disentangled pose network for real-time rgb-based 6-dof object pose estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7678–7687, 2019.

[37] Yi Li, Gu Wang, Xiangyang Ji, Yu Xiang, and Dieter Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018.

[38] Yannick Bukschat and Marcus Vetter. Efficientpose: An efficient, accurate and scalable end-to-end 6d multi object pose estimation approach. *arXiv preprint arXiv:2011.04307*, 2020.

[39] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. Spie, 1992.

[40] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Ep n p: An accurate o (n) solution to the p n p problem. *International journal of computer vision*, 81:155–166, 2009.

[41] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[42] Kentaro Wada, Edgar Sucar, Stephen James, Daniel Lenton, and Andrew J Davison. Morefusion: Multi-object reasoning for 6d pose estimation from volumetric fusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14540–14549, 2020.

[43] Guangliang Zhou, Yi Yan, Deming Wang, and Qijun Chen. A novel depth and color feature fusion framework for 6d object pose estimation. *IEEE Transactions on Multimedia*, 23:1630–1639, 2020.

[44] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.

[45] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *Proceedings of the IEEE international conference on computer vision*, pages 1521–1529, 2017.

[46] Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. Pvnet: Pixel-wise voting network for 6dof pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4561–4570, 2019.

[47] Gu Wang, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16611–16621, 2021.

[48] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[49] He Chen and Xiaoyu Yue. Swimtrans net: a multimodal robotic system for swimming action recognition driven via swin-transformer. *Frontiers in Neurorobotics*, 18:1452019, 2024.

[50] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 568–578, 2021.

[51] Zhongqun Zhang, Wei Chen, Linfang Zheng, Aleš Leonardis, and Hyung Jin Chang. Trans6d: Transformer-based 6d object pose estimation and refinement. In *European Conference on Computer Vision*, pages 112–128. Springer, 2022.

[52] Arash Amini, Arul Selvam Periyasamy, and Sven Behnke. Yolopose: Transformer-based multi-object 6d pose estimation using keypoint regression. In *International Conference on Intelligent Autonomous Systems*, pages 392–406. Springer, 2022.

[53] Yuning Ye and Hanhoon Park. Fusionnet: An end-to-end hybrid model for 6d object pose estimation. *Electronics*, 12(19):4162, 2023.

[54] Hyunwoo Yu, Yubin Cho, Beoungwoo Kang, Seunghun Moon, Kyeongbo Kong, and Suk Ju Kang. Embedding-free transformer with inference spatial reduction for efficient semantic segmentation. 2024.

[55] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary 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 11*, pages 548–562. Springer, 2013.

[56] Eric Brachmann, Alexander Krull, Frank Michel, Stefan Gumhold, Jamie Shotton, and Carsten Rother. Learning 6d object pose estimation using 3d object coordinates. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II 13*, pages 536–551. Springer, 2014.

[57] Wei Chen, Xi Jia, Hyung Jin Chang, Jinming Duan, and Ales Leonardis. G2l-net: Global to local network for real-time 6d pose estimation with embedding vector features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4233–4242, 2020.

[58] Chen Song, Jiaru Song, and Qixing Huang. Hybridpose: 6d object pose estimation under hybrid representations. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 431–440, 2020.

[59] Guangyuan Zhou, Huiqun Wang, Jiaxin Chen, and Di Huang. Pr-gcn: A deep graph convolutional network with point refinement for 6d pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2793–2802, 2021.

[60] Yangzheng Wu, Mohsen Zand, Ali Etemad, and Michael Greenspan. Vote from the center: 6 dof pose estimation in rgb-d images by radial keypoint voting. In *European Conference on Computer Vision*, pages 335–352. Springer, 2022.

[61] Xingyu Liu, Ruida Zhang, Chenyangguang Zhang, Gu Wang, Jiwen Tang, Zhigang Li, and Xiangyang Ji. Gdrnpp: A geometry-guided and fully learning-based object pose estimator. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[62] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 641–656, 2018.